engineering proceedings

MDPI

Proceeding Paper

A Results-Based Evaluation Framework for Smart Learning Environments: Use Cases, User Testing, and Validation Design of the OSEME Platform †

Konstantinos Chrysanthos Sofianos [1,*], Michael Stefanidakis [1], Manolis Maragoudakis [1], Alexios Kaponis [1] and Linas Bukauskas [2]

[1] Department of Informatics, Ionian University, Plateia Tsirigoti 7, 49100 Corfu, Greece; mistral@ionio.gr (M.S.); mmarag@ionio.gr (M.M.); alexios.kaponis@ionio.gr (A.K.)
[2] Institute of Computer Science, Vilnius University, LT-08303 Vilnius, Lithuania; linas.bukauskas@mif.vu.lt
* Correspondence: c20sofi@ionio.gr; Tel.: +30-6945674992
† Presented at the 2nd International Conference on Engineering, Technology, and Applied Science Innovations (ICETASI 2026), Zakynthos, Greece, 24–26 September 2026.

Abstract

Smart learning environments (SLEs) extend conventional Learning Management Systems by adapting content delivery to the cognitive profile, prior knowledge, and behavioral patterns of individual learners. Although the technical realization of such systems is increasingly well documented, their empirical evaluation remains comparatively underdeveloped, and assessment efforts frequently rely on subjective impressions rather than reproducible measurement. This paper addresses that gap by proposing a results-based, tri-pivotal evaluation framework that unifies pedagogical, technological-usability, and learning analytics dimensions into a single, database-grounded measurement model. The framework is instantiated on OSEME, an ontology-driven adaptive platform that integrates a Felder–Silverman learning style profiler, a domain-knowledge assessment in Music History, and real-time analytics visualization within a modular PHP/MySQL architecture. To operationalize the framework, we specify formal use cases, a structured user testing protocol with task-level success criteria, and a set of case-study designs, including a controlled A/B retention experiment and a System Usability Scale (SUS) assessment. Use cases, user tests, and case studies are presented in tabular form to support replication. The contribution is methodological: a transferable evaluation blueprint that converts platform telemetry into actionable evidence of pedagogical effectiveness, technical performance, and usability.

Keywords: smart learning environments; adaptive learning; Felder–Silverman model; learning analytics; usability evaluation; System Usability Scale; user testing; ontology-based personalization

Check for updates

Academic Editor: Konstantinos Poirazidis

Published: 9 November 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

The convergence of software engineering, artificial intelligence, and instructional design has accelerated the emergence of Smart Learning Environments (SLEs). In contrast to traditional Learning Management Systems (LMSs), which deliver static and uniform content irrespective of the recipient, SLEs adapt dynamically to individual learner profiles, cognitive preferences, and prior knowledge, engagement, and retention [1,2]. This capacity for adaptation is particularly consequential in heterogeneous domains such as music education and informatics, where learners enter with markedly divergent backgrounds and where a one-size-fits-all delivery model is liable to depress engagement and retention.

The OSEME platform represents an advanced SLE that reconciles heterogeneous data structures by coupling a dedicated domain ontology with a Massive Open Online Course (MOOC) layout integrated into Moodle [3]. Architecturally, OSEME adopts a decoupled, component-based PHP structure (index1.php, menou1.php, kentro.php, footer.php) backed by a MySQL relational layer, and exposes the following core capabilities:

- Dynamic user authentication delivered through accessible, interactive modal interfaces;
- Cognitive profiling by means of a 44-item Learning Style Questionnaire grounded in the Felder–Silverman Learning Style Model (FSLSM);
- Objective domain-knowledge assessment via a 20-item Music History examination;
- Real-time diagnostic visualization supported by asynchronous data streaming and client-side charting (Chart.js).

While the deployment of such systems is comparatively well understood, their empirical assessment requires an evaluation framework that bridges software-engineering metrics and pedagogical outcomes. This paper proposes such a framework and demonstrates its application through formally specified use cases, a structured user testing protocol, and reproducible case-study designs. The central argument is that the evaluation of an SLE should be anchored in platform telemetry rather than impressionistic judgment, so that claims of adaptivity and effectiveness become testable propositions.

2. Literature Review and Theoretical Background

2.1. Adaptive Learning and the Felder–Silverman Model

Personalization in e-learning presupposes an accurate and computationally tractable taxonomy of cognitive preferences. The Felder–Silverman Learning Style Model (FSLSM) [4] characterizes learners along four largely independent dimensions:

- Active versus Reflective—processing information through experimentation and discussion as opposed to introspective examination.
- Sensing versus Intuitive—orienting toward concrete facts and procedures as opposed to abstract concepts and relationships.
- Visual versus Verbal—retaining information more effectively through diagrams and video as opposed to textual or spoken explanation.
- Sequential versus Global—advancing in linear, incremental steps as opposed to holistic, macro-level synthesis.

These dimensions can be quantified using validated instruments, such as the 44-item Index of Learning Styles [5], so that a software backend can deterministically alter content-delivery pathways, mapping a numerical profile vector onto a corresponding presentation strategy [6–11].

2.2. Ontologies in Intelligent Tutoring Systems

Ontologies supply the semantic backbone required for machine-readable personalization. An ontology formalizes the relationships between learner profiles, pedagogical strategies, and course components, so that the metadata describing a learning object (whether a video resource or a programming exercise) can be aligned with the real-time cognitive state of the authenticated user. This explicit declarative representation is what sets apart intelligent tutoring systems from rule-bound LMS configurations, and forms the interpretive layer through which adaptive decisions can be executed and audited [12–14].

2.3. Evaluation Paradigms for E-Learning Infrastructures

Evaluating an SLE differs fundamentally from conventional software testing. Whereas traditional verification concentrates on defect detection, load capacity, and query optimiza-

https://doi.org/10.3390/engproc2026167020

tion, an adaptive educational system must additionally be assessed in terms of learner behavior observed over time.

The literature commonly applies the System Usability Scale (SUS) for interface diagnostics and Learning Analytics Dashboards (LADs) to transform raw log data into interpretable cognitive metrics. A rigorous evaluation therefore integrates instrumented software metrics with longitudinal behavioral evidence, and it is this integration that the present framework seeks to formalize [15–22].

3. The OSEME Platform Architecture

To contextualize the proposed evaluation framework, the software architecture of OSEME is outlined below. The system adopts a modular PHP design that enforces a clean separation of concerns: the Section Database Relational Model.

- Presentation layer components rendered by index1.php, kentro.php, and footer.php;
- Session and middleware control governed by menou1.php, which maintains the active $_SESSION global state and enforces database-routing constraints;
- Data-access logic triggered through submit_quiz1.php and submit_quiz2.php, which marshal raw request parameters into persistent storage via PDO.

The database relational model is presented in the following section. Figure 1 illustrates the component-level architecture and request flow of the platform.

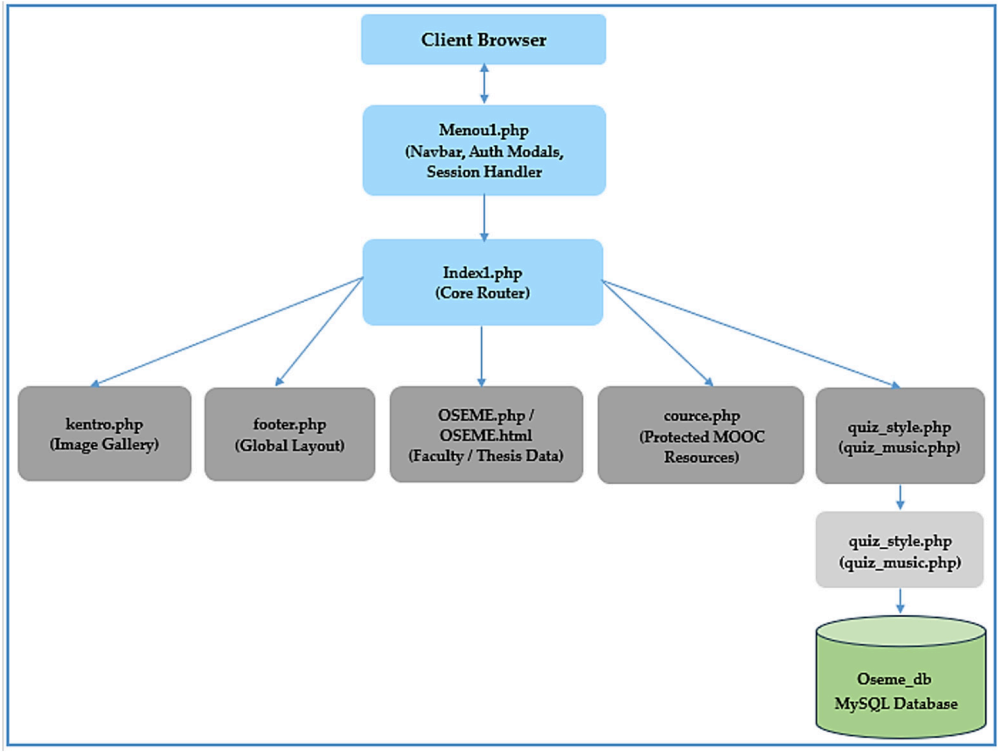


Figure 1. Component-level architecture and request flow of the OSEME platform.

Database Relational Model

The structural foundation of the results-based framework is the telemetry captured within the relational schema. Three tables are central to evaluation:

- users, which stores credential-verification tokens, unique system hashes, and creation timestamps;

- quiz_style_results, which maintains the integer vectors (score_ar, score_si, score_vv, score_sg) generated from the 44-item diagnostic instrument;
- quiz_music_results, which records the raw performance metrics (correct_answers, wrong_answers, percentage) derived from the domain-specific examination.

The platform was implemented using PHP (version 8.1, The PHP Group, San Francisco, CA, USA), MySQL (version 8.0, Oracle Corporation, Austin, TX, USA), and Chart.js (version 4.4, Chart.js Contributors, open-source). The learning management system was based on Moodle (version 4.3, Moodle Pty Ltd., Perth, Australia).

4. The Proposed Evaluation Framework

The principal contribution of this paper is a tri-pivotal evaluation model that replaces qualitative assumptions with quantitative, database-driven measurements. The framework decomposes evaluation into three complementary dimensions, each associated with distinct instruments and data sources. Its emphasis lies on the quality and provenance of the evidence gathered—what is observed, under what conditions, and against which established instrument—rather than on any particular numerical threshold. Where quantities appear in later sections, they serve the constructs rather than defining them. Figure 2 illustrates the tri-pivotal structure of the proposed evaluation framework, which decomposes evaluation into three complementary dimensions.

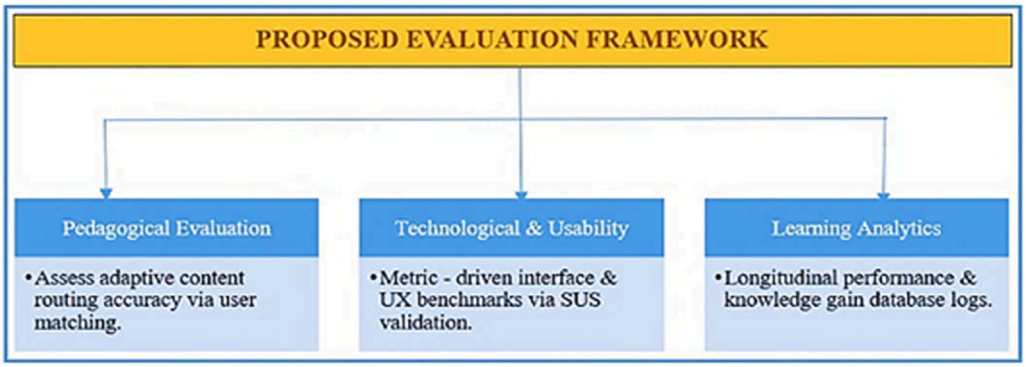


Figure 2. The tri-pivotal structure of the proposed evaluation framework.

4.1. Pedagogical Evaluation Dimension

This dimension measures how accurately the system maps content presentation onto identified learning styles. It verifies whether the structural component logic correctly reads the user-session array and serves a specialized user-interface configuration consistent with the learner's cognitive profile. Operationally, it tests the fidelity of the path from diagnostic vector to rendered layout.

4.2. Technological and Usability Dimension

This dimension applies standardized software-performance testing and user-experience instrumentation. It captures server response latency, database query-execution time under concurrent access, and perceived interface clarity during interactive states such as multi-tier dropdown navigation and modal forms. It therefore unites objective performance benchmarks with subjective usability evidence.

4.3. Learning Analytics Dimension

This dimension tracks learner performance over time. Drawing on persisted database records, it computes knowledge-gain parameters and characterizes the trajectory by which

a learner transitions from a baseline state to demonstrable proficiency. It is the temporal complement to the otherwise cross-sectional pedagogical and usability measures.

5. Use Cases for Evaluation

To articulate how the framework operates in practice, we specify use cases that delineate the intended behavior of an adaptive learning environment and identify the classes of evidence required to evaluate it. Each use case is expressed in terms of a primary actor, a set of preconditions, a flow of events, and the evaluation constructs that the flow renders observable. The intent here is conceptual rather than statistical: the use cases establish what ought to be measured and why a given observation counts as evidence of adaptation, while the choice of instruments and acceptance thresholds is deferred to the empirical instantiation set out in Sections 6 and 7. Because the framework is proposed rather than reported, the metrics named below are treated as constructs awaiting operationalization, and their concrete indicators are drawn from the validated instruments reviewed in Section 5.3.

5.1. Use Case 1: Adaptive Flow Profiling and Content Allocation

Primary Actor: Authenticated learner.

Preconditions: The user has completed registration through the entry modal, has initialized a unique session identifier, and has accessed quiz_style.php.

Flow of Events:

1. The user supplies answers to the 44 binary-choice items;
2. submit_quiz1.php parses the POST payload array;
3. The backend computes dimensional delta values in the interval $[-11, +11]$;
4. Results are written to quiz_style_results and visual arrays are generated through Chart.js;
5. The user requests course.php?id=python;
6. The platform evaluates the stored profile and reorders the displayed resources—placing video links above text blocks for visual learners, for example.

5.2. Use Case 2: Assessment, Score Tracking, and Graphic Visualization

Primary Actor: Registered student.

Preconditions: An active session is present and access has been granted to the Music History suite (quiz_music.php).

Flow of Events:

1. The student progresses through the paginated assessment view (ten questions per screen);
2. On submission, submit_quiz2.php cross-references the responses against the authoritative server-side answer key;
3. The script writes the raw metrics (correct_answers, wrong_answers, percentage) to quiz_music_results;
4. A doughnut chart is rendered dynamically to visualize the success ratio;

Evaluation Metrics:

- Data Point 1. Mean submission-completion velocity—seconds elapsed from form initialization to database commit;
- Data Point 2. Accuracy-rate distribution across the user cohort.

5.3. Established Metrics for Comparable Systems and Their Applicability to the Proposed Framework

A proposed framework acquires credibility not from the novelty of its indicators but from their lineage. It is therefore worth stating explicitly that each of the constructs

introduced above corresponds to a measure that has already been validated in the evaluation of comparable adaptive and intelligent learning systems, and that the framework advanced here is best understood as an organised recomposition of that existing instrumentation rather than as an attempt to displace it.

For the pedagogical dimension, the dominant tradition is the controlled comparison of adaptive against non-adaptive delivery, with post-test attainment as the outcome variable. This design underlies the majority of the primary studies aggregated in recent meta-analyses of AI-enabled adaptive learning, and the accompanying convention of reporting a standardized effect size alongside significance testing [23,24] exists precisely because attainment differences in educational settings are frequently significant yet substantively small. The framework adopts this pairing without modification, on the reasoning that a measure already used to adjudicate between adaptive systems can adjudicate the present one on equal terms.

For the technological and usability dimension, the System Usability Scale has functioned for three decades as the de facto common currency of interface evaluation. Its enduring utility derives less from the instrument itself than from the accumulated normative base against which any individual score can be positioned: successive empirical calibrations have established interpretation bands and item-level benchmarks that convert an otherwise arbitrary composite into a comparative statement [16,17,25]. Applying the SUS to the platform under study therefore yields a result that is meaningful in relation to the wider population of evaluated systems rather than only in relation to itself. The same rationale extends to the task-based measures that accompany it—completion rate, time-on-task, and error frequency—which constitute the standard behavioural triad of usability engineering [26] and whose small-sample properties have been characterised in the literature on defect discovery [22].

For the learning-analytics dimension, the relevant precedent lies in the educational data mining and analytics tradition, where interaction logs are treated as legitimate evidence of learning processes rather than as mere system telemetry. Reviews of learning-analytics dashboards have, however, drawn attention to a recurring weakness: visualisations are frequently evaluated on their perceived usefulness rather than on any demonstrated effect on attainment or engagement. The framework responds to this critique by requiring that the analytics dimension be assessed through change in learner outcomes over time, a design that follows the repeated measures logic long employed in the intelligent tutoring literature [2,27].

Two implications follow. The first is that the proposed framework is evaluable by means already accepted in the field; no bespoke instrument must be validated before the framework can be applied, and results obtained under it are directly commensurable with those reported for other adaptive platforms. The second is that the framework's contribution lies in integration rather than invention. Existing evaluations tend to be dimensionally partial—strong on usability but silent on attainment, or attentive to attainment but indifferent to whether the interface obstructed it—and the tri-pivotal structure proposed here binds the three established measurement traditions into a single reporting scheme, so that a claim of adaptive benefit cannot be sustained on one dimension while the others remain unexamined.

The thresholds cited in Sections 6 and 7 should be read in this light. They are conventional values inherited from the sources above rather than results, and they are stated in advance for the familiar methodological reason that a criterion fixed after the fact is not a criterion at all. Investigators instantiating the framework in a different disciplinary or institutional setting would be expected to revise them in accordance with the norms of that setting, without any consequent alteration to the structure of the framework itself.

https://doi.org/10.3390/engproc2026167020

6. User Testing Protocol

This section specifies the planned user testing protocol and acceptance criteria for the platform. Participant recruitment, task completion, and outcome reporting are described as part of the evaluation design rather than as completed empirical results in this manuscript. Following established practice in usability engineering, testing is task-based: each session presents a discrete, observable task with a predefined success criterion, and behavioral data are recorded directly rather than inferred from self-reports. A moderated think-aloud procedure is employed so that comprehension breakdowns can be located precisely within the interface.

Participant recruitment follows the small-sample convention established in the usability literature, whereby a cohort of approximately five to eight representative users uncovers the majority of severe interface defects [23–25]. Each participant completes the battery of tests summarized in Table 1; sessions are instrumented to capture the task-completion rate, time-on-task, error count, and, on conclusion, a System Usability Scale rating. Quantitative thresholds are stated in advance so that pass or fail is determined objectively rather than retrospectively.

Table 1. User testing protocol: task-level, scenarios, instrumented metrics, and predefined success criteria.

Test ID	Objective	Task Scenario	Participant Profile	Metrics Captured	Success Criterion
UT-01	Account creation and authentication	Register through the entry modal and log in using the new credentials	First-time user, no prior exposure	Task-completion rate; time-on-task; form abandonment	100% completion; abandonment < 5%
UT-02	Cognitive profiling	Complete the 44-item FSLSM questionnaire and reach the result screen	Registered learner	Completion rate; time-on-task; mid-task drop-off	≥90% completion; no dimension left unanswered
UT-03	Domain assessment	Complete the 20-item Music History examination across paginated screens	Registered learner	Completion velocity; navigation errors	≥95% completion; <2 navigation errors
UT-04	Adaptive rendering	Open a course resource and confirm the layout reflects the stored profile	Profiled learner (visual and verbal variants)	Profile-to-layout match rate; subjective relevance	≥90% correct layout; mean relevance ≥ 4/5
UT-05	Analytics visualization	Locate and interpret the score doughnut and style charts	Learner with completed quizzes	Chart render time; interpretation accuracy	Render < 200 ms; ≥80% correct interpretation
UT-06	Navigation and wayfinding	Reach a target MOOC resource via the multi-tier dropdown menu	Mixed-experience participants	Time-on-task; misclick count; success rate	≥90% success; ≤1 misclick on average

Defects surfaced during testing are triaged by severity, with severe and moderate issues resolved before the case-study phase. Because every test is bound to a measurable criterion, the protocol yields a reproducible record that can be re-executed after each release to confirm that usability has not regressed.

7. Case Studies

To demonstrate the empirical utility of the proposed framework, we specify three case studies designed for execution within a live academic environment. Case Study A now emphasizes pre-test covariate adjustment, effect sizes, and attrition reporting; Case Study B clarifies SUS interpretation and small-sample handling; Case Study C clarifies normalized gain, missing-data handling, and repeated-measures logic. Together they exercise all three evaluation dimensions: a controlled experiment for the pedagogical dimension, a standardized usability assessment for the technological-usability dimension, and a longitudinal analysis for the learning analytics dimension. The studies are summarized in Table 2 and elaborated thereafter.

Table 2. Case studies for framework validation: objectives, designs, sample primary metrics, and acceptance criteria.

Case Study	Objective	Design and Method	Sample	Primary Metric	Acceptance Criterion
A—A/B retention test	Show that adaptive delivery improves retention over static delivery	Randomized two-group experiment; two-week learning phase; independent two-sample *t*-test	100 students split into G_C and G_E (50 each)	Mean Music History score difference (μ_E − μ_C)	Reject H_0 at $p < 0.05$; Cohen's d ≥ 0.5
B—SUS usability profiling	Assess layout efficiency, navigation, and satisfaction	Post-use 10-item SUS questionnaire on a 5-point Likert scale; scores scaled × 2.5	All active participants	SUS index (0–100)	Mean SUS score ≥ 72
C—Longitudinal knowledge gain	Quantify proficiency growth attributable to adaptive routing	Repeated-measures comparison of baseline and post-module scores from database logs	Experimental cohort tracked across modules	Normalized knowledge gain	Longitudinal grade increase > 15%

7.1. Case Study A: Controlled Pedagogical A/B Testing

Objective. To demonstrate statistically that adaptive content delivery increases information retention relative to static delivery.

Cohort design. A sample of 100 students is randomly allocated into two equal groups. The control group (GC) receives uniform content within course.php without personalization, whereas the experimental group (GE) receives a layout curated to its FSLSM profile.

Statistical hypothesis. Let μC and μE denote the mean Music History scores of the control and experimental groups respectively. The hypotheses are:

H0: $\mu E = \mu C$.

H1: $\mu E > \mu C$.

Execution strategy. Following a learning phase of sufficient duration to permit consolidation rather than mere exposure, attainment is extracted directly from the platform's assessment logs, so that the outcome measure derives from behavioral records rather than from self-reports. The analysis proceeds on two complementary levels. Significance testing establishes whether the observed difference is attributable to chance, while a standardized effect size of at least moderate practical magnitude—accompanied by interval estimates for both the mean difference and the effect size itself—guards against differences that are statistically reliable yet practically negligible. Participant attrition is treated as a finding in its own right rather than as noise; drop-out is reported per condition, and results are presented under both an intent-to-treat and a per-protocol reading [26,27].

7.2. Case Study B: Interface Usability Profiling (SUS Methodology)

Objective. To assess architectural layout efficiency, navigation mechanics, and overall user satisfaction.

Methodology. Following system use, participants complete the System Usability Scale. The instrument is retained in its original form deliberately; its value lies less in the ten items themselves than in the comparative base they index, and any modification would forfeit precisely the interpretability that motivates its selection. The alternation of positively and negatively worded statements serves to mitigate acquiescence bias. Because a composite can mask heterogeneity, item-level responses are inspected alongside it, so that the aspects of the interface driving the overall judgment—navigation, learnability, and confidence in

use—are identified rather than inferred. The subjective measure is further triangulated against the behavioral record of the user testing protocol in Section 6.

Data conversion. Item responses are normalized and rescaled to yield a composite index on a hundred-point scale. The resulting figure is not a percentage and should not be read as one; it is a position within an empirically derived distribution, and its interpretation depends entirely on the normative bands established for that distribution. Where the participant cohort is small, summary statistics are reported with resampling-based interval estimates or nonparametric summaries, since the distribution of the composite in small samples cannot be assumed to be symmetric.

Target benchmark. A threshold of ≥72 is adopted as the criterion for production-grade acceptance, consistent with empirically derived interpretation bands that place this value above the established average and within the "good" range [28,29].

7.3. Case Study C: Longitudinal Knowledge-Gain Analysis

Objective. To quantify the proficiency growth attributable to adaptive routing over the duration of a course.

Materials and methods. We use a repeated-measures design to compare attainment at enrollment with attainment post-module so that each learner is their own control and between-subject variation in prior knowledge is removed from the estimate. A normalized gain measure of the type introduced by Hake is computed—expressing each learner's improvement as the proportion of the headroom available to them that was actually realized—so that learners entering at different proficiency levels contribute comparably and the ceiling artifact is neutralized. The method for dealing with incomplete records is predefined; partial attempts and missing observations are recorded and their treatment is announced prior to the analysis. If learners provide multiple repeated observations, the design favors a mixed-effects formulation with learner-level random intercepts over a simple pre–post contrast. The model includes a within-subject effect size so that the magnitude of change remains interpretable independently of cohort size.

Acceptance criterion. A longitudinal grade increase exceeding 15% across the experimental cohort is set as evidence that adaptive routing produces durable learning gains rather than transient performance fluctuation.

7.4. Cross-Cutting Provisions

There are three provisions which are common to all case studies, and which are integral to the framework rather than incidental to any one design. First, the analysis plan (constructs, instruments, acceptance criteria, and handling incomplete data) is specified before data collection starts. Second, evaluation data is kept in both raw and derived form. The raw observations, individual responses, item-level outcomes, and timestamped interactions are stored, along with the composites computed from them, so that any derived quantity can be recomputed, audited or re-analyzed under alternative assumptions. Third, the scoring of the diagnostic instrument is reported transparently. The mapping of items to dimensions and the interpretation bands applied to the resultant values are fully specified in Sections 2.1 and 4.1, and the corresponding implementation is available in the OSEME platform codebase.

8. Results-Based Metrics and Discussion

An evidence-grounded evaluation requires the conversion of raw interaction records into actionable platform insight. Table 3 maps system features to their analytical indicators within the framework, pairing each component with a target source, a primary key performance indicator (KPI), and its strategic implication.

Table 3. Mapping of evaluated components to data sources, KPI targets, and strategic implication.

Evaluated System Component	Target Database/ Source Origin	Primary KPI Target	Strategic Implication
Authentication modals	users table insertion logs	<5% form abandonment	Validates interface clarity during sign-in and sign-up
Profiler (44 items)	quiz_style_results	Balanced dimensional distributions	Validates diagnostic consistency across dimensions
Knowledge engine	quiz_music_results	Longitudinal grade increase >15%	Measures knowledge acquisition from core courses
Data visualization	Chart.js canvas rendering	Load times under 200 ms	Ensures smooth UI transitions during real-time graphing

Analyzing these distributions enables administrators to refine content mappings on an evidential basis. If, for instance, longitudinal tracking reveals that verbal learners consistently underperform on particular music-theory modules, pedagogical staff can restructure the corresponding textual units to improve comprehension. The framework thus closes the loop between measurement and intervention: telemetry informs design, and subsequent telemetry verifies the effect of that design.

9. Conclusions and Future Work

This paper has proposed a results-based evaluation framework for Smart Learning Environments, using the OSEME architecture as a reference instantiation. By combining formally specified use cases with a structured user testing protocol and reproducible case-study designs, controlled A/B testing, SUS profiling, and longitudinal knowledge-gain analysis, the framework advances beyond subjective feedback toward evaluation anchored in concrete, database-grounded evidence. Its principal value is methodological and transferable: the tri-pivotal structure can be re-instantiated on other adaptive platforms without modification to its underlying logic.

Future iterations will explore the integration of machine learning classification models directly into the MySQL query pipeline, so that learner profiles are updated continuously from active interaction logs rather than relying solely on an initial diagnostic. This would enable genuine real-time personalization and would extend the learning analytics dimension from retrospective measurement toward predictive adaptation.

Author Contributions: Conceptualization, K.C.S., M.S. and A.K.; methodology, K.C.S., M.S. and M.M.; software, K.C.S. and A.K.; validation, K.C.S., M.S., M.M. and A.K.; formal analysis, K.C.S. and M.S.; investigation, K.C.S., M.M. and A.K.; resources, M.S. and L.B.; data curation, K.C.S., M.M. and A.K.; writing—original draft preparation, K.C.S.; writing—review and editing, K.C.S., M.S., M.M., A.K. and L.B.; visualization, K.C.S. and A.K.; supervision, M.S. and L.B.; project administration, M.S. and A.K.; funding acquisition, M.S. and L.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within the article.

Acknowledgments: The authors would like to thank the Department of Informatics at Ionian University for providing the infrastructure and support for this research.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Hwang, G.-J. Definition, Framework and Research Issues of Smart Learning Environments—A Context-Aware Ubiquitous Learning Perspective. *Smart Learn. Environ.* **2014**, *1*, 4. [CrossRef]
2. Woolf, B.P. *Building Intelligent Interactive Tutors: Student-Centered Strategies for Revolutionizing E-Learning*; Morgan Kaufmann Publishers/Elsevier: Amsterdam, The Netherlands; Boston, MA, USA, 2009.
3. Sofianos, K.; Stefanidakis, M. OSEME: A Smart Learning Environment for Music Education. 2024. Available online: https://zenodo.org/records/12791732 (accessed on 2 November 2026).
4. Felder, R. Learning and Teaching Styles in Engineering Education. *J. Eng. Educ.* **1988**, *78*, 674–681.
5. Felder, R.M.; Spurlin, J. Applications, reliability and validity of the index of learning styles. *Int. J. Eng. Educ.* **2005**, *21*, 103–112.
6. Ezzaim, A.; Dahbi, A.; Aqqal, A.; Haidine, A. AI-Based Learning Style Detection in Adaptive Learning Systems: A Systematic Literature Review. *J. Comput. Educ.* **2025**, *12*, 731–769. [CrossRef]
7. Essa, S.G.; Celik, T.; Human-Hendricks, N.E. Personalized Adaptive Learning Technologies Based on Machine Learning Techniques to Identify Learning Styles: A Systematic Literature Review. *IEEE Access* **2023**, *11*, 48392–48409. [CrossRef]
8. Brusilovsky, P. Adaptive Hypermedia. *User Model. User-Adapt. Interact.* **2001**, *11*, 87–110. [CrossRef]
9. Akbulut, Y.; Cardak, C.S. Adaptive Educational Hypermedia Accommodating Learning Styles: A Content Analysis of Publications From 2000 to 2011. *Comput. Educ.* **2012**, *58*, 835–842. [CrossRef]
10. Al-Azawei, A.; Serenelli, F.; Lundqvist, K. Universal Design for Learning (UDL): A Content Analysis of Peer Reviewed Journals from 2012 to 2015. *JoSoTL* **2016**, *16*, 39–56. [CrossRef]
11. Graf, S.; Liu, T.-C.; Kinshuk. Analysis of Learners' Navigational Behaviour and Their Learning Styles in an Online Course. *Comput. Assist. Learn.* **2010**, *26*, 116–131. [CrossRef]
12. George, G.; Lal, A.M. Review of Ontology-Based Recommender Systems in E-Learning. *Comput. Educ.* **2019**, *142*, 103642. [CrossRef]
13. Iqbal, M.; Sarwar, S.; Safyan, M.; Nasralla, M. Personalized and Adaptive E-Learning Systems for Semantic Web: A Systematic Review and Roadmap. *Int. J. Web Inf. Syst.* **2025**, *21*, 327–352. [CrossRef]
14. Siemens, G. Learning Analytics: The Emergence of a Discipline. *Am. Behav. Sci.* **2013**, *57*, 1380–1400. [CrossRef]
15. Brooke, J. SUS: A "Quick and Dirty" Usability Scale. In *Usability Evaluation in Industry*; Jordan, P.W., Thomas, B., McClelland, I.L., Weerdmeester, B., Eds.; CRC Press: Boca Raton, FL, USA, 1996; pp. 207–212. [CrossRef]
16. Bangor, A.; Kortum, P.T.; Miller, J.T. An Empirical Evaluation of the System Usability Scale. *Int. J. Hum.–Comput. Interact.* **2008**, *24*, 574–594. [CrossRef]
17. Lewis, J.R. The System Usability Scale: Past, Present, and Future. *Int. J. Hum.-Comput. Interact.* **2018**, *34*, 577–590. [CrossRef]
18. Romero, C.; Ventura, S. Educational Data Mining and Learning Analytics: An Updated Survey. *WIREs Data Min. Knowl.* **2020**, *10*, e1355. [CrossRef]
19. Kaliisa, R.; Jivet, I.; Prinsloo, P. A Checklist to Guide the Planning, Designing, Implementation, and Evaluation of Learning Analytics Dashboards. *Int. J. Educ. Technol. High Educ.* **2023**, *20*, 28. [CrossRef]
20. Kaliisa, R.; Misiejuk, K.; López-Pernas, S.; Khalil, M.; Saqr, M. Have Learning Analytics Dashboards Lived Up to the Hype? A Systematic Review of Impact on Students' Achievement, Motivation, Participation and Attitude. In *Proceedings of the 14th Learning Analytics and Knowledge Conference*; ACM: Kyoto Japan, 2024; pp. 295–304. [CrossRef]
21. *Visualizations and Dashboards for Learning Analytics*; Sahin, M., Ifenthaler, D., Eds.; Advances in Analytics for Learning and Teaching; Springer International Publishing: Cham, Switzerland, 2021. [CrossRef]
22. Nielsen, J.; Landauer, T.K. A Mathematical Model of the Finding of Usability Problems. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems—CHI '93*; ACM Press: Amsterdam, The Netherlands, 1993; pp. 206–213. [CrossRef]
23. Virzi, R.A. Refining the Test Phase of Usability Evaluation: How Many Subjects Is Enough? *Hum. Factors* **1992**, *34*, 457–468. [CrossRef]
24. Nielsen, J. *Usability Engineering*; Academic Press: Boston, MA, USA, 1993.
25. Cohen, J. *Statistical Power Analysis for the Behavioral Sciences*, 2nd ed.; Psychology Press: New York, NY, USA, 2009.
26. Sullivan, G.M.; Feinn, R. Using Effect Size—Or Why the P Value Is Not Enough. *J. Grad. Med. Educ.* **2012**, *4*, 279–282. [CrossRef] [PubMed]
27. Sauro, J. *Quantifying the User Experience: Practical Statistics for User Research*; Morgan Kaufmann: Cambridge, MA, USA, 2016.
28. Dogan, B.; Camurcu, A.Y. Association Rule Mining from an Intelligent Tutor. *J. Educ. Technol. Syst.* **2008**, *36*, 433–447. [CrossRef]
29. Lewis, J.R.; Sauro, J. Item Benchmarks for the System Usability Scale. *J. Usability Stud.* **2018**, *13*, 158–167.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.