Arandu: A Multiagent Intelligent Tutoring System Architecture with Generative AI for Gamified Teaching of Heuristic Evaluation in HCI

Julio Cesar R. P. Filho[1], **Lia Alessandra da S. Martins**[2]

[1]Instituto Federal de Educação, Ciência e Tecnologia do Amazonas (IFAM)
Campus Manaus Zona Leste – Manaus – AM – Brasil

{2024000046, lia.martins}@ifam.edu.br

Abstract. *Heuristic Evaluation (HE) in HCI faces challenges because it is a practical skill that requires multiple exposures to interfaces with deliberate usability violations, a need that is not addressed by existing solutions. This paper proposes Arandu, a gamified Intelligent Tutoring System (ITS) with a multiagent architecture for teaching Nielsen's Usability Heuristics. The key contribution is the use of Large Language Models (LLMs) to dynamically generate simulated interfaces with deliberately embedded usability violations, ensuring continuous exercise variability. The system also incorporates LLM-based pedagogical feedback and adaptive spaced repetition. The proposal is demonstrated through a high-fidelity prototype and evaluated through theoretical analysis of architecture decisions.*

Resumo. *O ensino de Avaliação Heurística (AH) em IHC enfrenta desafios por ser uma habilidade prática que exige múltiplas exposições a interfaces com falhas de usabilidade, necessidade não suprida pelas soluções existentes. Este trabalho propõe o Arandu, um STI gamificado com arquitetura multiagente para o ensino das Heurísticas de Nielsen. O diferencial é o uso de Large Language Models (LLMs) para geração dinâmica de interfaces simuladas com falhas deliberadamente inseridas, garantindo variabilidade contínua de exercícios. O sistema incorpora ainda feedback pedagógico por LLM e revisão espaçada adaptativa. A proposta é demonstrada por um protótipo de alta fidelidade e avaliada por análise teórica das decisões de arquitetura.*

1. Introdução

A Avaliação Heurística (AH) é um método de inspeção de usabilidade amplamente adotado na prática de Interação Humano-Computador (IHC), no qual especialistas examinam interfaces com base em princípios estabelecidos para identificar problemas de usabilidade [Nielsen 1994]. Apesar de sua relevância, a competência não se desenvolve pela memorização das heurísticas, mas pela sua aplicação repetida em contextos variados, o que torna o ensino dessa técnica um desafio pedagógico em IHC bastante abordado na literatura. A principal barreira ao ensino de AH, documentada por Rosson et al. [2004] e Kang et al. [2022], é a escassez de experiências práticas realistas e diversificadas. Somado a isso, a dificuldade de prover *feedback* imediato e personalizado atua como um obstáculo central em contextos acadêmicos [VanLehn 2011].

Diante desse contexto, este trabalho apresenta o Arandu (termo de origem Guarani que sintetiza os conceitos de sabedoria e conhecimento vivenciado), um Sistema Tutor Inteligente (STI) gamificado com arquitetura multiagente para o ensino e a prática das Heurísticas de Nielsen. O sistema incorpora revisão espaçada adaptativa e economia virtual baseada em Açaí *Bits*, escolha que conecta a experiência lúdica à identidade cultural amazônica, para sustentação do engajamento a longo prazo. A contribuição central reside no uso de Inteligência Artificial Generativa (IAG) para gerar dinamicamente interfaces simuladas com falhas de usabilidade deliberadas.

Este trabalho está organizado da seguinte forma: a Seção 2 relata os referenciais teóricos sobre STIs e Large Language Models (LLMs), AH como objeto de ensino, gamificação e revisão espaçada; a Seção 3 discute os trabalhos relacionados, destacando a lacuna de pesquisa; a Seção 4 descreve a metodologia utilizada; a Seção 5 detalha a arquitetura multiagente do Arandu; a Seção 6 apresenta o protótipo de alta fidelidade; a Seção 7 discute decisões arquiteturais e limitações; e a Seção 8 traz as considerações finais.

2. Fundamentação Teórica

Esta seção apresenta os embasamentos teóricos que fundamentam a proposta do Arandu. Inicialmente, são discutidos na Subseção 2.1, os STIs e o papel emergente dos LLMS na educação personalizada. Em seguida, a Subseção 2.2 caracteriza a Avaliação Heurística como objeto de ensino, e a Subseção 2.3 aborda a gamificação como estratégia motivacional no contexto de STIs. Por fim, a Subseção 2.4 explora a Revisão Espaçada e a Curva do Esquecimento. A correlação entre essas quatro bases teóricas sustenta a arquitetura multiagente proposta, na qual a geração dinâmica de exercícios, o engajamento gamificado e a retenção de longo prazo são integrados de forma sistêmica.

2.1. Sistemas Tutores Inteligentes e *Large Language Models*

Sistemas Tutores Inteligentes (STIs) representam uma abordagem consolidada para o ensino adaptativo mediado por computador, classicamente combinando quatro módulos: (i) Domínio, (ii) Aluno, (iii) Tutor e (iv) Interface para oferecer instrução personalizada [Mousavinasab et al. 2018]. A eficácia dessa abordagem é respaldada por evidências consistentes na literatura, conforme demonstrado por VanLehn [2011], que revelou ganhos próximos aos de tutores humanos individuais; por Xu et al. [2019], que confirmaram efeito positivo sobre aproximadamente 10.000 estudantes do ensino básico; e mais recentemente, *LearnLM Team & Eedi*[2025] que reportaram desempenho equivalente ou superior ao de tutores humanos em um ensaio clínico randomizado com 165 estudantes no Reino Unido utilizando um modelo de IAG otimizado para tutoria.

Com o advento dos *Large Language Models* (LLMs), uma nova geração de sistemas de tutoria baseados em Inteligência Artificial (IA) emergiu, capaz de gerar conteúdo e fornecer *feedback* discursivo com sofisticação próxima à de tutores humanos [LearnLM Team and Eedi 2025]. Sistemas como o Ruffle&Riley [Schmucker et al. 2024] e o SPL (*Socratic Playground for Learning*) [Zhang et al. 2024] empregam LLMs para tutoria conversacional, mas o potencial dos LLMs para geração dinâmica de interfaces e artefatos visuais permanece pouco explorado no contexto de STIs.

2.2. Avaliação Heurística como Objeto de Ensino

A Avaliação Heurística (AH) é um método de inspeção de usabilidade proposto por [Nielsen 1994], no qual avaliadores examinam uma interface e identificam violações de um conjunto de princípios gerais de usabilidade, as 10 Heurísticas de Nielsen. Entre elas estão princípios como Visibilidade do *Status* do Sistema, Controle e Liberdade do Usuário e Prevenção de Erros. Por sua natureza analítica e aplicada, a AH é amplamente ensinada em disciplinas de IHC em cursos de Computação, Engenharia de *Software* e *Design*.

O ensino da AH apresenta um desafio estrutural em que a habilidade de identificar violações heurísticas só se desenvolve com prática em interfaces reais e variadas. Kang et al. [2022] destacaram que abordagens experienciais e práticas são as mais adotadas por educadores de IHC justamente por esse motivo. Iniciativas como o Heureka [Sobrino-Duque et al. 2022], revelam o potencial da gamificação para motivar o estudo das heurísticas de Nielsen, mas dependem de conteúdo estático, o que limita a variabilidade da prática oferecida aos alunos. A criação e manutenção de um banco diversificado de interfaces com falhas de usabilidade controladas acarreta um custo operacional alto para os docentes, restringindo a escala do treino disponível.

2.3. Gamificação na Educação

Gamificação é definida como o uso de elementos de *design* de jogos em contextos não-lúdicos com o objetivo de aumentar o engajamento e a motivação dos participantes [Deterding et al. 2011]. Seaborn e Fels [2015], em uma revisão sistemática da literatura sobre gamificação em sistemas interativos, identificaram resultados predominantemente positivos quanto ao seu potencial para motivar a ação e promover o engajamento em contextos educacionais, ressaltando que as bases teóricas convergem para teorias psicológicas de motivação. No contexto específico de Sistemas Tutores Inteligentes, Dermeval et al. [2019] propuseram um modelo ontológico para a aplicação de gamificação em STIs, conectando teorias de motivação e elementos de *design* de jogos aos módulos clássicos de um STI.

A Teoria da Autodeterminação distingue duas fontes motivacionais [Ryan and Deci 2000]: (i) a motivação extrínseca, sustentada por recompensas externas como moedas, pontos e *rankings*, e (ii) a motivação intrínseca, alimentada pela sensação de maestria, autonomia e progresso percebido. Dermeval et al. [2019] identificaram que STIs gamificados mais eficazes tendem a combinar os dois eixos, de modo que elementos extrínsecos sustentam o engajamento inicial, enquanto a progressão estruturada em níveis e a percepção de competência crescente cultivam a motivação intrínseca a longo prazo.

2.4. Revisão Espaçada e Curva do Esquecimento

A Teoria da Curva do Esquecimento, proposta originalmente pelo psicólogo Hermann Ebbinghaus no século XIX, explica que a retenção de informações declina progressivamente ao longo do tempo na ausência de revisões sistemáticas, com queda mais acentuada nas primeiras 24 horas após a aprendizagem. Lima e Nascimento [2026] abordam esse fenômeno no contexto de tecnologias educacionais, destacando sua relevância para sistemas que visam combater o esquecimento a longo prazo.

A técnica de Revisão Espaçada propõe combater esse processo por meio de revisões planejadas em intervalos crescentes, calculados com base no desempenho do aluno.

Conteúdos bem dominados são revisados em intervalos maiores, enquanto conteúdos com baixo domínio são reapresentados rapidamente, a abordagem é adotada por plataformas como o Duolingo e representa uma estratégia promissora para habilidades de aquisição gradual como a Avaliação Heurística.

3. Trabalhos Relacionados

No domínio da avaliação heurística e educação em usabilidade, Sobrino-Duque et al. [2022] desenvolveram o Heureka como estratégia gamificada para o ensino das Heurísticas de Nielsen, com efeitos motivacionais positivos, mas sem melhorias significativas de desempenho em relação a métodos tradicionais e com limitações de variabilidade por depender de conteúdo estático. Aljedaani et al. [2026] também se destacaram ao introduzirem o *HeuristicBuilder*, uma ferramenta gamificada multimodal avaliada por 224 participantes, oferecendo artefatos visuais interativos, porém estáticos, limitando a variabilidade dos exercícios e sem integração de LLMs ou mecanismos de revisão espaçada.

Outra linha de pesquisa envolve sistemas de tutoria conversacional baseados em LLM. O Ruffle&Riley [Schmucker et al. 2024] e o SPL [Zhang et al. 2024] empregaram GPT-4, respectivamente, para tutoria dialógica em diferentes domínios e para aplicar o método socrático via questionamentos iterativos. No entanto, esses sistemas podem ser considerados insuficientes, pois a prática de Avaliação Heurística exige que o aluno examine e julgue interfaces concretas. A comparação em cinco dimensões é apresentada na Tabela 1, evidenciando que nenhum sistema identificado reúne simultaneamente as cinco dimensões necessárias ao ensino prático de AH.

Tabela 1. Comparação entre sistemas de aprendizagem.

Sistema	Foco em AH	Usa LLM	Exercícios visuais gerados por LLM	Gamificação	Revisão espaçada
Heureka (Sobrino-Duque et al., 2022)	✓	×	×	✓	×
HeuristicBuilder (Aljedaani et al., 2026)	✓	×	×	✓	×
Ruffle&Riley (Schmucker et al., 2024)	×	✓	×	×	×
SPL (Zhang et al., 2024)	×	✓	×	×	×
Arandu (este trabalho)	✓	✓	✓	✓	✓

Ao buscar trabalhos relacionados nos anais do ENIAC e do *BRACIS* dos últimos cinco anos, combinando as *strings* de busca (”*intelligent tutoring systems*”*OR* ”sistema tutor inteligente”), (”*human-computer interaction*”*OR* ”IHC”) e (”ENIAC”*OR* ”*BRACIS*”), notou-se uma escassez de projetos que envolvessem STIs de forma geral, com os quais fosse possível realizar a análise nas 5 dimensões relatadas, impossibilitando a aplicação de critérios de inclusão e exclusão. Dentre os principais projetos encontrados no estado da arte, individualmente, nenhum dos sistemas atende ao conjunto de requisitos necessários ao ensino prático de AH. Os artigos com foco em avaliação heurística limitam-se a conteúdo estático sem integração de LLMs, os tutores conversacionais baseados em IAG operam sem interfaces visuais concretas, condição indispensável à prática de inspeção de

usabilidade, e nenhum dos sistemas analisados combina gamificação e revisão espaçada adaptativa no mesmo artefato. Portanto, a avaliação comparativa revela que o Arandu é o único sistema a reunir simultaneamente as cinco dimensões investigadas, respondendo à lacuna evidenciada pela Tabela 1 e constituindo a contribuição central deste trabalho.

4. Metodologia

Este trabalho adota a *Design Science Research* (DSR) como abordagem metodológica. Hevner et al. [2004] definem a DSR como o paradigma de pesquisa que busca ampliar as capacidades humanas e organizacionais por meio da criação de artefatos inovadores, em contraposição à ciência comportamental, que busca explicar ou prever fenômenos. O Arandu se enquadra nessa perspectiva: é um artefato computacional proposto para resolver um problema pedagógico concreto, o ensino prático e escalável da Avaliação Heurística em IHC, cuja solução não existe na literatura em forma integrada.

O desenvolvimento segue as seis etapas do processo DSR propostas por Peffers et al. [2007], conforme apresentado na Figura 1, fundamentadas nos *guidelines* de qualidade de Hevner et al. [2004]. A identificação do problema e motivação consistiu no levantamento das dificuldades no ensino de AH, documentadas na Seção 1. A definição dos objetivos produziu três requisitos centrais: (i) geração dinâmica de exercícios por IAG, (ii) gamificação para engajamento e (iii) revisão espaçada para retenção. O plano do desenvolvimento resultou na arquitetura multiagente descrita na Seção 5, em que a especialização em três agentes permite separar responsabilidades de geração de código, *feedback* pedagógico e controle de progressão algorítmica, o que não seria viável com um único agente generalista. A demonstração é realizada pelo protótipo de interface apresentado na Seção 6.



Figura 1. Processo DSR aplicado ao Arandu. Fonte: Peffers et al. [2007].

A avaliação, dado o estágio atual do trabalho, é conduzida por análise teórica das decisões de arquitetura à luz da literatura sobre STIs, gamificação e IAG na educação. Essa abordagem corresponde ao método *Informed Argument* definido na Tabela 2 de Hevner et al. [2004], classificado pelos autores como adequado para artefatos inovadores em estágio inicial, sem comparativo direto disponível na literatura. A avaliação empírica está planejada como trabalho futuro, e a comunicação da proposta se dá por meio deste artigo, encerrando o ciclo de Peffers et al. [2007].

5. O Sistema Arandu

As escolhas de tecnologias e modelos de IA descritas nesta seção constituem uma proposta de implementação, podendo ser ajustadas durante o desenvolvimento. Iniciando

pela Subseção 5.1 que demonstra a visão geral da arquitetura, seguido da Subseção 5.2 que descreve os agentes especializados do STI Arandu, assim como a Subseção 5.3 traz o currículo e a gamificação do sistema, e finalizando com a Subseção 5.4 que detalha o ciclo de tutoria do STI.

5.1. Visão Geral da Arquitetura

O Arandu é concebido como uma aplicação *web* com arquitetura multiagente que separa explicitamente dois tipos de conhecimento: o teórico (definições das Heurísticas de Nielsen, regras de violação e estrutura curricular) e o generativo (tutorias, cenários e interfaces dos exercícios práticos). O primeiro fica armazenado em um Módulo de Domínio estático em formato *JSON*, cujo conteúdo nenhum LLM reproduz ou redefine por conta própria. O segundo é produzido dinamicamente pela IAG, sempre a partir das definições fornecidas pelo Módulo de Domínio, mas sem nunca definir por conta própria o que constitui uma violação heurística. Essa separação pretende garantir que a teoria de Nielsen permaneça íntegra mesmo se a IA errar ou alucinar. As implicações dessa decisão de arquitetura são analisadas na Seção 7. O sistema propõe três camadas funcionais, ilustradas na Figura 2.

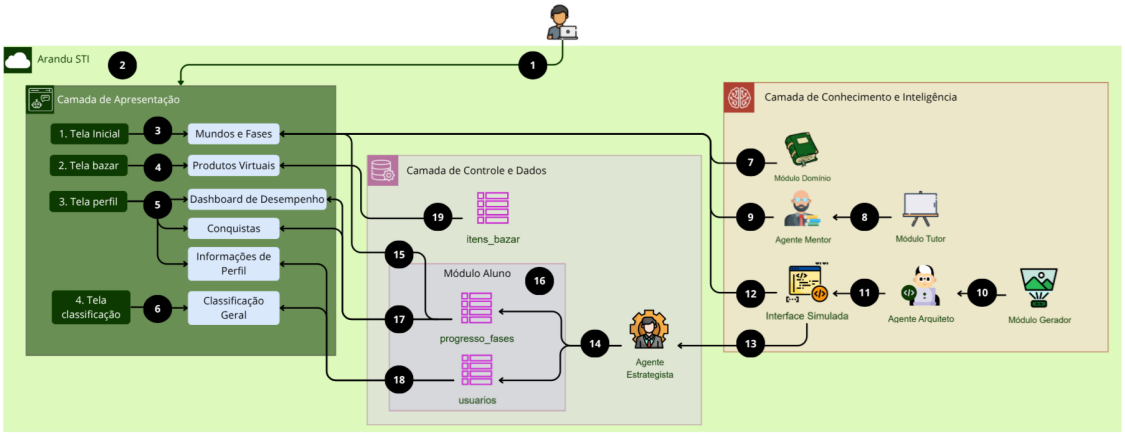


Figura 2. Arquitetura multiagente do STI Arandu.

Camada de Apresentação (fluxos 1-6): o aluno acessa as cinco telas de navegação, cobrindo mapa de mundos, loja virtual, perfil, classificação e fases de tutoria.

Camada de Conhecimento e Inteligência (fluxos 7-12): (7) Módulo de Domínio, que determina regras e conteúdo de cada fase; (8) Módulo Tutor, que orquestra a estratégia pedagógica do Agente Mentor; (9) Agente Mentor, que gera aulas e *feedbacks*; (10) Módulo Gerador, que sorteia contextos para garantir rejogabilidade; e Agente Arquiteto, que gera a Interface Simulada com a falha de usabilidade (11-12).

Camada de Controle e Dados (fluxos 13-19): Agente Estrategista, que processa os resultados recebidos pela Interface Simulada e atualiza as tabelas *progresso_fases* e *usuarios* com recompensas e datas de revisão espaçada (13-14); e Módulo Aluno, que consolida o progresso das fases e o desempenho do aluno para alimentar os Mundos e Fases, o *Dashboard* e a Classificação Geral (15-18). Por fim, a tabela *itens_bazar* alimenta os Produtos Virtuais da Tela Bazar (19).

5.2. Agentes Especializados

O Agente Arquiteto é projetado como o motor de geração de exercícios práticos do sistema. Tendo como modelo candidato o *Qwen* 2.5 *Coder (32B)*, especializado em geração de código, ele receberia as regras de violação heurística definidas no Módulo de Domínio e um contexto de cenário sorteado aleatoriamente (como aplicativo de *delivery*, *internet banking* ou rede social) e geraria sob demanda o código *JSX/TailwindCSS* de uma interface simulada com a falha de usabilidade deliberadamente inserida.

O Agente Mentor atuaria como o tutor pedagógico do sistema. Tendo como modelos candidatos o *Llama* 3 (*8B Instruct*) ou o *Mistral* (v0.3), otimizados para diálogo instrucional, ele seria responsável por duas funções: (i) gerar o conteúdo dos tutoriais conceituais de cada fase, apresentando as heurísticas por meio de analogias do cotidiano, e (iii) fornecer *feedback* corretivo quando o aluno erra, explicando o porquê do erro sem entregar diretamente a resposta. O Agente Mentor nunca revelaria o gabarito de forma direta, adotando uma postura socrática que estimularia o raciocínio analítico do aluno.

O Agente Estrategista não utilizaria LLM, ele seria um agente algorítmico puro, projetado em *Python*, responsável por calcular a data de próxima revisão de cada fase com base no desempenho imediato do aluno. O algoritmo classificaria o desempenho em três níveis: (i) Maestria (taxa de acerto $\geq 90\%$), que projetaria a revisão para 7 dias; (ii) Competência (70% a 89%), que exigiria reforço em 3 dias; e (iii) Dificuldade ($< 70\%$), que acionaria revisão prioritária em 24 horas. Ao atingir a data de revisão, o estado visual da fase no mapa mudaria de dourado para rachado, sinalizando ao aluno a necessidade de restauração do conteúdo.

5.3. Currículo e Gamificação

O currículo do Arandu é organizado em uma Árvore de Habilidades composta por cinco Mundos Temáticos, nos quais as heurísticas são agrupadas por afinidade pedagógica. Esses mundos correspondem a Comunicação, Navegação, Erros, Carga Cognitiva e Flexibilidade e Ajuda, representados, respectivamente, pelas Figuras 3(a), 3(b), 3(c), 3(d) e 3(e). Cada mundo contém seis fases progressivas, totalizando 30 cenários únicos gerados por IA, e culmina em uma fase de Chefão Final, caracterizada por uma auditoria de interface com múltiplas violações heurísticas simultâneas.

A identidade amazônica permeia o visual do mapa, com cada Mundo representado por um cenário fotográfico de significado próprio. A jornada percorre o Rio Amazonas (Figura 3(a)) e avança por espaços emblemáticos que revelam contrastes da região: uma área de desmatamento, que expõe a dura realidade socioambiental (Figura 3(b)); a Ponte sobre o Rio Negro, maior ponte estaiada do Brasil, símbolo de conectividade e desenvolvimento regional (Figura 3(c)); o Teatro Amazonas, ícone do legado cultural e histórico de Manaus (Figura 3(d)); e a Zona Franca de Manaus, polo de alta tecnologia que insere o estudante no contexto econômico e tecnológico regional (Figura 3(e)) . Ao trocar o estereótipo da floresta intocada por marcos reconhecíveis pelo público-alvo, os estudantes de cursos de computação do norte do Brasil, a jornada pedagógica promove pertencimento e potencializa o engajamento intrínseco previsto pela Teoria da Autodeterminação [Ryan and Deci 2000].

A progressão é reforçada por uma economia virtual baseada em Açaí *Bits*. O desempenho em cada fase geraria recompensas trocáveis na loja virtual por itens de auxílio,



(a) Mundo 1. (b) Mundo 2. (c) Mundo 3.

(d) Mundo 4. (e) Mundo 5.

Figura 3. Cenário dos Mundos Temáticos do Arandu.

como Congeladores de Ofensiva e recargas de Vidas, operacionalizando os dois eixos motivacionais da Teoria da Autodeterminação [Ryan and Deci 2000].

5.4. Ciclo de Tutoria

O ciclo de tutoria, exibido na Figura 4, organiza o percurso de aprendizagem do estudante a partir da integração entre as três camadas descritas na Seção 5.1. Essa organização permite representar, de forma sequencial, como as decisões realizadas na interface são processadas pelos módulos inteligentes do sistema e posteriormente registradas para fins de acompanhamento, progressão e revisão espaçada.

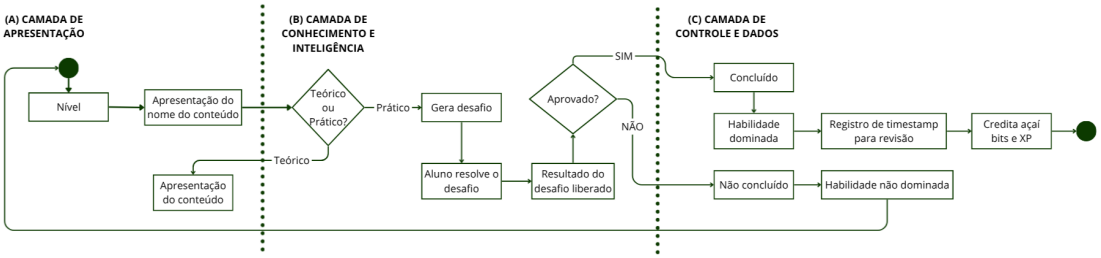


Figura 4. Ciclo de tutoria do Arandu.

O processo inicia-se, conforme ilustrado na Figura 4, na Camada de Apresentação, onde o estudante seleciona o nível a ser realizado e acessa o conteúdo correspondente, podendo seguir por uma etapa teórica ou prática. Em seguida, a Camada de Conhecimento e Inteligência conduz a experiência de aprendizagem: quando o percurso é teórico, o Agente Mentor disponibiliza a base conceitual necessária para o estudo; quando o percurso é prático, o Módulo Gerador sorteia um contexto e o Agente Arquiteto produz um desafio inédito, no qual o estudante deve identificar problemas de usabilidade em uma interface simulada.

Após a resolução do desafio, o resultado é encaminhado à Camada de Controle e Dados, responsável por avaliar o desempenho e atualizar o estado de progressão do estudante. Em caso de aprovação, a habilidade é marcada como dominada na Árvore de Habilidades, um *timestamp* é registrado para controle da revisão espaçada e as recompensas em Açaí *Bits* e *XP* são creditadas como reforço positivo. Caso a habilidade não seja dominada, o estudante retorna para uma nova tentativa, recebendo *feedback* corretivo do

Agente Mentor para apoiar a compreensão dos erros e orientar a continuidade do processo de aprendizagem.

6. Protótipo

O Arandu foi prototipado como uma interface de alta fidelidade no Figma[1]. Sua navegação principal é composta por cinco telas: Inicial, Bazar, Classificação, Perfil, Fase Tutorial e Fase Prática. A Tela Inicial organiza a Árvore de Habilidades, com os Mundos Temáticos e o estado de progressão de cada fase, funcionando como o mapa central do sistema. A Tela Bazar exibe a loja virtual, na qual Açaí *Bits* podem ser trocados por itens de auxílio à experiência gamificada. A Tela Classificação apresenta o *ranking* geral dos usuários, enquanto a Tela Perfil consolida o *dashboard* de desempenho, incluindo conquistas, nível de *XP* e histórico de progresso por heurística.



Figura 5. Tela de prática do Arandu.

A Tela de Fase Tutorial o conteúdo teórico da heurística em estudo e oferece acesso ao Agente Mentor para esclarecimento de dúvidas conceituais. A Tela de Fase Prática, ilustrada na Figura 5, por sua vez, representa a contribuição técnica central do sistema. Nela, o espaço onde o código *JSX* gerado pelo Agente Arquiteto seria renderizado está representado dentro de uma tela simulada, e o aluno identificaria a falha de usabilidade clicando na região problemática da interface. A Tela Fase (Chefão), ao final de cada Mundo, seguiria o mesmo modelo com múltiplas violações simultâneas. O protótipo evidencia a viabilidade da proposta e serve de base para a implementação técnica planejada como trabalho futuro imediato.

7. Discussão e Limitações

A decisão mais relevante do Arandu é a separação entre o conhecimento teórico e o generativo. Ao confinar as definições canônicas das 10 Heurísticas de Nielsen em um *da-*

[1]Protótipo disponível em: `https://www.figma.com/proto/9IWnooPSiyXxJBf3XpiMW8/Sem-t\%C3\%ADtulo?node-id=0-1&t=wuItbVwy1dUMzFoV-1`

taset JSON estruturado, o sistema evita alucinações sobre o conteúdo teórico. Essa escolha responde diretamente à limitação dos sistemas identificados na Seção 3, dado que o *Ruffle&Riley* [Schmucker et al. 2024] e o SPL [Zhang et al. 2024] empregam LLMs para interação textual sem uma camada determinística dedicada ao conhecimento do domínio. No Arandu, a criatividade generativa da IA é restrita ao eixo dos cenários e interfaces, enquanto o eixo do conhecimento heurístico permanece determinístico e auditável.

A especialização em três agentes com papéis distintos reflete o princípio de Mousavinasab et al. [2018] identificado nos módulos clássicos de STIs e responde a um argumento de viabilidade técnica, pois um único agente LLM generalista não poderia simultaneamente gerar código *JSX* tecnicamente correto, produzir explicações pedagógicas com analogias e calcular intervalos de revisão espaçada com lógica determinística. A economia de Açaí *Bits* e a Árvore de Habilidades operacionalizam os dois eixos da Teoria da Autodeterminação [Ryan and Deci 2000] no estilo gamificado.

A opção por um algoritmo *Python* puro para o Agente Estrategista responde a um requisito de determinismo, uma vez que a revisão espaçada exige cálculos reprodutíveis e auditáveis, incompatíveis com a natureza probabilística de LLMs. A viabilidade técnica da proposta é reforçada pela disponibilidade de todos os modelos candidatos como ferramentas *open-source* via *Hugging Face*, sem dependência de infraestrutura proprietária, tornando a proposta replicável em contexto acadêmico.

A principal limitação do trabalho é a ausência de avaliação empírica com usuários reais. Embora metodologicamente justificada pelo método *Informed Argument* [Hevner et al. 2004], a análise teórica não permite afirmar que o Arandu produzirá ganhos de aprendizado superiores aos de abordagens existentes. A validação empírica exige pré-teste, uso do sistema por período suficiente para ativar ao menos um ciclo de revisão espaçada, pós-teste com tarefas de AH em interfaces não treinadas e seguimento de semanas a meses para mensurar retenção a longo prazo, escopo inviável em estudos de curta duração típicos de disciplinas universitárias.

Três limitações adicionais merecem reconhecimento. Primeiro, mesmo guiado pelas regras do Módulo de Domínio, o Agente Arquiteto pode gerar interfaces com falhas visualmente ambíguas ou múltiplas violações simultâneas não intencionais; mitigar esse risco exigiria um agente verificador ou curadoria periódica por especialistas, nenhuma contemplada no escopo atual. Segundo, a eficácia do sistema é sensível à qualidade dos *prompts* dos Agentes Arquiteto e Mentor, e atualizações nos modelos candidatos podem alterar comportamentos esperados sem aviso, configurando limitação de manutenibilidade que demanda versionamento formal dos *prompts*. Terceiro, os intervalos de revisão espaçada de 7, 3 e 1 dia foram definidos heuristicamente com base na Curva do Esquecimento para conteúdo declarativo; a Avaliação Heurística, como habilidade procedimental, pode demandar calibração empírica desses parâmetros, possivelmente via algoritmos adaptativos como o *SM-2* utilizado por plataformas de revisões como o *Anki*.

8. Considerações Finais

Este trabalho apresentou o Arandu, um Sistema Tutor Inteligente gamificado com arquitetura multiagente e Inteligência Artificial Generativa para o ensino e a prática das Heurísticas de Nielsen em Interação Humano-Computador. A proposta foi fundamentada na *Design Science Research* e avaliada por análise teórica das decisões de arquitetura,

utilizando como demonstração um protótipo de alta fidelidade. O trabalho parte de um problema pedagógico documentado na literatura: o desenvolvimento da competência avaliativa em AH exige exposição repetida a interfaces variadas com falhas de usabilidade controladas, e nenhuma ferramenta existente reúne simultaneamente as cinco dimensões necessárias identificadas neste trabalho.

A contribuição central do Arandu reside no uso de LLMs para geração dinâmica de interfaces simuladas com falhas de usabilidade deliberadamente inseridas. Essa abordagem é viabilizada por uma arquitetura multiagente com especialização de responsabilidades entre geração de exercícios visuais, *feedback* pedagógico e controle algorítmico de progressão por revisão espaçada. As limitações do trabalho são reconhecidas, visto que a ausência de avaliação empírica impede afirmar superioridade do Arandu sobre abordagens existentes, enquanto a qualidade das interfaces geradas por IAG requer validação especializada e os intervalos de revisão espaçada demandam calibração com dados reais de desempenho em AH.

Como trabalho futuro, prevê-se a implementação técnica do sistema, adicionando um Agente Avaliador com a função de julgar o código gerado pelo Agente Arquiteto. Tal funcionalidade teria o fito de garantir a ausência de falhas de lógica do arquivo gerado e que apenas a falha heurística solicitada esteja presente na interface. Além disso, espera-se realizar um experimento controlado com estudantes de IHC, incluindo mensuração de ganho de aprendizado e retenção a longo prazo. Estima-se que o Arandu torne o ensino de AH mais acessível, escalável e eficaz na formação em Computação.

9. Declaração de Uso de Inteligência Artificial

Em conformidade com a Portaria CNPq Nº 2.664/2026, os autores declaram que ferramentas de Inteligência Artificial Generativa (IAG) não foram empregadas na concepção da ideia de pesquisa, no desenvolvimento do protótipo, na geração dos diagramas do ciclo de tutoria ou na definição da arquitetura do sistema Arandu. O uso de IAG restringiu-se à pesquisa e revisão da literatura (utilizando *NotebookLM* e *SciSpace*) e ao apoio e revisão gramatical e estrutural da escrita do artigo (utilizando *Claude* e *Deepseek*). Todo o conteúdo gerado ou sugerido por essas ferramentas foi integralmente revisado e validado pelos pesquisadores, que assumem total responsabilidade pela autoria intelectual, precisão e integridade do documento final, atestando a ausência de plágios ou imprecisões.

Referências

Aljedaani, W., Eler, M. M., Parthasarathy, P. D., Witherspoon, W., and Pamer, A. (2026). HeuristicBuilder: An interactive multimodal approach to teaching usability heuristics. In *Proceedings of the 57th ACM Technical Symposium on Computer Science Education (SIGCSE TS 2026)*, pages 17–23, New York, NY. ACM.

Dermeval, D., Albuquerque, J., Bittencourt, I. I., Isotani, S., Silva, A. P., and Vassileva, J. (2019). GaTO: An ontological model to apply gamification in intelligent tutoring systems. *Frontiers in Artificial Intelligence*, 2:13.

Deterding, S., Dixon, D., Khaled, R., and Nacke, L. (2011). From game design elements to gamefulness: Defining "gamification". In *Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments*, pages 9–15, New York, NY. ACM.

Hevner, A. R., March, S. T., Park, J., and Ram, S. (2004). Design science in information systems research. *MIS Quarterly*, 28(1):75–105.

Kang, J., Roestel, N. M. E., and Girouard, A. (2022). Experiential learning to teach user experience in higher education in past 20 years: A scoping review. *Frontiers in Computer Science*, 4:812907.

LearnLM Team and Eedi (2025). AI tutoring can safely and effectively support students: An exploratory RCT in UK classrooms.

Lima, A. L. A. L. and Nascimento, F. A. (2026). Tecnologias de memória na educação: desenvolvimento de uma aplicação alinhada à curva do esquecimento de Ebbinghaus. *Revista JRG de Estudos Acadêmicos*, 9(20).

Mousavinasab, E. S., Zarifsanaiey, N., Niakan Kalhori, S. R., Rakhshan, M., Keikha, L., Ghazisaeedi, M., and Saeedi, G. (2018). Intelligent tutoring systems: A systematic review of characteristics, applications, and evaluation methods. *Interactive Learning Environments*, 29(1):142–163.

Nielsen, J. (1994). *Usability Inspection Methods*. John Wiley & Sons, New York.

Peffers, K., Tuunanen, T., Rothenberger, M. A., and Chatterjee, S. (2007). A design science research methodology for information systems research. *Journal of Management Information Systems*, 24(3):45–77.

Rosson, M. B., Carroll, J. M., and Rodi, C. M. (2004). Case studies for teaching usability engineering. In *Proceedings of the 35th ACM Technical Symposium on Computer Science Education (SIGCSE 2004)*, pages 36–40, New York, NY. ACM.

Ryan, R. M. and Deci, E. L. (2000). Self-determination theory and the facilitation of intrinsic motivation, social development, and well-being. *American Psychologist*, 55(1):68–78.

Schmucker, R., Xia, M., Azaria, A., and Mitchell, T. (2024). Ruffle&Riley: Insights from designing and evaluating a large language model-based conversational tutoring system. In *Proceedings of the 25th International Conference on Artificial Intelligence in Education (AIED 2024)*, volume 14829 of *Lecture Notes in Computer Science*, pages 75–90, Cham. Springer.

Seaborn, K. and Fels, D. I. (2015). Gamification in theory and action: A survey. *International Journal of Human-Computer Studies*, 74:14–31.

Sobrino-Duque, R., Martínez-Rojo, N., Carrillo-de Gea, J. M., López-Jiménez, J. J., Nicolás, J., and Fernández-Alemán, J. L. (2022). Evaluating a gamification proposal for learning usability heuristics: Heureka. *International Journal of Human-Computer Studies*, 161:102774.

VanLehn, K. (2011). The relative effectiveness of human tutoring, intelligent tutoring systems, and other tutoring systems. *Educational Psychologist*, 46(3):197–221.

Xu, Z., Wijekumar, K. K., Ramirez, G., Hu, X., and Irey, R. (2019). The effectiveness of intelligent tutoring systems on K-12 students' reading comprehension: A meta-analysis. *British Journal of Educational Technology*, 50(6):3119–3137.

Zhang, L., Lin, J., Kuang, Z., Xu, S., and Hu, X. (2024). SPL: A socratic playground for learning powered by large language model.