Technical Sciences

UDC 004.42:004.5:004.89

Zhytomyrskyi Viktor

Senior Product Designer

HP (Hewlett-Packard)

ORCID: 0009-0008-7566-6284

DOI: https://doi.org/10.25313/2520-2057-2026-7-12110

MODEL FOR CONSISTENT AI INTERFACES BASED ON DESIGN-SYSTEM RULES

Summary. *The article proposes the Design-System AI Interface Consistency Model (DS-AICM), which maps task context, AI state, uncertainty, consequence and user control to design-system rules. Its novelty is explicit activation conditions, risk-proportional safeguards, artifact mapping and a governance feedback loop. A conceptual set expression, a matrix and three scenarios show internal coherence without software implementation or empirical effectiveness claims.*

Key words: *artificial intelligence, AI interface, design system, human-AI interaction, consistency, UX/UI design.*

Introduction. Artificial intelligence has become a visible interaction layer in digital products. Users no longer interact only with deterministic commands and predefined content: they formulate intentions, receive generated or ranked outputs, observe changing system states and decide whether an answer is sufficiently reliable for the next action. This shift changes the meaning of interface consistency. A visually uniform component library does not guarantee a

consistent experience when one AI feature explains its limits, another hides uncertainty, and a third offers no way to correct or reverse an outcome.

Design systems codify reusable components, interaction principles and organizational knowledge. They support consistent implementation and cross-functional collaboration through shared standards and terminology [1, p. 2, 17; 2, p. 1]. AI-enabled interaction nevertheless introduces variability that cannot be addressed by typography, color, layout and component behavior alone. The same component may require different copy, controls and safeguards depending on whether the system generates a draft, recommends an option, predicts an outcome or supports a consequential decision.

The research problem is therefore to convert dispersed human-AI interaction recommendations into a compact rule-selection model that can operate inside design-system practice. The model must remain usable for Product Design and UX/UI teams while explicitly connecting context, AI state, uncertainty, consequence, explanation, recovery and user control. The present article addresses this problem conceptually; it does not present source code, a software prototype or an empirical performance model.

Analysis of recent research and publications. Research on design systems shows that they function as more than repositories of UI components. They also coordinate documentation and collaboration, while maintenance requires stable knowledge to evolve with concrete product needs [1, p. 17, 21–22; 2, p. 1]. Studies of interface consistency further indicate that consistency is relational: comparable tasks and contexts should preserve recognizable choices, whereas different contexts may justify different behavior [3, p. 1]. This distinction is essential for AI products because the acceptable degree of automation and required information can change with task consequence.

Human-AI interaction research supplies a substantial set of design requirements. Amershi S. et al. group 18 guidelines by interaction phase – initially, during interaction, when wrong and over time – and include capability

framing, contextual information, correction and explanation [4, p. 3–4]. Yang Q. et al. identify uncertainty about AI capability and output complexity as distinctive sources of design difficulty [5, p. 1]. Liao Q. et al. show that explanation design should be driven by user- and context-dependent questions rather than model mechanics alone [6, p. 1, 3], while Eiband M. et al. translate transparency principles into a stage-based participatory process for product-specific decisions [7, p. 211–212].

Expectation management and control are equally important. Preparing users for imperfect AI can increase acceptance without implying that errors disappear [8, p. 1]. Onboarding research shows that people need information about capability, limitation, intended use and design objective before effective collaboration begins [9, p. 104:2–104:3]. Human-centered AI seeks high automation together with high human control [10, p. 495], and trust research frames the design goal as appropriate rather than maximal reliance [11, p. 50]. In decision support, explanations alone do not reliably eliminate overreliance or improve complementary team performance [12, p. 188:1–188:2; 13, p. 1–3]. Editable intermediate results and chained interactions can make complex AI behavior more inspectable and controllable [14, p. 1–2], while non-experts' prompt-design difficulties demonstrate the need for iterative support, error recognition and strategy formation [15, p. 1–3].

The literature therefore offers valuable principles, frameworks and empirical findings, but most contributions remain separate checklists or domain-specific results. Teams still need an operational answer to three questions: which rules activate in a given scenario, which design-system artifacts embody those rules, and how justified exceptions return to governance. DS-AICM addresses this gap by treating consistency as a conditional, traceable and maintainable rule-selection process.

Objectives of the article. The purpose of the article is to propose a conceptual model for ensuring consistency in AI-enabled interfaces by selecting

and applying design-system rules according to task context, AI state, uncertainty, consequence, irreversibility and required user control. The scientific novelty lies in combining literature-derived human-AI requirements with explicit activation conditions, proportional safeguards, design-system artifacts, traceability and a governance feedback loop. The practical value is a reusable structure for design specification, hand-off and review. In this article, risk is used only as an umbrella term covering consequence, irreversibility, time pressure, uncertainty-related exposure and required domain expertise; consequence remains the principal matrix variable.

To achieve this purpose, the study addresses three analytical tasks. First, it identifies the dimensions of consistency required for AI-enabled interfaces and explains why a component-oriented design system alone is insufficient. Second, it determines how AI-specific requirements can be represented as conditional design-system rules with explicit activation criteria. Third, it proposes a conceptual model for selecting, applying and reviewing these rules across typical product scenarios.

Research methods. The study uses conceptual synthesis, comparative analysis, classification and scenario-based analytical assessment. The literature search and page-level source verification were conducted from 1 June to 5 July 2026 using targeted title, abstract and keyword combinations in accessible bibliographic, publisher and full-text records. For potential replication in Scopus, the search strategy can be expressed through the following TITLE-ABS-KEY query families: TITLE-ABS-KEY ("design system*" AND (governance OR consistency OR component*)); TITLE-ABS-KEY (("human-AI interaction" OR "AI interface*") AND (guideline* OR uncertainty OR explainability OR "user control")); and TITLE-ABS-KEY (("AI-assisted decision-making" OR "generative AI") AND (reliance OR recovery OR onboarding OR transparency)). The strategy was iterative and designed for conceptual coverage rather than exhaustive systematic review; no PRISMA-style hit count is claimed.

Inclusion criteria were: peer-reviewed publication in English; direct relevance to design systems, interface consistency or observable human-AI interaction requirements; sufficient methodological or conceptual detail to derive a rule, activation condition or governance implication; and availability of verifiable bibliographic metadata and full text for page-level citation. Purely algorithmic explainability studies without an interface implication, non-peer-reviewed practitioner blogs, duplicate versions and papers used only for general AI background were excluded. The final analytical corpus contains 15 publications. Publisher and DOI records were used as the bibliographic source of truth. The corpus is therefore described as peer-reviewed literature rather than a Scopus-only sample.

The analysis followed five steps. First, recurring problems and requirements were extracted from the selected papers. Second, overlapping requirements were grouped into consistency dimensions and rule families. Third, existing approaches were compared against six criteria that expose the proposed model's incremental contribution. Fourth, the activation logic was formalized through a set expression and a consequence-uncertainty matrix. Fifth, three scenarios were used for non-empirical analytical verification. Two checks were applied: coverage, meaning that each proposed rule family was either linked to a literature-derived requirement or explicitly identified as an author-developed operational extension; and differentiation, meaning that scenarios with different consequence and uncertainty levels produced different mandatory rule sets. These checks demonstrate internal coherence, not external effectiveness.

Results

1. Theoretical foundations of AI interface consistency. For this study, an AI interface is an interaction layer through which a person specifies a goal, receives or supervises an AI-produced result and can influence subsequent system behavior. Consistency is defined as a stable relationship among user intent, the current AI state, output properties, interface feedback and available controls. The

definition deliberately shifts attention from identical appearance to predictable meaning and action.

Seven complementary dimensions are distinguished. Visual-structural consistency covers components, hierarchy, spacing and placement. Interaction-state consistency covers loading, generation, completion, interruption, error and recovery. Semantic consistency concerns stable terminology for AI actions and outputs. Epistemic consistency concerns the communication of uncertainty and limitation. Explanatory consistency concerns reasons, evidence and provenance. Agency consistency concerns editing, confirmation, cancellation, undo and escalation. Governance and accessibility consistency concerns documented ownership, exceptions, traceability and inclusive use.

The dimensions are interdependent. A "Regenerate" control is not consistent merely because it looks the same: its label, consequence, state transition and recovery behavior must also be predictable. Likewise, a confidence indicator may create inconsistency when related features use incompatible scales or fail to explain what the value means. Each reusable AI pattern should therefore specify four linked layers – component, content, behavior and control – while uncertainty and consequence determine which additional layers become mandatory. The rules presented in Table 1 are synthesized by the author from the reviewed literature and extended to the operational requirements of design-system governance and accessibility; they are not verbatim reproductions of individual publications.

Table 1

Dimensions of AI interface consistency and corresponding design-system rules

Dimension	Design objective	Representative rules and artifacts
Visual-structural	Preserve recognizable hierarchy and component use.	Shared tokens, layouts and AI-status components; generated content is distinguished without creating a separate visual language.

Dimension	Design objective	Representative rules and artifacts
Interaction-state	Make progress, change and failure predictable.	Named states for idle, queued, generating, partial result, completed, interrupted, failed and corrected; documented transitions and permitted actions.
Semantic and content	Keep terminology and instructions stable.	Consistent names for AI functions, outputs and controls; product-specific capability statements, prompt guidance and action-oriented messages.
Epistemic	Communicate uncertainty and limitations proportionally.	Limitations, confidence or verification needs appear when they affect action; false precision is avoided; facts, suggestions and hypotheses are distinguished.
Explanatory and provenance	Support understanding of relevant reasons and sources.	"Why this?", evidence, data origin or transformation history are provided according to task consequence through progressive disclosure.
Agency and recovery	Maintain meaningful user control.	Edit, retry, stop, undo, compare, reject and escalation controls are defined where applicable; consequential execution requires confirmation.
Governance and accessibility	Keep rules maintainable, auditable and inclusive.	Rule ownership, exception rationale, review dates and accessibility behavior for dynamic content, status announcements and keyboard interaction are documented.

Source: developed by the author through conceptual synthesis based on [1, p. 2, 17, 21–22; 3, p. 1; 4, p. 3–4; 6, p. 1, 3; 10, p. 495; 11, p. 50, 73–75; 14, p. 1–2] and extended to the operational requirements of design-system governance and accessibility

A separate comparison is necessary because scientific novelty cannot be established by listing useful principles alone. Table 2 presents the author's comparative synthesis of existing approaches and the proposed DS-AICM. The criteria are analytical dimensions selected in this study: evidence from prior work supports the first two approach columns, while the DS-AICM column reports the contribution proposed here.

Table 2

Existing approaches versus DS-AICM

Criterion	Conventional design systems	Human-AI guidelines and frameworks	Proposed DS-AICM contribution
Activation conditions	Usually implicit in component documentation or product convention.	Guidelines identify desirable behavior, but selection is often left to practitioner judgment.	The proposed model gives each rule an observable trigger tied to context, AI state, consequence, uncertainty or personalization.
Risk proportionality	Not normally central to component selection.	HCAI and trust frameworks establish the importance of control and calibrated reliance.	Within DS-AICM, safeguards increase with consequence, uncertainty and irreversibility instead of relying on one uniform disclaimer.
Design-system artifacts	Components, tokens, patterns, documentation and code.	Principles, checklists, studies and domain examples.	The proposed model maps activated requirements to components, content, behavior, controls, state definitions and ownership fields.
Governance loop	Ownership and maintenance may exist, but AI-specific exceptions are not necessarily structured.	Most approaches focus on design guidance or evaluation rather than repository evolution.	DS-AICM records context, owner, evidence and review date for exceptions; recurrent exceptions may become governed patterns.
Recovery rules	Generic error and undo patterns may be available.	Correction, dismissal and control are recommended, but not always linked to state and consequence.	When activated, DS-AICM specifies stop, retry, edit, undo, compare, reject or escalation behavior.
Traceability	Traceability is often limited to component usage and implementation.	The rationale for a selected guideline may remain external to the design specification.	The proposed model traces each safeguard from scenario condition to rule, design-system artifact, implementation behavior and review decision.

Source: author's comparative synthesis; descriptions of conventional design systems and human-AI approaches are based on [1, p. 2, 17, 21–22; 2, p. 1; 4, p. 3–4; 6, p. 1, 3; 10, p.

495; 11, p. 50, 73–75; 14, p. 1–2], while the DS-AICM column presents the contribution proposed in this study

2. Design-system rules for AI interfaces. Within DS-AICM, the author defines a rule not as a general instruction such as "be transparent", but as a conditional statement connecting an observable design situation with a required interface response. A usable record contains at least five fields: activation condition, required pattern, prohibited pattern, rationale and evidence or ownership. For example, when an AI result can materially influence a consequential decision, the proposed model requires the interface to expose the relevant basis of the recommendation, enable human review and prohibit silent automatic execution.

Capability framing and state visibility. Within DS-AICM, capability framing and state visibility are represented as conditional rule records. The reviewed literature supports disclosure of capability, limitation and context before use [4, p. 3; 9, p. 104:2–104:3]. The proposed model extends this basis by requiring a product-specific state vocabulary – queued, processing, partial, completed, interrupted and failed – and by defining the controls available in each state.

Uncertainty, explanation and provenance. The proposed model does not require a numerical confidence value in every interface. It requires an uncertainty representation that is understandable and actionable in the given context. The author proposes that a low-consequence creative suggestion may need concise limitation text and an easy retry, whereas decision support may require calibrated uncertainty, evidence, provenance and an explicit verification step. Within DS-AICM, explanation depth is proposed to increase with task consequence and novelty, while technical detail is disclosed progressively. This rule is derived through conceptual synthesis: prior work supports question- and context-dependent explanations, expectation management and appropriate reliance, while

also showing that explanations alone can encourage or fail to prevent inappropriate reliance [6, p. 1, 3; 8, p. 1; 11, p. 50, 73–75; 12, p. 188:1–188:2; 13, p. 1–3].

Agency, recovery and governance. Within DS-AICM, the author proposes that meaningful control include editing inputs, interrupting a long operation, rejecting an output, restoring a prior state, requesting alternatives and escalating to a human or authoritative process. Feedback controls must state whether they alter only the current result, contribute to product improvement or personalize future behavior. The governance fields – owner, exception rationale, evidence and review date – are operational extensions introduced by the model, informed by design-system evolution and human-AI correction and control research [1, p. 21–22; 4, p. 3; 14, p. 1–2; 15, p. 1–3].

3. Proposed Design-System AI Interface Consistency Model. The proposed Design-System AI Interface Consistency Model (DS-AICM) consists of six linked stages and a governance feedback loop. It begins with the user's situation rather than the existing component inventory, then narrows the design problem until an auditable set of rules and artifacts can be specified. Figure 1 presents the overall sequence.

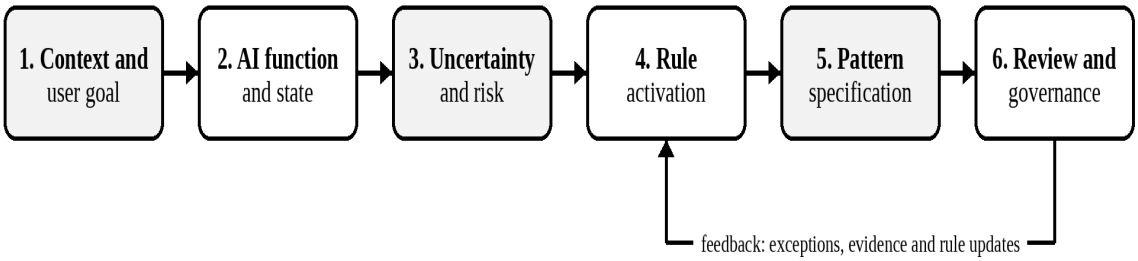

Fig. 1. Design-System AI Interface Consistency Model (DS-AICM)

Source: developed by the author through conceptual synthesis based on [1, p. 21–22; 4, p. 3–4; 14, p. 1–2]

Stage 1. Context framing. The model records the user role, task, expected outcome, conditions of use, decision consequence and acceptable delay.

Stage 2. AI function and state mapping. The feature is classified as generation, transformation, recommendation, classification, prediction or decision support. The relevant normal, interrupted and failure states are then identified. Together, the first two stages prevent a pattern from being selected solely because an existing component is available.

Stage 3. Assessment of uncertainty, consequence and irreversibility. The model assesses output uncertainty, the potential consequence of error, reversibility, time pressure and the need for domain expertise.

Stage 4. Rule activation. Always-on requirements are combined with additional safeguards triggered by the conditions identified at the previous stage. For conceptual use, low, medium and high ordinal levels are sufficient; the model does not imply a statistically calibrated risk score.

Stage 5. Pattern specification. The activated rule set is translated into design-system artifacts: components, content, behavior and controls. These artifacts include state definitions, interface copy, explanation depth, confirmation, recovery and provenance requirements.

Stage 6. Review and governance. The proposed solution is checked for cross-product consistency, accessibility and safety. A justified deviation is documented together with its context and owner, while recurring deviations return to the repository as candidates for a new governed pattern. This creates the governance feedback loop.

The activation logic can be expressed as a conceptual set union:

$$R_{\text{act}}(x) = R_{\text{base}} \cup R_{\text{risk}}(c, v) \cup R_{\text{uncertainty}}(u) \cup R_{\text{personalization}}(p).$$

Here, x denotes the design scenario; c is consequence; v is irreversibility; u is output or capability uncertainty; and p denotes personalization or feedback that changes future behavior. The expression is a conceptual design-decision representation rather than an executable software algorithm. Every scenario

receives the base set, while additional rule families become mandatory when their activation conditions are present.

Table 3

Consequence-uncertainty matrix for rule activation

Consequence	Uncertainty	Required level	Proposed mandatory rule set
Low	Low	Base	AI-role disclosure; stable naming; state visibility; accessibility; ordinary edit or cancel behavior.
Low	High	Base + uncertainty	Base rules plus limitation text, verification cue, retry or alternative, and explicit recovery from poor output.
High	Low	Base + consequence	Base rules plus provenance, review, confirmation, auditability and prohibition of silent automatic execution.
High	High	Full safeguards	All rule families: uncertainty, explanation, provenance, confirmation, human review, recovery, escalation and audit history.

Source: developed by the author through conceptual synthesis based on [4, p. 3–4; 10, p. 495; 11, p. 50, 73–75; 12, p. 188:1–188:3; 13, p. 1–3]

The matrix is proposed in this study and is not reproduced from an existing framework. Within DS-AICM, the author assumes a conservative upper-right condition and a lightweight lower-left condition. This prevents two common design errors: applying high-friction safeguards to every creative feature, and treating a generic disclaimer as sufficient for a high-consequence decision. When conditions conflict, the rule associated with the greater consequence or lower reversibility prevails. Personalization activates disclosure and consent requirements independently of the matrix because it changes future system behavior.

Table 4

Scenario-based application of DS-AICM

Scenario and conditions	Rule mapping proposed in DS-AICM	Conceptual consistency outcome
Generative writing assistant. Low consequence; medium uncertainty; reversible draft.	Base rules plus uncertainty and recovery: show generation state and limitations; mark generated content; provide edit, stop, regenerate, compare and undo; keep labels stable across entry points.	The output is understood as a draft. Users can revise it and encounter the same recovery behavior throughout the product.
Recommendation module. Medium consequence; preference-dependent output; multiple valid alternatives.	Explain the main recommendation factors; provide "Why this?", alternatives and preference controls; disclose relevant data use; allow rejection and correction.	Recommendations remain adjustable support rather than a single objective answer, and users retain control over criteria.
High-consequence decision-support dashboard. High uncertainty; expert review required.	Activate full safeguards: evidence and provenance, calibrated uncertainty, mandatory review and confirmation, no silent execution, audit history, recovery and escalation.	The interface supports appropriate reliance, preserves accountability and applies equivalent safeguards to comparable decisions.

Source: scenarios and rule mappings developed by the author; underlying requirements synthesized from [4, p. 3–4; 8, p. 1; 9, p. 104:2–104:3; 10, p. 495; 11, p. 50, 73–75; 12, p. 188:1–188:3; 13, p. 1–3; 14, p. 1–2; 15, p. 1–3]

Discussion. The findings correspond to the three analytical tasks defined at the beginning of the study. First, the analysis shows that AI-interface consistency includes visual, behavioral, semantic, epistemic, explanatory, agency and governance dimensions. A component-oriented system alone is insufficient because the meaning and required safeguard of a component depend on the AI state and the consequence of use. Second, the study represents each requirement as a conditional rule with an activation trigger and linked design-system artifacts. Third, the proposed six-stage structure, conceptual activation set, consequence-

uncertainty matrix and scenario walkthroughs provide a structured way to select, apply and review those rules in typical product situations.

The comparison in Table 2 clarifies the novelty claim. DS-AICM does not claim originality for human control, explainability or uncertainty communication. It integrates those established concerns with design-system governance through explicit activation conditions, risk-proportional rule selection, scenario-to-rule-to-artifact mapping, implementation traceability and a governance feedback loop. A designer can justify why a confirmation or recovery control appears, while an engineer can identify the AI state, artifact and behavior to implement.

Risk proportionality is the second contribution. As defined above, risk is an umbrella category, while consequence remains the principal matrix variable. The model neither classifies every AI feature as high risk nor reduces responsible design to a universal disclaimer. Requirements increase when uncertainty, consequence or irreversibility increases. At the same time, the governance feedback loop preserves context-sensitive design. Design systems must evolve with concrete product needs [1, p. 21–22]; DS-AICM supports that evolution by recording justified exceptions and returning recurrent patterns to the governed rule set.

The scenario analysis is an internal analytical check only. It shows that the model differentiates low- and high-consequence use cases, but it does not prove improved usability, safety or development efficiency. External validation would require expert review, application to a real product and user or audit data. This limitation is preferable to presenting uncollected expert opinions or simulated performance as empirical evidence.

Conclusions and prospects for further research. The article proposes DS-AICM as a conceptual model for ensuring consistency in AI-enabled interfaces through design-system rules. Consistency is defined as a stable relationship among user intent, AI state, output properties, feedback and user control. The model organizes the problem into seven dimensions and six linked

stages: context framing, AI-function and state mapping, assessment of uncertainty, consequence and irreversibility, rule activation, pattern specification, and review with a governance feedback loop.

The scientific contribution is the conditional connection of human-AI interaction knowledge with operational design-system artifacts. The comparison table identifies the incremental value over existing approaches; the conceptual set expression and decision matrix formalize rule activation; and the scenarios illustrate proportionate requirements for a writing assistant, a recommendation module and a high-consequence dashboard. The practical result is a concise method that Product Design and UX/UI teams can use before implementation.

The study remains conceptual. It has not been implemented as software and has not been evaluated in a user study or external expert panel. The literature selection was targeted rather than systematic, and the scenarios do not cover every AI modality or domain. Further research should include structured assessment by three to five UX/UI and engineering experts, checklist evaluation of existing AI interfaces, application to a real product, usability testing, refinement of activation thresholds and measurable compliance criteria for design-system audits.

References

1. Lamine Y., Cheng J. Understanding and Supporting the Design Systems Practice. *Empirical Software Engineering.* 2022. Vol. 27, No. 6. Art. 146. 26 p. DOI: https://doi.org/10.1007/s10664-022-10181-y

2. Yew J. C. L., Convertino G., Hamilton A., Churchill E. F. Design Systems: A Community Case Study. *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems.* New York : Association for Computing Machinery, 2020. P. 1–8. DOI: https://doi.org/10.1145/3334480.3375204

3. Burny N., Vanderdonckt J. (Semi-)Automatic Computation of User Interface Consistency. *Companion of the 2022 ACM SIGCHI Symposium on*

Engineering Interactive Computing Systems. New York : Association for Computing Machinery, 2022. P. 1–9. DOI: https://doi.org/10.1145/3531706.3536448

4. Amershi S., Weld D. S., Vorvoreanu M., Fourney A., Nushi B., Collisson P., Suh J., Iqbal S., Bennett P. N., Inkpen K., Teevan J., Kikin-Gil R., Horvitz E. Guidelines for Human-AI Interaction. *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems.* New York : Association for Computing Machinery, 2019. P. 1–13. DOI: https://doi.org/10.1145/3290605.3300233

5. Yang Q., Steinfeld A., Rosé C. P., Zimmerman J. Re-examining Whether, Why, and How Human-AI Interaction Is Uniquely Difficult to Design. *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems.* New York : Association for Computing Machinery, 2020. P. 1–17. DOI: https://doi.org/10.1145/3313831.3376301

6. Liao Q. V., Gruen D., Miller S. Questioning the AI: Informing Design Practices for Explainable AI User Experiences. *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems.* New York : Association for Computing Machinery, 2020. P. 1–15. DOI: https://doi.org/10.1145/3313831.3376590

7. Eiband M., Schneider H., Bilandzic M., Fazekas-Con J., Haug M., Hussmann H. Bringing Transparency Design into Practice. *Proceedings of the 23rd International Conference on Intelligent User Interfaces.* New York : Association for Computing Machinery, 2018. P. 211–223. DOI: https://doi.org/10.1145/3172944.3172961

8. Kocielnik R., Amershi S., Bennett P. N. Will You Accept an Imperfect AI? Exploring Designs for Adjusting End-user Expectations of AI Systems. *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems.* New York : Association for Computing Machinery, 2019. P. 1–14. DOI: https://doi.org/10.1145/3290605.3300641

9. Cai C. J., Winter S., Steiner D., Wilcox L., Terry M. "Hello AI": Uncovering the Onboarding Needs of Medical Practitioners for Human-AI Collaborative Decision-Making. *Proceedings of the ACM on Human-Computer Interaction.* 2019. Vol. 3, No. CSCW. Art. 104. P. 104:1–104:24. DOI: https://doi.org/10.1145/3359206

10. Shneiderman B. Human-Centered Artificial Intelligence: Reliable, Safe & Trustworthy. *International Journal of Human-Computer Interaction.* 2020. Vol. 36, No. 6. P. 495–504. DOI: https://doi.org/10.1080/10447318.2020.1741118

11. Lee J. D., See K. A. Trust in Automation: Designing for Appropriate Reliance. *Human Factors.* 2004. Vol. 46, No. 1. P. 50–80. DOI: https://doi.org/10.1518/hfes.46.1.50_30392

12. Buçinca Z., Malaya M. B., Gajos K. Z. To Trust or to Think: Cognitive Forcing Functions Can Reduce Overreliance on AI in AI-Assisted Decision-Making. *Proceedings of the ACM on Human-Computer Interaction.* 2021. Vol. 5, No. CSCW1. Art. 188. P. 188:1–188:21. DOI: https://doi.org/10.1145/3449287

13. Bansal G., Wu T., Zhou J., Fok R., Nushi B., Kamar E., Ribeiro M. T., Weld D. S. Does the Whole Exceed Its Parts? The Effect of AI Explanations on Complementary Team Performance. *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems.* New York : Association for Computing Machinery, 2021. P. 1–16. DOI: https://doi.org/10.1145/3411764.3445717

14. Wu T., Terry M., Cai C. J. AI Chains: Transparent and Controllable Human-AI Interaction by Chaining Large Language Model Prompts. *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems.* New York : Association for Computing Machinery, 2022. P. 1–22. DOI: https://doi.org/10.1145/3491102.3517582

15. Zamfirescu-Pereira J. D., Wong R. Y., Hartmann B., Yang Q. Why Johnny Can't Prompt: How Non-AI Experts Try (and Fail) to Design LLM Prompts. *Proceedings of the 2023 CHI Conference on Human Factors in*

Computing Systems. New York : Association for Computing Machinery, 2023. P. 1–21. DOI: https://doi.org/10.1145/3544548.3581388