A Card Game for Architectural Patterns Education

Maria Clara Ribeiro de Menezes[1], **Pedro Henrique Dias Valle**[2], **Alessandreia Marta de Oliveira**[1]

[1]Departamento de Ciência da Computação - UFJF - Juiz de Fora, MG – Brasil

[2]Universidade de São Paulo - (IME-USP) - Brasil

mariaclara.ribeiro@estudante.ufjf.br, pedrohenriquevalle@usp.br, alessandreia.oliveira@ufjf.br

Abstract. *Software Architecture education presents significant pedagogical challenges due to its abstraction. Existing approaches, such as educational games, often fail to address the skill of identifying architectural patterns from narrative-based problems. In this context, we propose Architectural Stories, a non-digital card game designed to fill this gap. The mechanics require players to investigate scenarios and deduce a solution by linking a problem diagnosis to a specific architectural pattern. To evaluate its effectiveness, we conducted a study with 34 undergraduate students using the MEEGA+ instrument. The results were positive, indicating that the game is usable, engaging, motivating, and an effective tool for understanding architectural patterns.*

Resumo. *O ensino de Arquitetura de Software apresenta desafios pedagógicos significativos devido ao seu nível de abstração. Abordagens existentes, como jogos educacionais, frequentemente não contemplam a habilidade de identificar padrões arquiteturais a partir de problemas baseados em narrativas. Nesse contexto, apresenta-se o Architectural Stories, um jogo de cartas não digital concebido para preencher essa lacuna. A mecânica exige que os jogadores investiguem cenários e deduzam uma solução ao relacionarem o diagnóstico do problema a um padrão arquitetural específico. Para avaliar sua eficácia, foi conduzido um estudo com 34 estudantes de graduação utilizando o MEEGA+. Os resultados foram positivos, indicando que o jogo é aplicável, envolvente, motivador e uma ferramenta eficaz para a compreensão de padrões arquiteturais.*

1. Introduction

Software Architecture (SA) is defined as the set of structures needed to reason about a software system, including its elements and relationships [Bass et al. 2021]. Architecture links design and requirements engineering by defining system components and their organization into communicating entities [Sommerville 2018]. Designing an architecture that meets system requirements is a key early development activity, as architectural decisions directly affect business objectives and functional and quality requirements [Oliveira et al. 2022, Sousa and Marques 2020]. A central aspect of architecturally driven development is selecting architectural styles or patterns [Xavier et al. 2002]. Architectural patterns describe reusable organizational structures that address recurring design problems in specific contexts and define roles, responsibilities, interactions, and trade-offs among quality attributes [Sommerville 2018, Bass et al. 2021].

Software architecture education is essential for the professional formation of future software engineers, but presents significant pedagogical challenges

[Araújo et al. 2024, Galster and Angelov 2016]. The abstract nature of architectural concepts limits meaningful learning through lectures [Castro 2023, Oliveira et al. 2022, Ouh and Irawan 2019], which often fail to provide experience in decision-making and negotiation [Montenegro et al. 2017]. Consequently, hands-on experience and active methodologies are required [Van Deursen et al. 2017]. A key challenge is training students to identify architectural patterns from narrative descriptions, requiring diagnostic reasoning beyond the classroom [Lago et al. 2019].

Several strategies have been explored, including project-based learning [Vidoni et al. 2018], gamification, and educational tools [Carvalho et al. 2025]. Educational games promote engagement, critical thinking, and content assimilation [Giacobo 2023, Souza et al. 2023, Feichas et al. 2021, Lelis and Marques 2024]. However, most proposals focus on design methods such as ADD [Cervantes et al. 2016], evaluation methods such as ATAM [Montenegro et al. 2017], or decision-making processes [Lago et al. 2019], rather than training pattern identification from narrative contexts. Existing games rely on predefined options or on method execution rather than deduction.

To bridge this gap, we introduce and evaluate **Architectural Stories**, a non-digital card game for architectural pattern education. Unlike initiatives focused on broad conceptual reinforcement [Sousa and Marques 2020], the game emphasizes narrative interpretation and pattern mapping. Inspired by Black Stories[1], players investigate "architectural crimes" to reach solutions, fostering practical understanding and logical reasoning.

The main objective of this work is to design, document, and formally evaluate an educational resource in the form of a narrative-based card game that supports students in interpreting architectural problem narratives and mapping them to relevant architectural patterns. To achieve this, we develop the Architectural Stories game by defining its narrative mechanics, learning intents, and pattern-oriented diagnostic structure. The study also investigates the game's usability, playability, and clarity of its mechanics for classroom use, as well as students' perceptions of learning, engagement, and usefulness. An empirical evaluation conducted with 34 undergraduate students provides evidence of the pedagogical value of the game as a complementary tool to traditional instruction, supporting active learning, diagnostic reasoning, and meaningful engagement with SA concepts.

The remainder of this paper is organized as follows. Section 2 reviews educational games related to SA. Section 3 presents the research methodology used to design Architectural Stories. Section 4 describes the game and its components. Section 5 reports the evaluation process and results. Section 6 discusses threats to validity, and Section 7 presents conclusions and future work.

2. Related Work

The scientific literature presents several game-based approaches addressing aspects of the software architecture lifecycle. This section analyzes how these initiatives relate to the specific problem addressed in this work, namely supporting the identification of Architectural Patterns from contextual descriptions, and explains why none fully meet this educational need. Four representative initiatives are discussed, highlighting their pedagogical goals, core mechanics, and main contributions. Smart Decisions [Cervantes et al. 2016] is designed to teach the Attribute-Driven Design (ADD) method by encouraging reflection on how design concepts influence quality attributes. Although it inspired the use of cards

[1]https://www.metropolybar.com.br/black-stories-diversao-macabra-para-quem-e-bom-de-deducao/

to represent architectural concepts, it diverges from our proposal by focusing on solution synthesis driven by explicit criteria rather than diagnostic reasoning. Because the activity relies on predefined options to satisfy clear requirements, it does not require students to interpret narrative contexts or infer architectural patterns from symptoms.

ATAM-RPG [Montenegro et al. 2017] teaches architectural evaluation through role-playing scenarios that simulate stakeholders and conflicting concerns, emphasizing negotiation and prioritization of quality attributes. Its contribution lies in exploring sociotechnical aspects of evaluation, which differ from the technical diagnostic reasoning addressed in this work. As a result, ATAM-RPG focuses on assessing existing architectures rather than training and deducing pattern solutions from problem descriptions.

DecidArch [Lago et al. 2019] focuses on Software Architecture Design Decision Making (SADDM), encouraging justification of decisions and consideration of multiple viewpoints. Although it teaches architectural reasoning, it diverges from Architectural Stories by relying on explicit "Concern Cards" and predefined design options. This approach supports reflection on trade-offs but removes the investigative challenge of identifying the root cause of architectural problems, a central aspect of our proposal.

The LEARN Board Game [Sousa and Marques 2020] reinforces foundational Software Architecture concepts through quiz-based progression, supporting review of quality attributes and patterns. However, it focuses on declarative knowledge rather than application skills and therefore does not engage learners in the analytical reasoning required to map contextual narratives to architectural solutions.

Table 1 compares these initiatives in terms of educational goals, formats, sample sizes, and evaluation strategies, showing that validation commonly relies on empirical studies with small to medium samples using instruments such as MEEGA+. This highlights the relevance of formal evaluation in educational games, which we also adopt. Overall, existing games support method simulation and conceptual reinforcement, but depend on structured inputs. Architectural Stories addresses this gap through a deductive, mystery-solving mechanic that isolates diagnostic reasoning, requiring students to infer architectural patterns from symptoms without predefined options.

Table 1. Comparison of Related Games

Game	Pedagogical Goal	Format	N	Evaluation Method & Metrics
Smart Decisions [Cervantes et al. 2016]	Attribute-Driven Design (ADD)	Card Game	41	Feedback Questionnaire (Likert 1–5): Realism, Fun, Challenge, Helpfulness.
ATAM-RPG [Montenegro et al. 2017]	ATAM & Negotiation	Web-based RPG	16	Two Surveys (Likert 1–5): 1. Perception of Effectiveness (Understanding, Skill). 2. User Satisfaction (Case Study, Tool Design).
DecidArch [Lago et al. 2019]	Software Architecture Design Decision Making (SADDM)	Card Game	83	Survey based on Learning Objectives (LO): LO1–Reasoning, LO2–Differences, LO3–Reconsideration; plus Playability (Fun, Clarity).
LEARN [Sousa and Marques 2020]	Fundamental Concepts (QA, Patterns)	Board Game	16	MEEGA+ Model: Usability (Aesthetics, Learnability...); Player Experience (Social Interaction, Fun, Learning...).

3. Research Methodology

To achieve the general objective, we followed the methodology below.

Phase 1: Identification of Existing Approaches

We conducted a Systematic Mapping Study (SMS) on Software Architecture education to identify pedagogical strategies and game-based approaches available in the literature [Menezes et al. 2026]. While the review highlighted games that effectively addressed design methods, architectural evaluation, decision-making, and conceptual reinforcement, our results revealed a specific gap: none of the identified games explicitly supported the inference of architectural patterns from narrative problem descriptions. This finding motivated the development of a new approach focused on diagnostic reasoning.

Phase 2: Definition of Learning Objectives

Based on the identified gap, we defined specific learning goals for the game. The central purpose is to transition students from passive conceptual understanding to active diagnostic application. The first objective is to enhance pattern recognition. To achieve this, students must interpret open-ended narrative scenarios. They are challenged to identify specific architectural symptoms within these contexts. The second objective is to foster deductive reasoning. The game cultivates the skill of linking a problem diagnosis to a specific architectural solution. This process relies on logical deduction to map symptoms to the correct pattern. Finally, the design encourages theoretical argumentation. Students are required to articulate their choices using technical vocabulary. They must also justify their decisions through trade-off analysis.

Phase 3: Game Design and Development

We developed Architectural Stories as a reusable educational artifact designed to train the identification of architectural patterns from narrative scenarios. To support this goal, we created "Architectural Crimes": open-ended narratives that describe failures or requirement violations without naming the underlying issue, requiring players to ask investigative questions to uncover it. We selected a set of architectural patterns as solutions, prioritizing fundamental structural styles and widely used distributed approaches. The set includes Layers, Client-Server, MVC, Pipe and Filter, Event-Driven Architecture, SOA, Publish-Subscribe, Microservices, Microkernel, Broker, P2P, Hexagonal Architecture, Saga, Adapter, Facade, Proxy, Mediator, Message Bus, and Blackboard.

Regarding format, we adopted a non-digital, card-based approach to prioritize face-to-face interaction, negotiation, and oral argumentation—skills essential for software architects but harder to practice in digital environments.Finally, the gameplay adopts a hybrid competitive–cooperative structure to promote engagement through competition while encouraging collaborative reasoning during investigation.

Phase 4: Evaluation Planning and Execution

To validate the game's educational effectiveness and usability, we conducted an empirical case study using the MEEGA+ model [Petri et al. 2019] in a Software Architecture course. The application took place in a classroom with 34 undergraduate students, who were organized into groups of three to five. Each group received the game materials and learned the rules exclusively from the manual, while the instructor provided only occasional clarifications to avoid influencing group reasoning. This setup aimed to assess whether students could understand the game solely by relying on the instruction manual.

Phase 5: Data Collection and Analysis

Immediately after gameplay, data were collected using the MEEGA+ questionnaire, answered individually and anonymously. The instrument combined quantitative Likert-scale items with open-ended questions to capture qualitative feedback on strengths

and areas for improvement. Finally, data analysis was performed. Quantitative data were analyzed using descriptive statistics (mean and standard deviation), while qualitative comments were examined alongside quantitative results to identify consistent perceptions about mechanics, engagement, and learning value.

4. Architectural Stories: An Educational Resource for Software Architecture

As a direct result of the design phase described in Section 3, we present **Architectural Stories**, a non-digital investigative deduction game. Implementing the planned narrative-deduction mechanics, the game adapts the "Black Stories" format to the domain of software design, effectively operationalizing the concept of "Architectural Crimes" defined in our methodology. It aims to transition students from an abstract understanding to the practical skill of identifying patterns in contextual descriptions.

Its design focuses on four competencies: (i) *pattern recognition*, linking narrative symptoms to appropriate architectural solutions; (ii) *systemic thinking*, by analyzing interactions and negative dependencies in the system; (iii) *logical reasoning*, through hypothesis formulation and testing; and (iv) *technical communication*, as players articulate architectural concepts when asking questions and presenting solutions.

Architectural Stories is played in groups of three to five and includes a scoreboard, **20 Mystery Cards**, pawns, an instruction manual, and pattern information cards. The **Mystery Card** is the core pedagogical component and has two sides. The **Enigma**, visible to all players, presents a brief narrative describing a software failure. As shown in Figure 1(a), the enigma "The 5,000-Line File" depicts a monolithic web page mixing HTML, validation logic, and SQL. The **Solution**, visible only to the **Architecture Master (AM)**, provides the complete answer. As illustrated in Figure 1(b), it identifies the root cause as low cohesion and high coupling and prescribes the *MVC* pattern, highlighting trade-offs such as improved maintainability versus increased complexity.

(a) The Enigma. (b) The Solution. (c) Colored Pawns.

Figure 1. Examples of game cards and colored pawns used in the activity.

We developed the **Insight Points Board** to track player performance centrally. Progress on the board is shown by colored 3D-printed pawns (Figure 1(c)), each corresponding to a participant and allowing status updates across rounds. Supplementary re-

sources include an **Instruction Manual** to ensure consistent use and **Pattern Reference Cards**. These cards provide theoretical scaffolding by summarizing objectives, structural characteristics, and design implications of each pattern. During gameplay, investigators consult the cards during investigation to relate symptoms revealed by the Architecture Master to definitions. This mechanic prioritizes analytical reasoning over rote memorization, enabling students to validate hypotheses before proposing solutions. All materials are available for replication and adoption by other instructors in an online repository[2], including the rulebook, scenario cards, pattern references, and facilitator guidelines.

The gameplay follows an asymmetric, rotating-role structure with one Architecture Master (AM) and multiple investigators (Figure 2), implementing the proposed hybrid competitive–cooperative model. In each round, the AM presents an Enigma and answers closed-ended questions with ”Yes”, ”No”, or ”Irrelevant”, while investigators analyze responses to form hypotheses. Players score by proposing a ”Complete Solution” combining diagnosis and pattern; correct answers grant +1 Insight Point (IP), while incorrect ones incur 1 IP. Rounds end with a debriefing, and the game concludes when a player reaches the end of the board or the time limit expires.

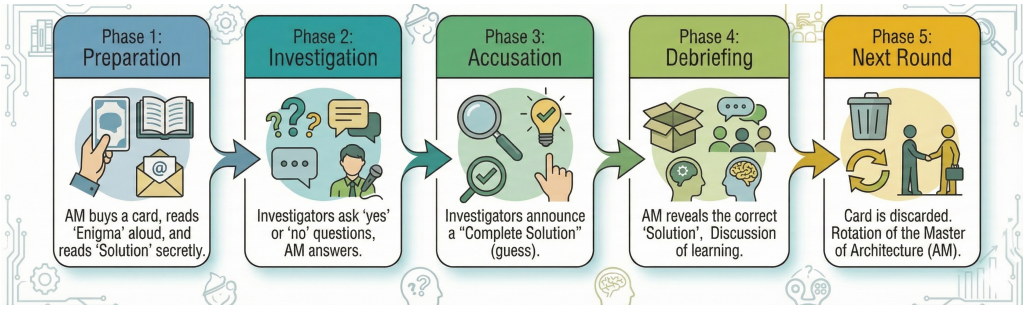

Figure 2. Game Round Structure.

5. Evaluation of Architectural Stories

5.1. Evaluation Planning

This study is an evaluative case study with a descriptive and exploratory nature, aiming to assess students' perceptions of the proposed educational resource, considering its usability, pedagogical value, and learning experience. To achieve this, we selected the MEEGA+ model as the sole evaluation instrument. MEEGA+ is a statistically validated model for computing education games, whose structure decomposes quality into *Usability* and *Player Experience*, providing sufficient coverage without ad hoc questionnaires. Its high reliability (Cronbach's alpha $\alpha = 0.928$) ensures rigorous data collection.

The instrument uses a standardized 5-point Likert-scale questionnaire. Although the standard MEEGA+ model includes 35 items, we adopted the authors' adaptation for non-digital games, excluding items 10, 11, and 12, which relate to software customization and error protection. Thus, the final instrument comprised 32 quantitative items, complemented by open-ended questions for qualitative feedback. The evaluation followed a non-experimental, one-shot post-test design. A single gameplay session was conducted to fit course constraints, lasting approximately 30 minutes and allowing multiple rounds. After gameplay, 40 minutes were allocated for completing the MEEGA+ questionnaire.

[2]https://github.com/MariaClara11/Architectural-Stories-Material.git

5.2. Evaluation Execution

The evaluation was conducted during a scheduled Software Engineering class with 34 undergraduate students (Figure 3). Students had prior exposure to fundamental architectural concepts, though familiarity with specific patterns varied, and no training on the game mechanics was provided. Participants formed groups , received the game materials and manual, and learned the gameplay autonomously to assess learnability.

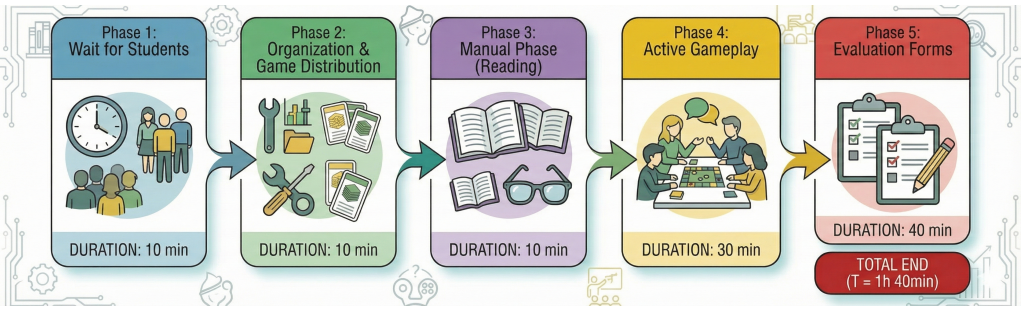

Figure 3. Flow of the evaluation application.

The gameplay session lasted approximately 30 minutes. Due to the self-paced nature of the groups, the number of completed rounds varied, with an estimated average of 8 rounds per group. Since the Architecture Master (AM) role rotates after each round, this volume ensured that the AM role was distributed among participants, allowing most students to experience both the investigator and facilitator perspectives. Throughout the session, the researcher remained present but intentionally limited interventions. Assistance was restricted to clarifying operational doubts about rules, without interfering with students' investigative reasoning or decision-making processes.

After gameplay, data collection was conducted using an online questionnaire (Google Forms). To ensure data integrity and reduce social desirability bias, the form collected anonymous responses and blocked multiple submissions from the same device. Quantitative data were exported for statistical analysis, while qualitative responses were analyzed to identify recurring perceptions and opportunities for improvement.

5.3. Results and Data Analysis

An analysis is presented based on data collected from 34 participants ($N = 34$) who evaluated Architectural Stories using the MEEGA+ instrument. The analysis integrates descriptive statistics, internal consistency indicators, and qualitative feedback to provide a comprehensive perspective on the game's pedagogical value and experiential quality.

Participants' Profile

A total of 34 students participated in the evaluation. Most were young adults, with 30 participants (88.2%) aged 18–28 years, followed by 3 participants (8.8%) aged 29–39, and 1 participant (2.9%) aged 40–50. This distribution indicates a strong concentration in younger age ranges. Regarding gender identity, the sample comprised 20 male participants (58.8%), 13 female participants (38.2%), and 1 non-binary participant (2.9%). Although unbalanced, the distribution reflects a diverse range of gender identities.

Regarding the frequency of non-digital games, the data revealed varied participation levels. Specifically, 18 participants (52.9%) reported playing rarely, while 12 participants (35.3%) played monthly. Only 2 participants (5.9%) played weekly, one (2.9%) played daily, and one (2.9%) never played non-digital games.

Quantitative Overview of the MEEGA+ Dimensions

Figure 4 presents the distribution of responses across all MEEGA+ items[3] . The predominance of dark blue bars (Strongly Agree") highlights the game's strongest aspects. Items related to *Social Interaction* and *Relevance* achieved near-unanimous consensus. For example, the statement *It is clear to me how the contents of the game are related to the course*" reached approximately 91% of Strongly Agree" responses, confirming students' perception of the activity's educational purpose. Likewise, the visual design was highly rated, with 88% strongly agreeing that *The colors used in the game are meaningful*".

Conversely, the chart explains the lower mean in the *Focused Attention* dimension. The item "*I forgot about my immediate surroundings while playing this game*" shows the greatest divergence, with a visible portion of disagreement (red and pink bars totaling approx. 27%). This result is expected for a classroom activity based on loud negotiation and group debate, where isolation from the environment is neither achieved nor intended.

Table 2 summarizes the descriptive statistics calculated from participant scores. The consistently high mean values across dimensions, many exceeding 4.5 on a 5-point Likert scale, demonstrate the strong acceptance and effectiveness of the game.

Table 2. Descriptive Statistics of MEEGA+ Dimensions ($N = 34$).

Factor	Subdimension	Items	Mean (μ)	Std. Dev. (σ)
Usability	Aesthetics	1–2	4.40	0.72
	Learnability	3–5	3.94	1.24
	Operability	6–7	4.29	0.99
	Accessibility	8–9	**4.81**	0.47
Player Experience	Confidence	13–14	4.07	0.47
	Challenge	15–17	4.12	0.98
	Satisfaction	18–21	4.36	0.76
	Social Interaction	22–24	**4.87**	0.34
	Fun	25–26	4.72	0.54
	Focused Attention	27–29	3.69	1.22
	Relevance	30–33	4.56	0.78
	Perceived Learning	34–35	4.63	0.69

Usability: Strong Aesthetic Appeal and Exceptional Accessibility

Usability scores indicate a highly positive perception of the game. The **Aesthetics** mean ($\mu = 4.40$) and low standard deviation ($\sigma = 0.72$) show broad consensus that the visual design met high-quality expectations, corroborated by qualitative comments praising the board, typography, and visual identity.

The strongest Usability subdimension was **Accessibility** ($\mu = 4.81$, $\sigma = 0.47$), with Item 9 reaching a high mean of 4.88. This confirms that the visual environment supports comprehension, even for students unfamiliar with tabletop mechanics.

This high accessibility score is significant when cross-referenced with participants' gaming habits (Section 5.3). As 52.9% reported rarely playing non-digital games, the positive evaluation suggests that the design effectively mitigates entry barriers and engages users regardless of their prior gaming experience.

[3]Full dataset of student responses available at: `https://tinyurl.com/yc4dpu59`

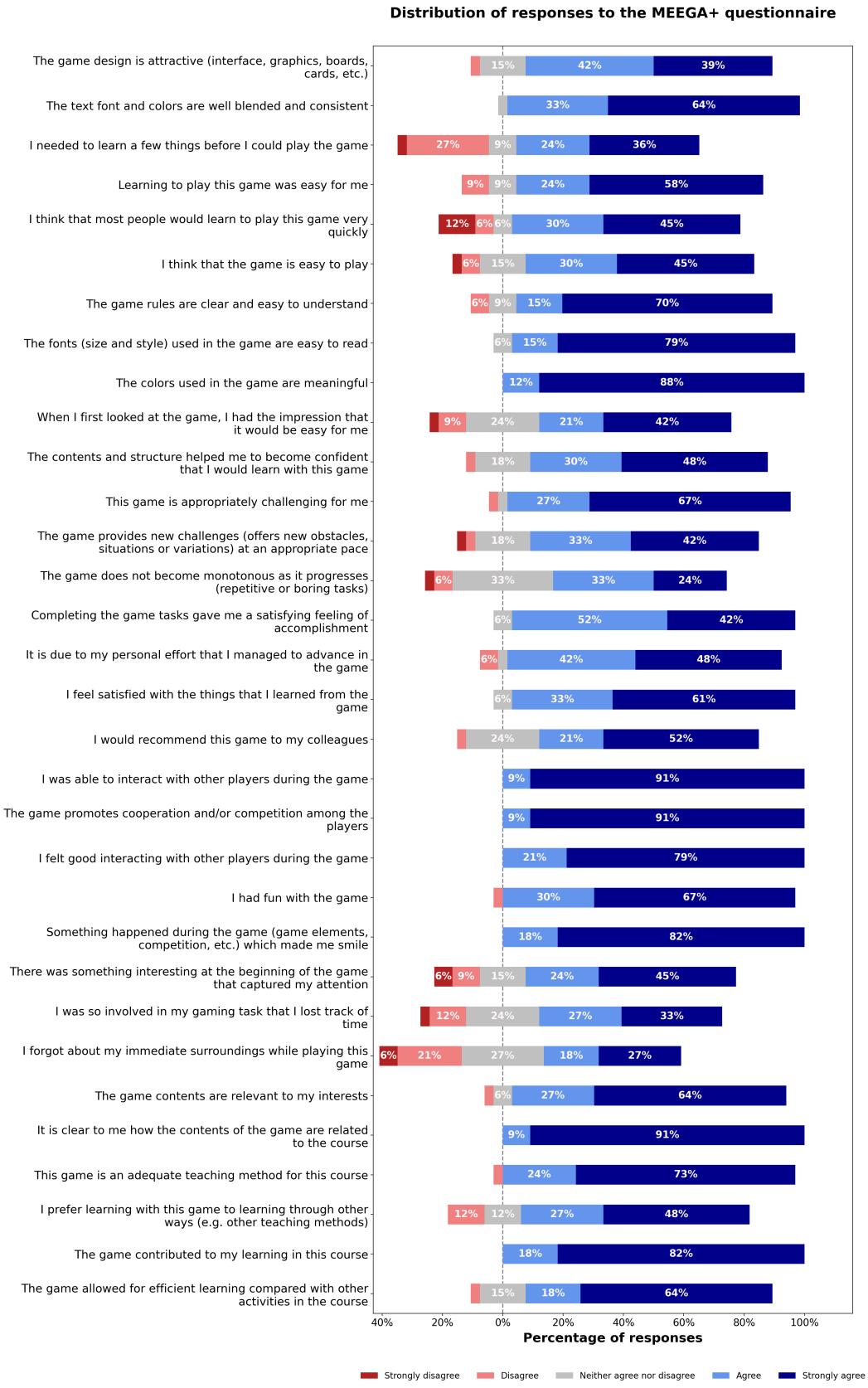

Figure 4. Distribution of responses across MEEGA+ items.

Conversely, **Learnability** ($\mu = 3.94$, $\sigma = 1.24$) showed the highest variability, reflecting heterogeneous familiarity with Architectural Patterns. For students with limited domain knowledge, the text-heavy descriptions posed a cognitive barrier, as noted by P9:

> *"The role of the patterns could be better structured, with images... to help those who do not know the patterns." (P9)*

Such insights reinforce the need for future iterations of the game to include visual scaffolding (e.g., diagrams, icons, and simplified pattern summaries). These refinements would reduce extraneous cognitive load and promote quicker onboarding.

Player Experience: Engagement, Collaboration, and Positive Learning Dynamics

The **Player Experience** results offer the strongest evidence of the effectiveness of Architectural Stories. All subdimensions scored above 4.0, except for *Focused Attention*, a finding that aligns with the nature of socially driven learning activities.

Social interaction emerged as the main driver of engagement. **Social Interaction** was the highest-rated subdimension across all MEEGA+ metrics ($\mu = 4.87$, $\sigma = 0.34$), indicating near-unanimous perception that the game promotes collaborative reasoning, debate, and collective meaning making, core elements of social constructivist learning.

> *" I liked the interactivity with the other classmates that the game promotes..." (P9)*

The results also indicate strong affective engagement, as evidenced by high scores in *Fun* ($\mu = 4.72$) and *Satisfaction* ($\mu = 4.36$). These values suggest a well-calibrated balance between challenge and reward, allowing players to enjoy the activity as they progress. Student comments further reinforce this reception. When asked "What did you like most about the game?", participants responded with comments such as:

> *"Reminded me of Black Stories. I found it really fun." (P19)*
> *"Learning while playing." (P14)*

Although *Focused Attention* obtained a lower mean ($\mu = 3.69$) and higher standard deviation ($\sigma = 1.22$), this should not be seen as a shortcoming. As the game is designed to stimulate discussion, negotiation, and shared analysis, the classroom becomes more dynamic. In this context, attention is distributed across peers and collective interactions rather than anchored in solitary concentration.

Finally, *Challenge* ($\mu = 4.12$) and *Confidence* ($\mu = 4.07$) form a coherent pair, indicating that tasks were perceived as demanding yet manageable. Respondents reported a clear sense of accomplishment when solving scenarios and identifying Architectural Patterns, suggesting that the game promotes cognitive stimulation and self-efficacy.

Perceived Learning and Educational Value

The most significant pedagogical outcomes are reflected in the **Relevance** ($\mu = 4.56$) and **Perceived Learning** ($\mu = 4.63$) scores, confirming that the game effectively reinforced core SA concepts beyond entertainment. The item "The game contributed to my learning in the course" achieved one of the highest means ($\mu = 4.82$), indicating strong consensus that the activity effectively supported learning. In qualitative responses, students emphasized how the game facilitated understanding:

> *"I was able to learn concepts that I didn't know before."* (P25)
> *"The reference sheet of the 'Patterns' is a direct and efficient summary of the content, and interacting with it helped reinforce the concepts."* (P6)

Synthesis of Findings

Based on the evidence, we observed that Architectural Stories provides a highly effective learning experience, particularly in terms of engagement, collaboration, and instructional value. The consistently high scores across Aesthetics, Accessibility, Social Interaction, Fun, and Relevance indicate that the game is well received both as an educational activity and as a dynamic classroom experience. Taken together, the quantitative strength of the results and the positive qualitative feedback underscore the pedagogical potential of the game as an active-learning strategy for teaching SA.

Although Learnability showed greater variability, due to the text-intensive pattern descriptions, qualitative feedback indicates this limitation can be mitigated through visual scaffolding, diagrams, and clearer explanations. These refinements would streamline onboarding without compromising core mechanics that contributed to the game's success. However, the evaluation also highlights opportunities for improvement, primarily in Learnability. Score variability and student comments indicate that text-heavy pattern descriptions created a cognitive barrier, particularly for learners with limited prior knowledge. Many participants suggested adding visual aids (diagrams, icons, simplified summaries, and concrete examples) to improve pattern comprehension.

Some students also noted that the role structure and turn flow could be more intuitive. Clearer rules and a more linear action sequence may reduce initial confusion during gameplay. Additionally, a few comments highlighted the need for more diverse scenarios and new pattern combinations to maintain engagement across multiple rounds. Overall, these observations outline a clear roadmap for improving usability, onboarding, and instructional clarity in future versions of the game.

6. Threats to Validity

Despite efforts to ensure methodological rigor in evaluating Architectural Stories, certain threats to validity may have impacted the findings. Identifying these threats enables a more accurate interpretation of the results [Mourão et al. 2020]. The main concerns and mitigation strategies are outlined below. A primary threat to **external validity** concerns the generalizability of results. The evaluation involved 34 students from a single Software Engineering course at one university. This sample may not represent students from other institutions, academic backgrounds, or industry professionals. Therefore, the findings should be interpreted cautiously when extrapolated to broader populations. Future replications with diverse cohorts and educational contexts would strengthen generalizability.

Threats to **construct validity** relate to whether the evaluation captured constructs such as motivation, engagement, and perceived learning. The study relied solely on self-reported MEEGA+ measures, which assess perceptions rather than objective learning outcomes. Despite anonymity, the novelty of the activity may have inflated reported motivation and enjoyment compared to traditional classroom practices. Regarding **internal validity**, the short gameplay duration (30 min) limits the assessment of long-term learning or retention. Additionally, autonomous rule learning from the manual may have led to variations in interpretation and gameplay flow, influencing participants' experiences and responses. Finally, threats to **conclusion validity** stem from the exclusive use of descriptive MEEGA+ statistics. Without inferential analysis, statistical significance or relationships between factors cannot be assessed, and the small sample size ($N = 34$) limits statistical power. In exploratory studies, conclusions should be interpreted cautiously.

7. Conclusion and Future Work

Teaching SA involves helping students reason about abstract design decisions and justify architectural choices; nevertheless, few game-based initiatives explicitly train learners to interpret narrative problems and infer Architectural Patterns. Architectural Stories was designed precisely to address this gap.

The main contribution of this work is the conception and evaluation of a narrative-deduction card game that positions students as investigators who analyze an "architectural crime" and collaboratively construct a Complete Solution. Unlike existing games that focus primarily on recognizing or assembling patterns, Architectural Stories emphasizes pattern interpretation in context, requiring players to reason from narrative clues to arrive at a logical justification for their architectural choices. By focusing on diagnostic reasoning, argumentation, and collaborative decision-making, the proposed resource directly contributes to the education of Software Engineering students, reinforcing competencies expected from future Software Architects. This focus distinguishes the game as an instructional tool that operationalizes a higher-order competency often difficult to achieve through traditional lectures or static exercises.

Beyond proposing the game, the study provides a reusable educational package, including a rulebook, scenario cards, facilitator guidelines, and reference materials on Architectural Patterns, enabling instructors to integrate it into undergraduate courses covering architectural styles, quality attributes, pattern selection, and early-stage design reasoning. Regarding prior knowledge, students benefit from a basic understanding of Software Architecture and Architectural Patterns, though full mastery is not required. Architectural Stories can be used before lectures, after introducing patterns, or for collaborative review, and its non-digital format encourages discussion, justification, and negotiation, skills central to architectural practice but often underexplored in traditional classrooms.

In future work, several directions are envisioned. One avenue is expanding the game materials with individual pattern cards, diagrams, and illustrated examples to enhance accessibility for learners with different experience levels. Second, the game should be replicated with diverse audiences, including undergraduates, graduate students, and industry practitioners, to compare perceptions across groups. Third, more rigorous experimental designs, including pre- and post-tests and additional instruments such as the Intrinsic Motivation Inventory (IMI) [Ryan et al. 1991] and the Instructional Materials Motivation Survey (IMMS) [Keller 2009], will enable a broader assessment of learning outcomes and motivation beyond self-reports. Fourth, comparative studies with other pedagogical approaches could better position the game within the instructional landscape. Finally, a digital or hybrid version is planned to support remote facilitation, automated feedback, adaptive difficulty, and wider dissemination.

Architectural Stories offers an engaging educational approach to teaching Architectural Pattern selection through narrative interpretation and collaboration. By refining materials, expanding evaluation, and exploring digital formats, the game may become a versatile tool for architectural education in academic and professional contexts.

AI Use Disclosure

We disclose that Artificial Intelligence (AI) tools were used only for non-substantive editorial assistance. ChatGPT was employed for language refinement, Gemini Colab for generating illustrative figures, and Grammarly for minor proofreading. All methodologi-

cal decisions, data analyses, and interpretations were performed by the authors alone. No AI tool contributed to the scientific content, claims, or conclusions of this paper.

References

Araújo, A., Coelho, A., Rodrigues, M. E., Viana, W., and Marques, A. B. (2024). Todo esforço será recompensado: Gamificaçao no ensino de arquitetura de software com o uso de badges. In *Workshop sobre Educação em Computação (WEI)*, pages 341–352. SBC.

Bass, L., Clements, P., and Kazman, R. (2021). *Software architecture in practice*. Addison-Wesley Professional.

Carvalho, E., Barbosa, J., Neto, R. B., Neto, V. G., and Valle, P. (2025). Decora: Um sistema de apoio ao ensino de decisões de projetos arquiteturais. In *Anais do V Simpósio Brasileiro de Educação em Computação*, pages 719–729, Porto Alegre, RS, Brasil. SBC.

Castro, L. M. (2023). Role-playing software architecture styles. In *2023 IEEE 20th International Conference on Software Architecture Companion (ICSA-C)*, pages 171–174. IEEE.

Cervantes, H., Haziyev, S., Hrytsay, O., and Kazman, R. (2016). Smart decisions: an architectural design game. In *Proceedings of the 38th International Conference on Software Engineering Companion*, pages 327–335.

Feichas, F. A., Seabra, R. D., and de Souza, A. D. (2021). Gamificação no ensino superior em ciência da computação: Uma revisão sistemática da literatura. *Revista Novas Tecnologias na Educação*, 19(1):443–452.

Galster, M. and Angelov, S. (2016). What makes teaching software architecture difficult? In *Proceedings of the 38th International Conference on Software Engineering Companion*, pages 356–359.

Giacobo, D. (2023). Dbboard game: Um jogo de tabuleiro para o ensino e aprendizagem de conceitos de banco de dados. In *Anais Estendidos do XXII Simpósio Brasileiro de Jogos e Entretenimento Digital*, pages 626–636, Porto Alegre, RS, Brasil. SBC.

Keller, J. M. (2009). *Motivational design for learning and performance: The ARCS model approach*. Springer Science & Business Media.

Lago, P., Cai, J. F., de Boer, R. C., Kruchten, P., and Verdecchia, R. (2019). Decidarch: Playing cards as software architects. In *Proceedings of the 52nd Hawaii International Conference on System Sciences (HICSS)*, pages 7815–7824. Hawaii International Conference on System Sciences (HICSS).

Lelis, M. and Marques, A. (2024). Learn 2.0: Evolução de um jogo de tabuleiro para o ensino de arquitetura de software. In *Anais do XXIII Simpósio Brasileiro de Jogos e Entretenimento Digital*, pages 1397–1408, Porto Alegre, RS, Brasil. SBC.

Menezes, M. C. R. d., Valle, P. H. D., and Oliveira, A. M. (2026). Two decades of software architecture education: State of the art, challenges, and future directions. In *Proceedings of the 48th International Conference on Software Engineering*, ICSE-SEET '26, Rio de Janeiro, Brazil. IEEE/ACM. Accepted for publication.

Montenegro, C. H., Astudillo, H., and Álvarez, M. C. G. (2017). Atam-rpg: A role-playing game to teach architecture trade-off analysis method (atam). In *2017 XLIII Latin American Computer Conference (CLEI)*, pages 1–9. IEEE.

Mourão, E., Pimentel, J. F., Murta, L., Kalinowski, M., Mendes, E., and Wohlin, C. (2020). On the performance of hybrid search strategies for systematic literature reviews in software engineering. *Information and software technology*, 123:106294.

Oliveira, B. R., Garcés, L., Lyra, K. T., Santos, D. S., Isotani, S., and Nakagawa, E. Y. (2022). An overview of software architecture education. In *Congresso Ibero-Americano em Engenharia de Software (CIbSE)*, pages 76–90. SBC.

Ouh, E. L. and Irawan, Y. (2019). Applying case-based learning for a postgraduate software architecture course. In *Proceedings of the 2019 ACM Conference on Innovation and Technology in Computer Science Education*, pages 457–463.

Petri, G., Von Wangenheim, C. G., and Borgatto, A. F. (2019). Meega+: Um modelo para a avaliação de jogos educacionais para o ensino de computação. *Revista Brasileira de Informática na Educação*, 27(03):52–81.

Ryan, R. M., Koestner, R., and Deci, E. L. (1991). Ego-involved persistence: When free-choice behavior is not intrinsically motivated. *Motivation and emotion*, 15(3):185–205.

Sommerville, I. (2018). Engenharia de software (10ª edição).

Sousa, T. A. and Marques, A. B. (2020). Learn board game: A game for teaching software architecture created through design science research. In *Proceedings of the XXXIV Brazilian Symposium on Software Engineering*, pages 834–843.

Souza, A. C. C., Souza, F. C. M., Vilela, R. F., and Valle, P. H. (2023). Pmbok game ii: Um jogo educacional para apoiar o ensino de gestão de projetos de software. In *Workshop sobre Educação em Computação (WEI)*, pages 454–464. SBC.

Van Deursen, A., Aniche, M., Aué, J., Slag, R., De Jong, M., Nederlof, A., and Bouwers, E. (2017). A collaborative approach to teaching software architecture. In *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education*, pages 591–596.

Vidoni, M., Montagna, J. M., and Vecchietti, A. (2018). Project and team-based strategies for teaching software architecture.

Xavier, J., Werner, C., and Travassos, G. (2002). Uma abordagem para a seleção de padrões arquiteturais baseada em características de qualidade. In *Anais do XVI Simpósio Brasileiro de Engenharia de Software*, pages 52–67, Porto Alegre, RS, Brasil. SBC.