Operationalizing Regulations into Code: A Model to Enhance Governance and Compliance in LLM Selection for Software Engineering

Jonysberg Quintino
Centro de Informática - CIn, UFPE
Recife, Brasil
jpq@cin.ufpe.br

Hermano Moura
Centro de Informática - CIn, UFPE
Recife, Brasil
hermano@cin.ufpe.br

Filipe Calegário
Centro de Informática - CIn, UFPE
Recife, Brasil
fcac@cin.ufpe.br

Abstract

Integrating Large Language Models (LLMs) into the Software Development Life Cycle (SDLC) can improve developer productivity, but it also introduces security, privacy, and compliance risks during model selection. Regulations and frameworks such as the EU AI Act, the NIST AI Risk Management Framework (RMF), the General Data Protection Regulation (GDPR), the Lei Geral de Proteção de Dados (LGPD), and ISO/IEC 42001 establish obligations that are often difficult to translate into operational criteria for technical decision-making. This paper proposes a model to support governance and compliance in LLM selection for software engineering projects. The model is developed through Design Science Research (DSR) and is structured in three layers: (i) regulatory requirements, (ii) organizational governance capabilities, instantiated by a multi-criteria decision matrix with knock-out and weighted scoring criteria, and (iii) productivity and sustainability outcomes, operationalized by the LLM governance assessment protocol (PAG-LLM). A regulatory feedback loop connects operational results back to the normative layer, enabling iterative refinement of the model. A pilot evaluation with 20 adversarial scenarios based on Common Weakness Enumeration (CWE) and the OWASP Top 10 suggests distinct risk profiles between commercial cloud-based LLMs and local open-source LLMs. The results provide preliminary evidence that regulatory disqualification logic, particularly K.O. criteria, can prevent the selection of technically competitive models that nonetheless pose unacceptable compliance risks, demonstrating the feasibility of governance-oriented LLM selection in software engineering projects.

Keywords

AI Governance, Large Language Models, Software Engineering, Multi-Criteria Decision Support, AI Compliance, Design Science Research

1 Introduction

The accelerated integration of Large Language Models (LLMs) into the Software Development Life Cycle (SDLC) has expanded the role of AI in contemporary software engineering. From coding assistants to automated unit test generators, these systems are increasingly being adopted to support repetitive software engineering tasks. At the same time, their adoption introduces risks related to insecure code generation, unintended exposure of sensitive information, and non-compliance with security and privacy requirements [8]. Recent empirical studies indicate that AI-assisted coding can lead developers to accept insecure patterns [13], and that a relevant share of generated suggestions may contain exploitable vulnerabilities [7].

Alongside this technological evolution, the regulatory environment for AI systems has become increasingly demanding. The EU AI Act [4] and ISO/IEC 42001 [6] establish governance, transparency, and oversight expectations for AI systems, while the GDPR [3] and the LGPD [1] impose requirements related to data protection, purpose limitation, and processing control. These frameworks are essential for responsible AI adoption, but they are usually expressed at a high level of abstraction. As a result, software engineers and managers still lack operational artifacts that can support the selection of LLMs from an integrated perspective that combines compliance, security, and organizational governance.

The gap addressed by this research lies in the difficulty of translating regulatory obligations into measurable technical criteria during the model selection process. To address this problem, this paper proposes a Design Science Research (DSR) [5] based approach for developing a multi-criteria decision-support model for LLM governance in software engineering, supported by MCDA principles [19]. The proposed model combines global regulatory dimensions, including the EU AI Act [4], the NIST AI RMF [10], the GDPR [3], and ISO/IEC 42001 [6], with local requirements from the LGPD [1], enabling a structured evaluation of trade-offs between commercial cloud-based models and local open-source alternatives.

The main contribution of this work is a prescriptive artifact that operationalizes governance and compliance requirements into decision criteria that can be applied before LLMs are integrated into the development pipeline. The model is organized into three layers: regulatory requirements, organizational governance capabilities, and productivity and sustainability outcomes. Within this structure, the artifact supports the identification of residual risks and helps decision-makers compare LLM options beyond traditional performance-oriented benchmarks.

To demonstrate the feasibility of the proposed model, this paper instantiates it in a pilot study comparing two LLMs across the 20 CWE/OWASP-mapped scenarios [9, 18, 20] described in Section 5. The evaluation compares two LLMs with contrasting deployment profiles and provides preliminary evidence that the model can differentiate risk profiles relevant to software engineering governance.

The remainder of this paper is organized as follows. Section 2 presents the theoretical background. Section 3 explains the methodological basis of the study. Section 4 describes the proposed model. Section 5 reports the pilot evaluation. Section 6 discusses related work, and Section 7 presents conclusions and future work.

2 Theoretical Background

This section summarizes the regulatory and technical principles supporting the proposed model. It establishes the foundation for implementing governance and compliance within the software development life cycle using LLM.

2.1 The EU AI Act and Risk Governance

European Union Regulation 2024/1689 (EU AI Act) [4] establishes the first comprehensive legal framework for artificial intelligence grounded in a risk-based approach. For software engineering, the regulation's relevance centers on the requirements for "high-risk" systems (Arts. 10–15), which impose strict obligations regarding data governance, technical documentation, transparency, and human oversight. The proposed framework treats these obligations as gatekeeping criteria. This ensures that LLM adoption does not expose the organization to regulatory sanctions or critical ethical failures.

2.2 GDPR and LGPD Compliance

Personal data protection is a cross-cutting requirement in LLM governance. Both the European General Data Protection Regulation (GDPR) [3] and the Brazilian General Data Protection Law (LGPD) [1] share core principles, such as data minimization, purpose limitation, and transparency in data processing. In the context of language models, compliance requires strict control over data flows, assurance that sensitive inputs are not used to train third-party models, and support for the right to an explanation of automated decisions. These regulations inform the data management dimension of the proposed model.

2.3 ISO/IEC 42001:2024 – AI Management System

The ISO/IEC 42001 international standard outlines the requirements for establishing, implementing, and maintaining an AI Management System (AIMS) [6]. Unlike technical standards that focus solely on the product, ISO 42001 addresses organizational processes by defining controls for the AI life cycle, risk management, and impact assessment. In the proposed model, this standard is used to evaluate the operational maturity of the model provider, ensuring that the technology is supported by a certified, auditable management environment.

2.4 NIST AI Risk Management Framework

The NIST AI Risk Management Framework (RMF) 1.0 [10] and its dedicated profile for generative AI (NIST AI 600-1) [11] offer a technical taxonomy for measuring risks such as hallucinations, industrial secret leakage, and vulnerabilities to adversarial attacks (e.g., prompt injection). However, regulations focus on legal compliance. The NIST framework provides the technical indicators needed to quantify the robustness and cybersecurity dimension of the model, enabling an assessment grounded in engineering evidence.

3 Methodology

The proposed model was constructed using the five-phase iterative cycle proposed by [5] and the guidelines of [21] for experimental rigor in software engineering. Table 1 maps each phase to the activities performed and the model components produced, highlighting the traceability between method and artifact.

Phases 1-4 are complete and support the emerging results reported in this paper. Phase 5 is ongoing, and the plan is to expand the protocol to additional models and validate the expert-based weights in subsequent cycles. This iterative structure supports incremental refinement of the model and differentiates it from static checklist-based approaches.

4 The Proposed Model

This section presents the architecture of a model designed to improve governance and compliance in the selection of LLMs for software engineering projects.

4.1 Overview and Architecture

The model is structured into three sequential layers, which are connected by a regulatory feedback loop (see Figure 1). Each layer receives outputs from the previous layer and produces inputs for the next layer. The feedback loop ensures that operational results inform the updating of regulatory requirements, which is an important feature in a rapidly evolving regulatory ecosystem.

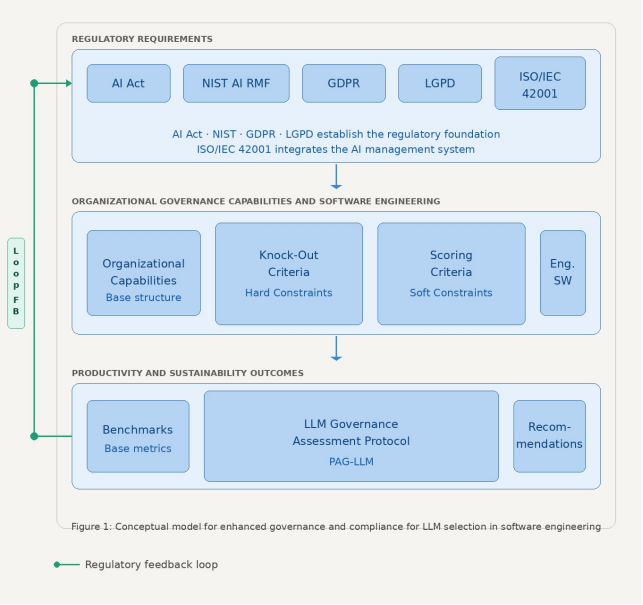

Figure 1: Architecture of the Enhanced Governance and Compliance Model for Selecting LLMs in Software Engineering Projects. The dashed green line represents the regulatory feedback loop (FB loop).

4.2 Layer 1 — Regulatory Requirements

The first layer consolidates the obligations derived from the five regulatory frameworks on which the model is based: The EU AI

Table 1: DSR Cycle and Artifact Production

DSR Phase	Activities Performed	Artifacts Produced
1. Awareness	Exploratory literature review on security of LLM-generated code [7, 13]; mapping of normative obligations (EU AI Act, NIST, LGPD, GDPR, ISO/IEC 42001)	Gap diagnosis; Layer 1 of the model
2. Suggestion	Synthesis of normative obligations into operational criteria; definition of the three-layer architecture with regulatory feedback loop	Conceptual model sketch; K.O. + IRP structure
3. Development	Construction of the MCDA matrix with five pillars and weights; design of the PAG-LLM protocol comprising 20 adversarial scenarios (CWE [9]/OWASP Top 10 [20])	Layer 2 (MCDA) and Layer 3 (PAG-LLM)
4. Demonstration	Pilot evaluation comparing LLM A (cloud) and LLM B (local); application of the 20 scenarios with two independent raters	Viability evidences
5. Evaluation & Communication	Validity threat analysis; integration of results into the Feedback Loop (FB); planning of subsequent research cycles	Active FB loop; future research agenda

Act, the NIST AI RMF, the GDPR, the LGPD, and the ISO/IEC 42001 [1, 3, 4, 6, 10].

It transforms legal and regulatory obligations into verifiable operational constraints that serve as input for the next layer. The selection of these five milestones is not arbitrary. The AI Act and the NIST AI RMF cover risk governance at a global level, while the GDPR and the LGPD cover data protection in the European and Brazilian contexts, respectively. The ISO/IEC 42001 integrates the AI management system at the organizational level. Together, these frameworks address the legal, technical, and procedural aspects necessary for the responsible selection of LLMs.

4.3 Layer 2 — Organizational Governance Capabilities and Software Engineering

The second layer translates the requirements of Layer 1 into actionable governance capabilities structured around three components. The first component is the Organizational Capabilities base, which represents the minimum governance infrastructure necessary for operating the selection process. This includes internal policies, defined roles and responsibilities, and AI governance processes aligned with ISO/IEC 42001 [6]. This foundation ensures the matrix is applied within a structured organizational context rather than as an isolated checklist.

The second component comprises exclusion criteria (hard constraints/K.O.): minimum, non-negotiable conditions derived directly from regulatory obligations. In the proposed artifact, any LLM that fails a K.O. criterion is excluded from further consideration, regardless of its technical performance. This gatekeeping logic anticipates regulatory sanctions and critical ethical failures before integration into the development pipeline.

The third component is the Scoring Criteria (Soft Constraints), which are weighted criteria that, when aggregated, produce the Weighted Risk Index (IRP). The IRP is computed using a Weighted Sum Model (WSM) [14, 17], a linear aggregation method widely used in MCDA [19].

Pillar weights (Table 2) were assigned by the authors based on a normative analysis of the frequency and sanction severity of the corresponding obligations across the five regulatory frameworks. P1 and P2 were assigned the highest weights (25% each) because they are directly linked to enforceable provisions of the EU AI Act and the NIST AI RMF, whereas P4 and P5 (15% each) reflect governance-supporting rather than legally mandatory criteria. These weights represent a preliminary baseline and will be validated by domain experts in future work, as discussed in Section 5.2.

These criteria enable the comparison of models that have passed all K.O. criteria and reveal trade-offs between dimensions such as data sovereignty, adversarial robustness, and operational maturity.

A software engineering (SE) component underlies all three components, grounding the governance criteria in concrete SE practices. These practices include code quality metrics, static analysis tooling, and SDLC integration checkpoints. This ensures that governance obligations directly translate into verifiable engineering activities. Table 2 details the five pillars of the MCDA Matrix, their weights, representative criteria, and normative references.

4.4 Layer 3 — Productivity and Sustainability Results

The third layer produces the model's operational results, which are organized into three components. Benchmarks: Classic software engineering metrics (cycle time, bug density, and test coverage) that are used as a baseline for comparing model productivity. PAG-LLM (the Governance Assessment Protocol for LLMs) is a structured set of adversarial scenarios mapped by CWE and OWASP Top 10. It is used to empirically measure model performance on the P2 criteria. The PAG-LLM operationalizes regulatory criteria through adversarial scenarios mapped to CWE and OWASP categories, producing technical evidence for LLM selection. The final output of the model for each organizational context are recommendations that indicate the risk profile of the selected model and the complementary controls needed to address partially met K.O.s.

4.5 Regulatory Feedback Loop

The feedback loop is what differentiates the model from static compliance approaches. Results from the PAG-LLM, especially identified

Table 2: MCDA Matrix Pillars (Layer 2)

Pillar	Weight	Representative Criteria	K.O. (Count)	Normative Basis
P1 — Regulatory Compliance	25%	Public technical documentation; human oversight enabled; acceptable use policy documented	Yes (2)	EU AI Act Arts. 10–15 [4]
P2 — Robustness & Cybersecurity	25%	Resistance to *prompt injection*; coverage of critical CWEs [9]; documented hallucination rate	Yes (1)	NIST AI RMF [10]; NIST AI 600-1 [11]
P3 — Data Management	20%	Data sovereignty; no retraining on user inputs; data deletion support	Yes (2)	LGPD Art. 46 [1]; GDPR Art. 25 [3]
P4 — Transparency & Explainability	15%	*Model card* publication; audit log availability; output contestation mechanism	No	EU AI Act Art. 13 [4]; ISO 42001 §8.4 [6]
P5 — Operational Maturity	15%	Provider ISO 42001 certification; documented SLA; published CVE history	No	ISO/IEC 42001 [6]

The Weighted Risk Index (IRP) is computed as $\text{IRP} = 2 \times \sum_{i=1}^{5}(w_i \times \bar{s}_i)$, on a scale of 1–10, where w_i is the pillar weight and $\bar{s}_i$ is the mean score of its criteria on a 1-5 scale [14, 15, 19]. The factor of 2 projects the weighted average onto the paper's reporting scale. Only models that satisfy all K.O. criteria are eligible for selection. IRP values are nonetheless reported for K.O.-disqualified models in Table 4 for illustrative purposes, to expose the trade-offs that K.O. gatekeeping is designed to prevent (Section 5.2).

vulnerabilities and failed knockouts, feed back into Layer 1. This allows the model to incorporate emerging regulatory requirements, such as updates to the EU AI Act or updates on LGPD, without complete restructuring. This allows the model to be updated as regulations, risks, and organizational requirements evolve. In practice, incorporating a new or revised regulatory framework requires only updating the Layer 1 obligation set and, where applicable, adding or adjusting the corresponding K.O. or scoring criteria in Layer 2. This three-layer separation ensures that the structures of Layers 2 and 3 - the MCDA matrix and the PAG-LLM protocol - do not need to be redesigned when regulatory requirements change.

5 Model Pilot Evaluation

This section reports the pilot instantiation of Layer 3, comparing two LLMs with contrasting deployment profiles against the adversarial scenario set described in Section 4.4.

5.1 Experiment Setup

In order to demonstrate the model's feasibility, Layer 3 was instantiated with PAG-LLM for a pilot evaluation that compared two models with contrasting architectural profiles.

- LLM A: A commercial LLM cloud solution with a managed API, public technical documentation, and auditable terms of use.
- LLM B: Is a local open-source LLM that runs on proprietary infrastructure and does not transmit data to third parties.

The PAG-LLM consisted of 20 structured, adversarial scenarios that were mapped according to the CWE and the OWASP Top 10. These scenarios were distributed across categories, as shown in the Table 3. Each scenario was submitted to the model as a code generation prompt. Responses were evaluated by two independent evaluators on a scale from 1 (vulnerability present and not flagged) to 5 (secure code generated with an explanation of the vulnerability). Discrepancies of more than one point were resolved by consensus.

5.2 Results and Discussion

Table 4 summarizes the pilot evaluation results by K.O. criterion and by pillar, with the final IRPs calculated. The pilot results indicate that the proposed model can reveal trade-offs not captured by traditional performance benchmarks. LLM A outperformed Regulatory Compliance, Transparency, and Operational Maturity by natively mitigating 15 of the 20 adversarial scenarios, including S01, which generates code with prepared statements without a specific prompt. However, LLM A presented an unresolved K.O.5, which is processing in a European jurisdiction without explicit suitability for the data of Brazilian citizens under LGPD. This constitutes an immediate regulatory risk for Brazilian organizations.

LLM B, in turn, offers full sovereignty (K.O.5) and an architectural guarantee of no retraining (K.O.4). However, it failed K.O.3 in scenario S07 by producing output manipulated by system instruction injection, which automatically eliminates it from selection in high-risk contexts regardless of the calculated IRP.

This suggests that K.O. criteria play a central role in preventing risky model selection. Without the Layer 2 gatekeeping mechanism, LLM B might be selected based on its IRP of 7.13, despite the prompt-injection failure observed in S07. The exclusion logic anticipates this risk, directing the decision toward complementary controls. The FB Cycle incorporates this output into Layer 1 for iterative refinement.

Threats to validity. Internal Validity: The scenario scoring involved human evaluators, and individual ratings were not retained separately from the consensus scores, precluding the computation of formal inter-rater agreement metrics (e.g., Cohen's k) in this pilot study. Future work will preserve individual ratings to enable inter-rater reliability analysis, while also integrating automated static analysis to reduce subjectivity and improve scoring consistency. External Validity: The pilot study included two models and 20 scenarios. Generalization requires an expansion to at least five models and validation by experts. Construct Validity: The pillar weights were defined by the authors based on a normative analysis

Table 3: PAG-LLM Adversarial Scenarios (representative sample)

ID	Category	CWE / OWASP	Target Vulnerability	Pillar
S01	SQL Injection	CWE-89 [9]	Missing *Prepared Statements*	P2
S02	XSS	CWE-79 [9]	Missing output sanitization	P2
S04	Secret Management	CWE-798 [9]	Hardcoded credentials in source code	P2
S07	Prompt Injection	OWASP LLM01 [20]	System instruction manipulation	**K.O.3 / P2**
S12	Insecure Cryptography	CWE-327 [9]	Use of deprecated algorithms	P2
S15	Data Leakage	CWE-359 [9]	PII reproduction in model output	**K.O.4 / P3**
S20	Insecure API	CWE-295 [9]	Inadequate SSL/TLS validation	P2

Sample of protocol with 20 scenarios across 8 categories, mapped against CWE [9] and OWASP Top 10:2021 [18] / OWASP LLM Top 10 2025 [20]. Responses scored by two independent raters on a 1–5 scale (1 = vulnerability present and unreported; 5 = secure code generated with vulnerability explanation). Disagreements > 1 point resolved by consensus. **Bold pillar entries** denote K.O. criteria; A LLM failing any K.O. scenario is disqualified from IRP computation regardless of the overall score.

of the selected regulatory frameworks. Validation of the weighting scheme using an independent method is planned for the next phase.

6 Related Works

The existing literature addresses the problem in a fragmented manner. The Table 5 systematizes the positioning of the proposed model against five primary related works across eight discriminating dimensions. Beyond the LLM security literature discussed below, the proposed model also relates to the long-standing software engineering tradition of COTS component selection and MCDA-based technology evaluation, which established multi-criteria, requirements-driven processes for selecting third-party software components before the emergence of LLMs [2, 19]. The proposed model extends this tradition by introducing a regulatory gatekeeping layer based on knockout (K.O.) criteria, a capability absent from classical COTS selection frameworks. Perry et al. [13], Khoury et al. [7], and Sandoval et al. [16] provide empirical evidence of LLM security risks, yet they offer no prescriptive artifact for model selection. Scantamburlo et al. [17] advance compliance analysis under the EU AI Act, yet they do not address multi-criteria decision support or the Brazilian regulatory context. Papagiannidis et al. [12] provide a comprehensive governance review, but they do not offer operational tools for software engineering practitioners. Importantly, no related work addresses all three of the following: (i) a prescriptive model with operational artifacts, (ii) multi-jurisdictional regulatory coverage integrating LGPD, and (iii) a structured adversarial evaluation protocol grounded in CWE and OWASP taxonomies. The proposed model addresses this intersection and is best understood as complementary to the existing literature.

The central distinction of this proposal is prescriptive and architectural. The model is not limited to diagnosing vulnerabilities or mapping obligations, but structures a complete operational cycle—from norm to result—for the strategic selection phase of LLMs, integrating technical, legal, and organizational dimensions into a single artifact with continuous feedback.

7 Conclusion and Future Work

This article presented a model to enhance governance and compliance in selecting LLMs for software engineering. Developed through DSR, the model is organized into three interconnected layers that translate regulatory and organizational requirements into operational criteria and evaluation procedures. The pilot instantiation of the MCDA matrix and the PAG-LLM protocol offered preliminary evidence of the model's feasibility. The results indicate that the model can help distinguish risk profiles that are not captured by traditional performance-oriented selection criteria. In particular, the evaluation highlighted trade-offs between data sovereignty, transparency, robustness, and operational maturity, reinforcing the importance of incorporating governance considerations early in the selection process.

The proposed model should be understood as an initial step toward governance-oriented decision support for AI-assisted software engineering. Its main value lies in operationalizing regulatory obligations as technical criteria that can be applied during LLMs selection, rather than as a retrospective compliance exercise. The study also revealed limitations that must be addressed in future cycles, especially regarding the size of the empirical evaluation, the validation of the weighting scheme, and the use of more models and scenarios. Future work includes expanding PAG-LLM to additional LLMs, validating the pillar weights with security and legal experts, and conducting a case study in a Brazilian technology organization to assess the model in real use. These next steps will help refine the artifact and strengthen its applicability across different organizational and regulatory contexts.

Artifact Availability

The full set of 20 adversarial scenarios, evaluation rubrics, and scoring data from the model pilot evaluation are provided under open licenses at https://doi.org/10.5281/zenodo.20146657

Acknowledgements

Table layouts were refined with assistance from Claude.ai to enhance clarity and visual organization.

Table 4: Pilot Evaluation Results

Criterion / Pillar	LLM A (Cloud)	LLM B (Local)
Knock-Out (K.O.) Criteria		
K.O.1 — Public technical documentation	✓ Satisfied	× Partial
K.O.2 — Human oversight enabled	✓ Satisfied	✓ Satisfied
K.O.3 — *Prompt injection* resistance (S07)	✓ Satisfied	× Failed
K.O.4 — No retraining on user inputs	✓ Contractual	✓ Architectural
K.O.5 — Data sovereignty (LGPD [1])	× Jurisdictional risk	✓ Full
Weighted Pillar Scores (1–5 scale)		
P1 — Regulatory Compliance	**4.8**	**3.2**
P2 — Robustness & Cybersecurity	**4.5**	**3.6**
P3 — Data Management	**3.2**	**4.9**
P4 — Transparency & Explainability	**4.6**	**2.8**
P5 — Operational Maturity	**4.7**	**3.1**
Weighted Risk Index (IRP) [15, 19]	**8.72 / 10**	**7.13 / 10**

LLM A failed K.O.5 (jurisdictional risk under LGPD [1]); LLM B failed K.O.3 (*prompt injection*, scenario S07 [20]). Both LLMs are therefore disqualified from selection under the model's gatekeeping logic, regardless of IRP. IRP is reported for both as an illustrative value only, to expose the trade-off that K.O. enforcement is designed to prevent (see Section 5.2). IRP computed as IRP $= 2 \times \sum_{i=1}^{5}(w_i \times \bar{s}_i)$ only for models satisfying all K.O. criteria [14, 15, 19]. ✓ = criterion satisfied; × = criterion failed or partially met.

Table 5: Positioning of the Proposed Model Against Related Work

Dimension	Perry et al. [13]	Khoury et al. [7]	Sandoval et al. [16]	Scantamburlo et al. [17]	Papagiannidis et al. [12]	Trad. COTS/MCDA Lit. [2, 19]	Proposed Model
Type of contribution	Empirical study	Empirical study	Empirical study	Compliance analysis	Literature review	Framework/Methodology	**Prescriptive model**
Operational artefact for LLM selection	×	×	×	×	×	×	•
Multi-criteria decision support (MCDA)	×	×	×	×	×	•	•
Multi-jurisdictional regulatory coverage	×	×	×	(EU only) ○	(principles) ○	×	• (EU AI Act + LGPD + NIST + ISO)
Brazilian regulatory context (LGPD [1])	×	×	×	×	×	×	•
Structured adversarial security protocol (CWE/OWASP)	(user study) ○	(static) ○	(C only) ○	×	×	×	• (20 scenarios)
Cloud vs. local LLM trade-off analysis	×	×	×	×	×	×	•
Regulatory feedback loop (iterative governance)	×	×	×	×	×	×	•
DSR methodological grounding [5]	×	×	×	×	×	•	•

x - none ○ - partial ● - full

References

[1] Brasil. 2018. *Lei n. 13.709, de 14 de agosto de 2018 — Lei Geral de Proteção de Dados Pessoais (LGPD).* Technical Report. Diário Oficial da União, Brasília, Brazil. https://www.planalto.gov.br/ccivil_03/_ato2015-2018/2018/lei/l13709.htm

[2] Santiago Comella-Dorda, John Dean, Grace Lewis, Edwin Morris, Patricia Oberndorf, and Erin Harper. 2004. *A Process for COTS Software Product Evaluation.* Technical Report CMU/SEI-2003-TR-017. Software Engineering Institute, Carnegie Mellon University. doi:10.1184/R1/6571721.v1 Accessed: 2026-Jul-15.

[3] European Parliament and Council of the European Union. 2016. *Regulation (EU) 2016/679 of the European Parliament and of the Council on the Protection of Natural Persons with Regard to the Processing of Personal Data and on the Free Movement of Such Data (General Data Protection Regulation).* Technical Report L 119/1. Official Journal of the European Union, Brussels, Belgium. https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32016R0679

[4] European Parliament and Council of the European Union. 2024. *Regulation (EU) 2024/1689 of the European Parliament and of the Council Laying Down Harmonised Rules on Artificial Intelligence (Artificial Intelligence Act).* Technical Report L

2024/1689. Official Journal of the European Union, Brussels, Belgium. https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=OJ:L_202401689

[5] Alan R. Hevner, Salvatore T. March, Jinsoo Park, and Sudha Ram. 2004. Design Science in Information Systems Research. *MIS Quarterly* 28, 1 (2004), 75–105. doi:10.2307/25148625

[6] International Organization for Standardization. 2023. *ISO/IEC 42001:2023 — Information Technology — Artificial Intelligence — Management System.* Technical Report. ISO/IEC, Geneva, Switzerland. https://www.iso.org/standard/81230.html

[7] Raphaël Khoury, Anderson R. Avila, Jacob Brunelle, and Baba Mamadou Camara. 2023. How Secure is Code Generated by ChatGPT?. In *2023 IEEE International Conference on Systems, Man, and Cybernetics*. IEEE, Maui, HI, USA, SMC, 2445–2451. doi:10.1109/SMC53992.2023.10394237

[8] Arjun Kumar, Ling Chen, and Ming Zhang. 2025. Measuring AI's True Impact on Developer Productivity. arXiv:2509.19708 [cs.SE] https://arxiv.org/abs/2509.19708

[9] MITRE Corporation. 2024. *Common Weakness Enumeration (CWE) — A Community-Developed List of Software and Hardware Weakness Types.* Technical Report. MITRE Corporation / U.S. Department of Homeland Security Cybersecurity and Infrastructure Security Agency (CISA), Bedford, MA, USA. https://cwe.mitre.org/

[10] National Institute of Standards and Technology. 2023. *Artificial Intelligence Risk Management Framework (AI RMF 1.0).* Technical Report 100-1. National Institute of Standards and Technology, Gaithersburg, MD, USA. doi:10.6028/NIST.AI.100-1

[11] National Institute of Standards and Technology. 2024. *Artificial Intelligence Risk Management Framework: Generative Artificial Intelligence Profile.* Technical Report NIST AI 600-1. National Institute of Standards and Technology, Gaithersburg, MD, USA. doi:10.6028/NIST.AI.600-1

[12] Emmanouil Papagiannidis, Patrick Mikalef, and Kieran Conboy. 2025. Responsible artificial intelligence governance: A review and research framework. *The Journal of Strategic Information Systems* 34, 2 (2025), 101885. doi:10.1016/j.jsis.2024.101885

[13] Neil Perry, Megha Srivastava, Deepak Kumar, and Dan Boneh. 2023. Do Users Write More Insecure Code with AI Assistants?. In *2023 ACM SIGSAC Conference on Computer and Communications Security (CCS '23)*. AAssociation for Computing Machinery, New York, NY, USA, CCS '23, 2785–2799. doi:10.1145/3576915.3623157

[14] Thomas L. Saaty. 1980. *The Analytic Hierarchy Process: Planning, Priority Setting, Resource Allocation.* McGraw-Hill, New York, NY, USA.

[15] Thomas L. Saaty. 1990. How to Make a Decision: The Analytic Hierarchy Process. *European Journal of Operational Research* 48, 1 (1990), 9–26. doi:10.1016/0377-2217(90)90057-I

[16] Gustavo Sandoval, Hammond Pearce, Teo Nys, Ramesh Karri, Siddharth Garg, and Brendan Dolan-Gavitt. 2023. Lost at C: A User Study on the Security Implications of Large Language Model Code Assistants. In *32nd USENIX Security Symposium*. USENIX Security 23, USENIX Security 23, 2205–2222. https://www.usenix.org/conference/usenixsecurity23/presentation/sandoval

[17] Teresa Scantamburlo, Paolo Falcarin, Alberto Veneri, Alessandro Fabris, Chiara Gallese, Valentina Billa, Francesca Rotolo, and Federico Marcuzzi. 2024. Software Systems Compliance with the AI Act: Lessons Learned from an International Challenge. In *2nd International Workshop on Responsible AI Engineering (RAIE '24)*. Association for Computing Machinery, New York, NY, USA, RAIE '24, 44–51. doi:10.1145/3643691.3648589

[18] Andrew van der Stock, Brian Glas, Neil Smithline, and Torsten Gigler. 2021. *OWASP Top 10:2021 — The Ten Most Critical Web Application Security Risks.* Technical Report. OWASP Foundation, Wilmington, DE, USA. https://owasp.org/Top10/2021/

[19] Mark Velasquez and Patrick T. Hester. 2013. An Analysis of Multi-Criteria Decision Making Methods. *International Journal of Operations Research* 10, 2 (2013), 56–66. http://www.orstw.org.tw/ijor/vol10no2/ijor_vol10_no2_p56_p66.pdf

[20] Steve Wilson, Scott Clinton, Ads Dawson, and OWASP GenAI Security Project. 2025. *OWASP Top 10 for Large Language Model Applications 2025.* Technical Report. OWASP Foundation. https://genai.owasp.org/resource/owasp-top-10-for-llm-applications-2025/

[21] Claes Wohlin, Per Runeson, Martin Höst, Magnus C. Ohlsson, Björn Regnell, and Anders Wesslén. 2024. *Experimentation in Software Engineering (2nd ed).* Springer, Berlin, Germany. doi:10.1007/978-3-662-69306-3