

SUSTENTABILIDADE GAMIFICADA: DESENVOLVIMENTO DE UM APLICATIVO MÓVEL PARA PROMOVER A CONSCIENTIZAÇÃO AMBIENTAL

Eduardo Schmitt Carniel¹, Luciano Antunes²

Resumo: Este trabalho apresenta o Progresso, um aplicativo móvel gamificado para conscientização ambiental que integra diagnóstico inicial, recomendação e acompanhamento de hábitos, quizzes educativos (adaptativo e reconhecimento visual) e sistema de conquistas. O objetivo é empregar gamificação para promover mudança de comportamento sustentável, conectando interação lúdica com dados reais do usuário. Metodologicamente, adota-se uma pesquisa aplicada com desenvolvimento de um protótipo multiplataforma: frontend em React Native/Expo (TypeScript) e backend em Spring Boot (API REST e PostgreSQL), guiado por diretrizes de usabilidade (heurísticas de Nielsen) e testes funcionais via Postman. Os resultados indicam aderência da API aos requisitos (códigos HTTP corretos, serviços de backend estáveis) e boa conformidade às heurísticas (feedback imediato, consistência de navegação, prevenção/recuperação de erros). Discute-se que, frente a iniciativas correlatas (EkoPlay, Recycle Quiz, REHI, GoBeEco), o diferencial do Progresso está no monitoramento contínuo de hábitos e na personalização por perfil. Conclui-se que o protótipo é tecnicamente viável e coerente com a proposta.

Palavras-chave: gamificação; educação ambiental; aplicativo móvel; hábitos sustentáveis; usabilidade.

¹Curso de Ciência da Computação, Universidade do Extremo Sul Catarinense (Unesc), eduardoscarniel@unesc.net

²Orientador. Curso de Ciência da Computação, Universidade do Extremo Sul Catarinense (Unesc), luc@unesc.net

ABSTRACT: This work presents Progresso, a gamified mobile application for environmental awareness that integrates an initial assessment, habit recommendation and tracking, educational quizzes (both adaptive and visual recognition), and an achievement system. The goal is to employ gamification to promote sustainable behavior change by connecting playful interaction with real user data. Methodologically, it adopts applied research through the development of a cross-platform prototype: frontend in React Native/Expo (TypeScript) and backend in Spring Boot (REST API and PostgreSQL), guided by usability guidelines (Nielsen's heuristics) and functional testing via Postman. The results indicate that the API meets the requirements (correct HTTP codes, stable backend services) and shows good compliance with heuristics (immediate feedback, navigation consistency, error prevention/recovery), with areas for improvement identified in efficiency for advanced users and lack of contextual help. Compared to related initiatives (EkoPlay, Recycle Quiz, REHI, GoBeEco), Progresso stands out for its continuous habit monitoring and profile-based personalization. It is concluded that the prototype is technically feasible and aligned with its proposal.

Keywords: gamification; environmental education; mobile application; sustainable habits; usability.

1 INTRODUÇÃO

Uma pesquisa global do Fórum Econômico Mundial, revelou que 75% das pessoas ao redor do mundo acreditam que reciclar é importante, mas enfrentam barreiras para adotar as práticas, como falta de programas acessíveis e falta de confiança nos sistemas de reciclagem (Wood, 2021) (tradução própria).

Já em relação às mudanças climáticas, estudos apontam que o público, em geral, possui uma compreensão limitada e, muitas vezes, distorcida sobre o tema, o que pode levar à desinformação, à resistência em adotar medidas de mitigação e adaptação e à dificuldade de mobilização social e política. Essa compreensão varia de acordo com fatores como localização geográfica, condições socioeconômicas, nível de escolaridade e orientação política. Por isso, torna-se essencial abordar o assunto de forma interdisciplinar na educação, possibilitando uma compreensão mais ampla de diversos cenários climáticos (Zezzo; Coltri, 2022).

Segundo o Programa das Nações Unidas para o Meio Ambiente PNUMA (2024), mais de dois bilhões de toneladas de resíduos sólidos urbanos são gerados anualmente no mundo. Além desses, a atividade humana

produz diversos tipos de resíduos, cuja quantidade é diretamente influenciada pelos hábitos de consumo e descarte das pessoas. Assim, os resíduos urbanos estão profundamente conectados à tripla crise planetária: mudanças climáticas, poluição e perda de biodiversidade.

Existem algumas formas de resolver esses problemas relacionados ao impacto ambiental, entre elas, está a educação ambiental, que conforme disposto na Lei n.º 9.795/1999, compreende os processos pelos quais indivíduos e coletividades constroem valores, conhecimentos, habilidades e atitudes voltadas à conservação do meio ambiente (Brasil, 1999).

A tecnologia, por sua vez, surge como aliada nesse processo. Conforme Silva e Sousa (2020), a tecnologia, como opção facilitadora, tem aproveitado o mundo cada vez mais globalizado para expandir a variedade de aplicativos móveis voltados para usuários interessados no meio ambiente e suas questões.

No contexto de aplicações educacionais anteriores, buscou-se engajar públicos mais jovens por meio de aplicativos interativos e jogos, com abordagens lúdicas centradas em quizzes, pontuação progressiva e níveis de desafio. Essas experiências mostraram potencial para reforçar a aprendizagem ambiental entre crianças e adolescentes, ainda assim, costuma haver menos foco em acompanhamento contínuo de hábitos e em feedbacks adaptativos (Martins et al., 2024; Souza et al., 2023).

Martins et al. (2024) desenvolveu um aplicativo voltado à educação ambiental infantil, com foco em crianças de 5 a 13 anos. A proposta alia recursos lúdicos e tecnologias móveis para promover o aprendizado contínuo sobre sustentabilidade, partindo do mapeamento prévio do conhecimento das crianças e de suas expectativas quanto à gamificação.

Souza et al. (2023) propôs o *Recycle Quiz*, um aplicativo gamificado para ensinar reciclagem a crianças de 8 a 11 anos. Com quizzes interativos e sistema de pontuação, o app demonstrou potencial para estimular práticas sustentáveis, apesar de desafios como a resistência de professores e limitações orçamentárias.

Betancourt, Torres e Rozo (2024) investigou a gamificação no ensino superior com o jogo de tabuleiro *REHI*, voltado à educação ambiental crítica sobre o uso da água. Aplicado a estudantes universitários na Colômbia, o jogo gerou engajamento e mudanças de percepção sobre o uso consciente dos recursos hídricos.

Novo et al. (2024) apresentou o *GoBeEco*, uma aplicação web gamificada voltada a adultos, com desafios práticos ligados aos ODS e ma-

teriais pedagógicos para educadores. O projeto envolveu mais de 1.500 participantes e demonstrou eficácia ao transformar conscientização ambiental em ação prática.

Essas pesquisas confirmam a gamificação como uma abordagem eficaz para a educação ambiental (Martins et al., 2024; Souza et al., 2023; Betancourt; Torres; Roza, 2024; Novo et al., 2024). No entanto, em sua maioria, concentram-se na transmissão de conteúdos e no uso de mecânicas lúdicas pontuais, sem integrar acompanhamento contínuo e individual de hábitos nem feedbacks personalizados ao perfil do usuário. Além disso, raramente vinculam o aprendizado a rotinas mensuráveis do cotidiano, o que mantém a lacuna entre conscientização e ação que este trabalho busca enfrentar. Observa-se também a ausência de uma avaliação sistemática de usabilidade, o que reforça a pertinência de inspecionar a interface com base em heurísticas neste estudo.

Neste contexto, este trabalho busca preencher essa lacuna por meio do desenvolvimento de um aplicativo móvel multiplataforma com arquitetura gamificada orientada à mudança de comportamento, integrando quizzes e desafios a um módulo de registro e acompanhamento individual de hábitos, com feedbacks personalizados gerados a partir dos dados do próprio usuário; ao conectar a interação lúdica a métricas de uso e progresso, a proposta vincula a aprendizagem ambiental a práticas cotidianas mensuráveis.

Para o desenvolvimento deste trabalho tem como objetivo geral empregar gamificação no desenvolvimento de um aplicativo móvel para promover a conscientização ambiental dos usuários. Para atingir esse objetivo, têm-se como objetivos específicos: revisar a literatura sobre educação ambiental, sustentabilidade e impacto ambiental para fundamentar a proposta; identificar e adaptar mecânicas de gamificação adequadas ao contexto da educação ambiental; desenvolver um módulo para cálculo do impacto ambiental pessoal e posterior monitoramento de hábitos sustentáveis; integrar o monitoramento a um sistema de quizzes e recompensas simbólicas; e avaliar o funcionamento e a usabilidade do aplicativo.

O presente artigo está organizado da seguinte forma: a seção de Materiais e Métodos apresenta a prototipação da interface, a implementação da API e do aplicativo móvel, além dos testes funcionais realizados. Em Resultados e Discussão, são descritos os achados técnicos, a avaliação heurística de usabilidade e a comparação com trabalhos correlatos. Por fim, a seção de Conclusão sintetiza as principais contribuições, reconhece

limitações do estudo e aponta direções para pesquisas futuras.

2 MATERIAIS E MÉTODOS

Esta pesquisa é aplicada, de base tecnológica, com abordagem qualitativa e caráter exploratório-descritivo. Foi desenvolvido um aplicativo móvel com foco na conscientização ambiental e na formação de hábitos sustentáveis, utilizando gamificação como estratégia pedagógica. O sistema adota uma arquitetura cliente-servidor: o aplicativo, desenvolvido em React Native, concentra a interação com o usuário por meio de quizzes, insígnias e registro de hábitos; o backend, construído em Spring Boot, fornece uma API REST com autenticação por token e persistência dos dados em banco PostgreSQL.

O ambiente de desenvolvimento consistiu em um computador pessoal com arquitetura x64, 16 GB de RAM e sistema operacional Windows 10. Utilizou-se o IntelliJ IDEA para o backend, o Visual Studio Code para o frontend e o Android Studio como emulador. DBeaver e pgAdmin foram usados na administração do PostgreSQL. O fluxo de iteração no frontend ocorreu via Expo, com validações rápidas no navegador e testes finais no emulador. As rotas da API foram validadas com o Postman.

2.1 PROTOTIPAÇÃO DE INTERFACE

Realizou-se a prototipação no Figma, definindo arquitetura de navegação, hierarquia de informação e componentes de interface do usuário (UI). As telas principais (onboarding/diagnóstico, atividades, recomendação de hábitos e acompanhamento de progresso/conquistas) foram alinhadas a princípios de usabilidade, antecipando aderência às heurísticas de Nielsen que seriam verificadas na etapa de avaliação

2.1.1 Diretrizes de UI/UX

Para orientar o design antes mesmo da criação das primeiras telas, adotou-se como referência geral a abordagem das 10 Heurísticas de Usabilidade de Jakob Nielsen para análise e solução de problemas de interface. Utilizou-se a síntese atualizada pelo Nielsen Norman Group, bem como o capítulo Heuristic Evaluation, do livro Usability Inspection Methods (Wiley, 1994), que formaliza o método e sua aplicação prática. As diretrizes consideradas estão listadas na Tabela 1.

Tabela 1 - Heurísticas de usabilidade de Nielsen (10 princípios).

Princípio	Descrição
Visibilidade do status do sistema	O sistema deve manter os usuários informados sobre o que está acontecendo, com feedback apropriado e em tempo razoável.
Compatibilidade com o mundo real	Deve usar a linguagem dos usuários e seguir convenções do mundo real.
Controle e liberdade do usuário	Deve permitir desfazer/refazer ações e oferecer uma “saída de emergência”.
Consistência e padrões	Seguir convenções, evitando que o usuário precise adivinhar significados.
Prevenção de erros	Projetar de forma a evitar erros antes que eles ocorram.
Reconhecimento em vez de memorização	Tornar ações e opções visíveis para reduzir a carga de memória.
Flexibilidade e eficiência de uso	Aceleradores devem permitir eficiência para usuários experientes.
Design estético e minimalista	Remover informações irrelevantes que competem com as relevantes.
Ajudar na recuperação de erros	Mensagens claras, indicando o problema e como corrigi-lo.
Ajuda e documentação	Deve ser fácil de buscar, objetiva, e centrada na tarefa do usuário.

Fonte: Adaptado de Usability Inspection Methods (Nielsen; Mack, 1994; Nielsen, 1994).

2.1.2 Prototipação

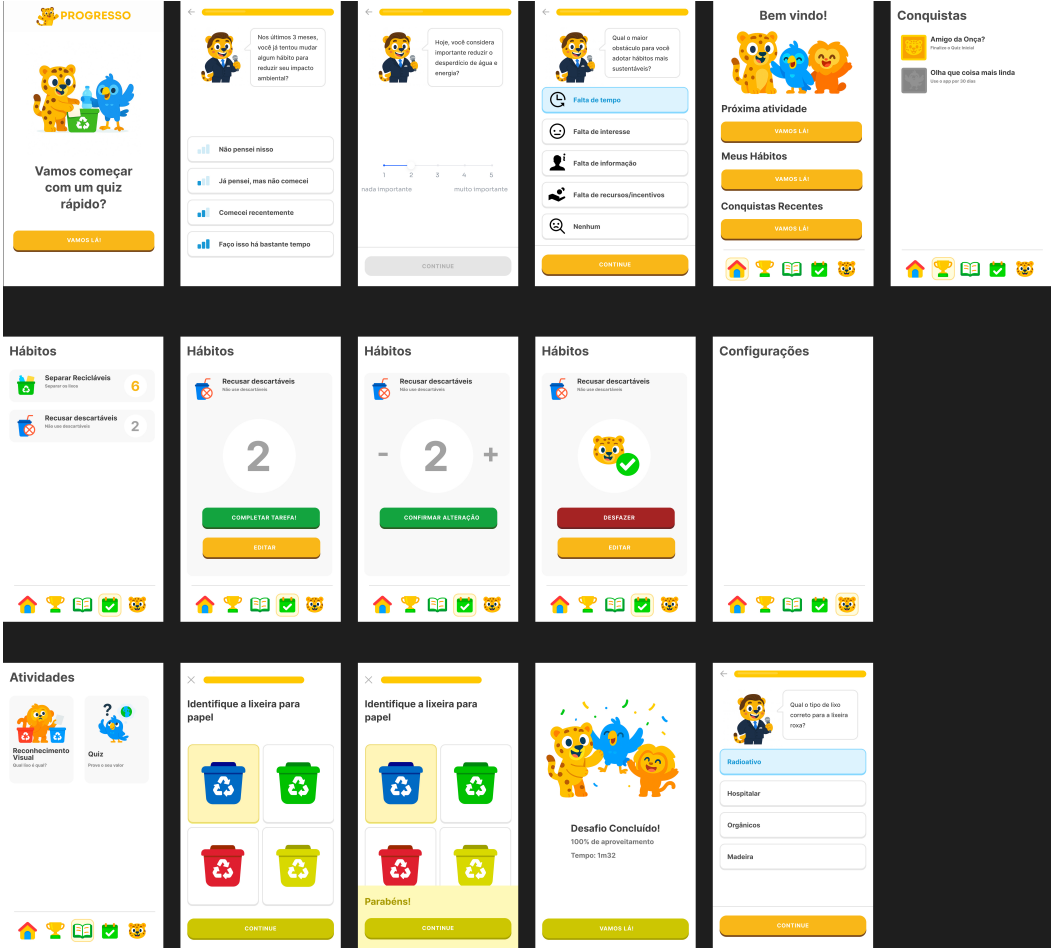
Com o objetivo de antecipar decisões de navegação antes da implementação em código, foi desenvolvido um protótipo inicial no Figma, com quadros organizados lado a lado para simular os fluxos principais do aplicativo: diagnóstico, recomendações de hábitos, conquistas e exercícios. Esse protótipo serviu como artefato de validação preliminar das diretrizes de usabilidade adotadas, com base nas heurísticas de Nielsen.

A partir desse protótipo, também foram definidas as diretrizes visuais e a identidade do aplicativo. O nome Progresso foi escolhido por remeter à frase presente na bandeira do Brasil, transmitindo a ideia de avanço, movimento contínuo e transformação positiva, conceitos alinhados à proposta de incentivar hábitos sustentáveis. As mascotes (onça-pintada, arara-azul e mico-leão-dourado), todas da fauna nacional, reforçam a identidade local e apoiam o caráter lúdico do aplicativo.

Para traduzir esse conceito de forma acessível e motivadora, o design do aplicativo foi inspirado no Duolingo, uma das principais referências globais em gamificação e aprendizagem com interfaces simples, coloridas e envolventes. Elementos como mascotes ilustrativos, linguagem lúdica e a sensação de progressão recompensadora foram adaptados ao contexto da sustentabilidade.

Buscando consistência visual e agilidade no desenvolvimento, foi utilizado um kit de interface do usuário (UI) gratuito do Figma, baseado no Duolingo e ajustado às necessidades do projeto. As ilustrações e ícones personalizados, incluindo os mascotes do aplicativo, foram gerados com o ChatGPT Pro, permitindo criar imagens originais em um estilo compatível com o restante da interface. A Figura 1 apresenta uma visão geral do fluxo resultante.

Figura1 - Protótipo Inicial no Figma: Visão Geral do Fluxo de Navegação



Fonte: Elaborado pelo autor.

A jornada do usuário foi desenhada para ser direta: após o primeiro acesso, o app conduz a um diagnóstico inicial com perguntas curtas

que geram um perfil ambiental. Em seguida, a tela principal apresenta as seções de Hábitos, Conquistas, Atividades e Configurações; na seção de Hábitos, o sistema exibe sugestões personalizadas para ativação, possibilita a criação de novos hábitos e a gestão do progresso. Quizzes temáticos alinhados ao diagnóstico reforçam o aprendizado e alimentam o mural de conquistas, formando um ciclo de diagnóstico, prática, recompensa e evolução.

As decisões de navegação seguiram heurísticas de usabilidade. A barra inferior fixa com ícones e rótulos dá consistência e reduz carga cognitiva; fluxos curtos com uma ação por tela mantêm a interface objetiva; feedback imediato aparece em checks, mensagens e insígnias; o botão “Desfazer” garante controle e correção de erros. Elementos como botões grandes, áreas de toque amplas, contraste adequado e linguagem clara atendem à acessibilidade e ampliam a compreensão em diferentes perfis de usuário.

2.2 MODELAGEM DE DADOS

O banco de dados do aplicativo Progreso foi desenvolvido em PostgreSQL, versão 17, com a modelagem e criação realizadas por meio do PgAdmin 4. Para apoio visual e organização do diagrama entidade-relacionamento (ERD), também foi utilizado o DBeaver. Ao todo, foram criadas 18 tabelas, sendo duas delas destinadas à infraestrutura do backend: *flyway schema history*, para controle de migrações, e *tokens login*, para gerenciamento de sessões de autenticação.

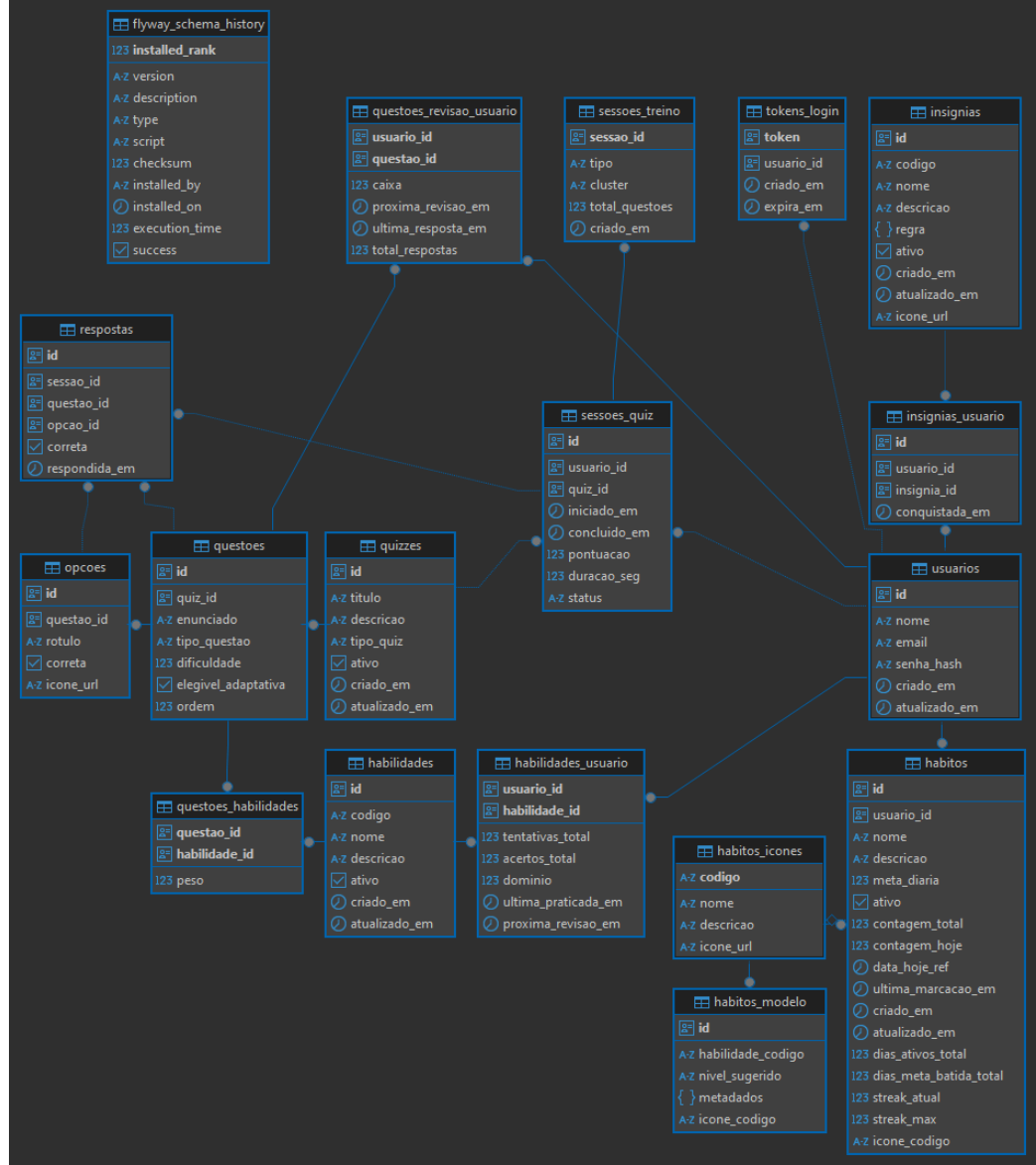
2.2.1 Modelo Conceitual

O modelo conceitual é uma visão de alto nível que descreve as entidades do domínio e seus relacionamentos, independente de tecnologia, orientando a construção dos modelos lógico e físico. O módulo de hábitos é sustentado por *habitos*, que registra os hábitos ativos dos usuários, *habitos_modelo*, que armazena sugestões pré-configuradas, e *habitos_ícones*, que concentra os ícones reutilizáveis pelas duas anteriores. A lógica de gamificação está presente nas tabelas *insignias*, que define as conquistas possíveis, e *insignias_usuario*, que registra os desbloqueios realizados por cada usuário.

O módulo de avaliação por quizzes reúne as tabelas *quizzes*, *questoes* e *opcoes*, que definem cada teste, além de *respostas* e *sesoes_quiz*, que registram o histórico de participação. A tabela auxiliar *ques-*

toes_habilidades relaciona cada pergunta às competências avaliadas. No eixo de habilidades, *habilidades* armazena as competências disponíveis, enquanto *habilidades_usuario* registra o progresso de cada participante. A prática adaptativa é organizada por *sessoes_treino* e pela tabela complementar *questoes_revisao_usuario*, que aplicam lógica de revisão espaçada com base no desempenho. No conjunto, essa estrutura relacional integra autenticação, hábito, aprendizagem e gamificação, garantindo coerência entre os dados e as funcionalidades do aplicativo. A Figura 2 apresenta o modelo conceitual do banco de dados, com os principais agrupamentos de tabelas e suas relações.

Figura 2 - Modelo conceitual do banco de dados



Fonte: Elaborado pelo autor.

2.3 IMPLEMENTAÇÃO DA API

A API foi desenvolvida em Java 21 com Spring Boot 3.5.5, uma ferramenta que simplifica a criação de aplicações web e microserviços ao automatizar boa parte da configuração necessária. Baseado no Spring Framework, amplamente utilizado em ambientes corporativos, o Spring Boot permite o desenvolvimento de sistemas independentes e prontos para produção, com mínima configuração manual. Entre seus principais recursos estão a autoconfiguração, uma abordagem opinativa para definições padrão e a facilidade de execução direta, sem necessidade de servidores externos ou setups complexos (IBM, 2025).

2.3.1 Arquitetura e Módulos

A aplicação adota uma arquitetura em camadas no padrão Controller, Service e Repository. Os controladores expõem os endpoints HTTP, a camada de serviço concentra as regras de negócio e os repositórios, baseados em JpaRepository, realizam a persistência. As entidades do domínio são modeladas com @Entity, e a comunicação externa é feita por meio de DTOs. O tratamento de erros é centralizado com @ControllerAdvice, e configurações transversais como CORS, serialização e segurança são organizadas em classes com @Configuration. O controle de versão do banco de dados é feito com o Flyway, que aplica migrations versionadas para criar ou alterar tabelas, índices e chaves, garantindo mudanças seguras e reproduzíveis.

Para a implementação da API, foram utilizados módulos essenciais do Spring Boot para dados, validação, segurança e migração do banco. A autenticação foi adicionada na transição do protótipo para o código, garantindo acesso individualizado. A segurança é *stateless*, baseada em tokens Bearer validados a cada requisição, protegendo as rotas privadas e mantendo abertas apenas as operações de registro e login.

2.3.2 Sincronização de Dados Diários

No módulo de hábitos, o sistema reconhece automaticamente a virada de dia para preservar a integridade das contagens diárias. Como o foco do projeto está nos exercícios e no diagnóstico adaptativo, essa funcionalidade foi implementada de forma mais simples, sem armazenar um histórico das realizações dos hábitos.

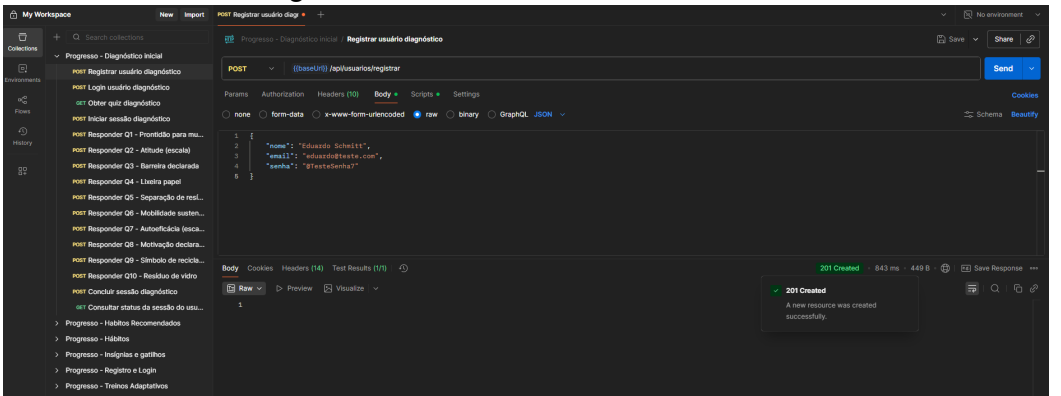
Para cada registro, o sistema utiliza a variável *contagem_hoje*, responsável por armazenar o número de execuções no dia atual. Como hoje o sistema permite uma contagem mais livre, essa variável é zerada

automaticamente à meia-noite para evitar inconsistências no acompanhamento de streaks e nas metas diárias. Essa abordagem simplifica o modelo de dados e mantém o desempenho alinhado aos objetivos da versão atual.

2.4 TESTES FUNCIONAIS

Todos os fluxos expostos pela API foram exercitados manualmente no Postman para verificar códigos HTTP, *payloads* e efeitos colaterais previstos. A coleção principal utiliza variáveis para *baseUrl*, credenciais padrão e *Bearer token*, permitindo repetir cenários sem refazer o login a cada execução. Antes de iniciar os ensaios, o ambiente foi configurado para a instância correta do backend e o token foi obtido pelo fluxo de autenticação. Os testes cobriram caminhos felizes e cenários de erro recorrentes (identificadores inexistentes, corpo de requisição inválido e ausência de autenticação), reduzindo o risco de regressões nos contratos expostos à aplicação cliente. A Figura 3 ilustra o ensaio da rota de registro de usuário, retornando *201 Created* e confirmando a criação do recurso.

Figura 3 - Teste funcional no Postman



Fonte: Elaborado pelo autor.

2.5 IMPLEMENTAÇÃO DO APLICATIVO MÓVEL

O aplicativo foi desenvolvido com *React Native*, um framework mantido pelo Meta Open Source para criação de interfaces nativas a partir de componentes declarativos baseados em JavaScript. Utilizou-se o ecossistema *Expo* para acelerar o ciclo de desenvolvimento e facilitar testes em múltiplos dispositivos. Todo o código-fonte foi escrito em *TypeScript*, aproveitando os recursos de tipagem estática para maior robustez e legibilidade. (Meta Open Source, 2025)

Ao iniciar o aplicativo, o usuário se depara com a tela de autenticação, onde são exibidas as opções Cadastrar e Entrar. Caso o usuário ainda não possua uma conta, ao se cadastrar, ele é automaticamente dire-

cionado para o quiz diagnóstico inicial (Figura 4).

O diagnóstico inicial é um questionário cujo objetivo é estimar o nível de domínio do usuário em diferentes habilidades relacionadas ao aprendizado. Cada resposta gera um score entre 0 e 1, de acordo com o tipo da pergunta:

- Questões de **certo ou errado** atribuem 1 ponto para respostas corretas e 0 para incorretas;
- Questões de **escala** (por exemplo, de 1 a 5) são normalizadas para um valor entre 0 e 1;
- Questões de **múltipla escolha** utilizam uma tabela de conversão pré-definida para transformar cada alternativa em um score numérico.

Caso uma questão esteja vinculada a mais de uma habilidade, o score da resposta é proporcionalmente distribuído entre todas elas. Cada habilidade acumula tanto o somatório dos scores recebidos quanto a quantidade de questões (ou pesos) associadas a ela.

As perguntas do diagnóstico podem ser de diferentes tipos, incluindo questões tradicionais e questões de associação visual. Todas são previamente cadastradas no banco de dados e vinculadas ao tipo de quiz “Diagnóstico Inicial”. Além disso, cada questão está associada a pelo menos uma habilidade, o que permite que o sistema estime o nível do usuário em áreas específicas.

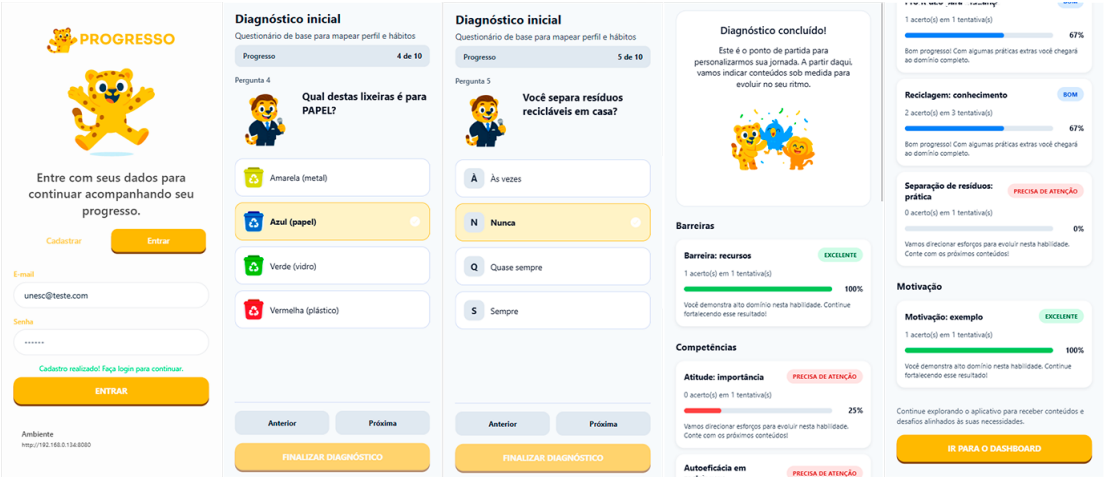
As habilidades estão organizadas em três categorias principais: competências, barreiras e motivação. Essa estrutura foi projetada para ser flexível: novas perguntas ou habilidades podem ser cadastradas no banco de dados sem a necessidade de modificar o código-fonte da aplicação.

Ao final do diagnóstico, o sistema calcula o domínio estimado inicial de cada habilidade como a média ponderada dos scores (ou seja, a soma dos scores dividida pela soma dos pesos), sempre limitada ao intervalo entre 0 e 1. Além disso, são armazenados dados sobre tentativas e acertos, possibilitando análises posteriores.

As perguntas relacionadas a barreiras e motivação funcionam de forma semelhante a um checklist. A seleção de uma alternativa implica que a habilidade correspondente será considerada com domínio de 100%, enquanto as habilidades da mesma categoria que não forem selecionadas serão atribuídas como não dominadas.

Por fim, como ilustrado na Figura 4, o aplicativo exibe ao usuário, em formato percentual, o nível de domínio em cada competência, permitindo o acesso ao dashboard para acompanhar seu progresso.

Figura 4 - Conjunto de telas do diagnóstico inicial



Fonte: Elaborado pelo autor.

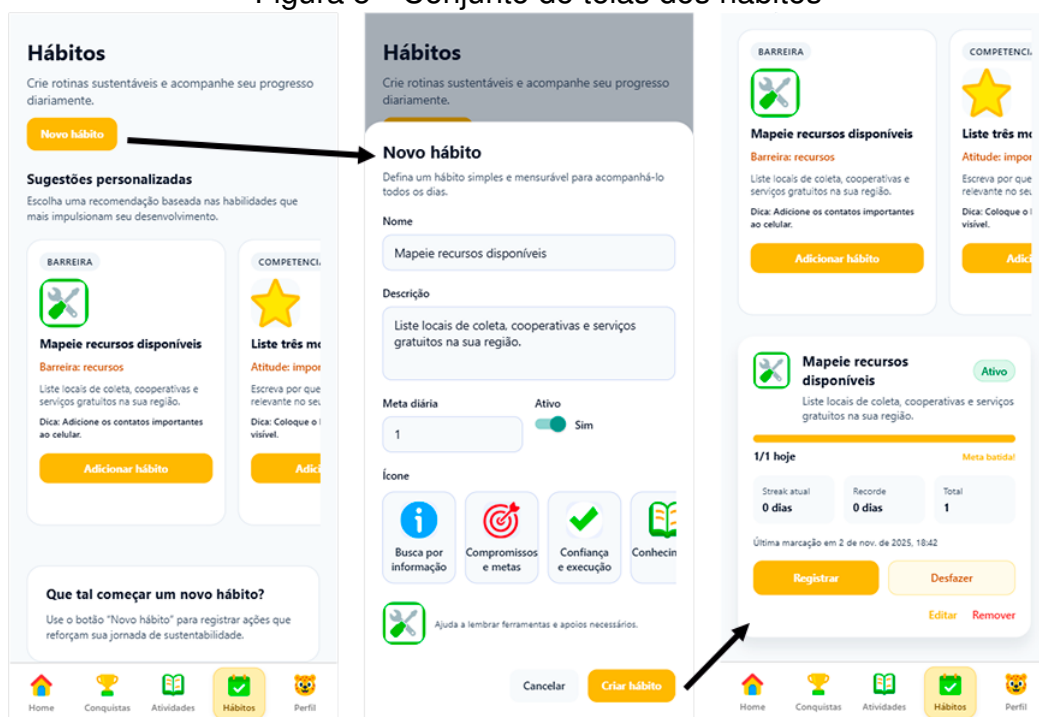
O sistema oferece sugestões personalizadas de hábitos com base nas habilidades do usuário registradas na tabela de habilidades do usuário. A API responsável pela recomendação classifica cada habilidade em uma das três categorias fixas: barreira, competência ou motivação, de acordo com listas pré-definidas de códigos.

Para habilidades do tipo competência, o domínio estimado é utilizado para definir o nível da recomendação: valores abaixo de 34% correspondem ao nível iniciante, entre 34% e 66% indicam o nível intermediário, e valores iguais ou superiores a 67% representam o nível avançado. Já as habilidades classificadas como barreira ou motivação utilizam sempre o nível padrão (default), independentemente do domínio estimado.

Com base nessa categorização, o sistema consulta a tabela de exemplos de hábitos, buscando registros que combinem o código da habilidade com o nível calculado. Os hábitos retornados incluem texto explicativo, ícones e uma categorização clara para que o usuário possa compreender facilmente o objetivo da ação sugerida. Para otimizar a experiência, os resultados são mantidos em cache durante a montagem da interface.

A Figura 5 ilustra esse fluxo: primeiro, são exibidas sugestões de hábitos recomendados com base nas habilidades do usuário. O usuário pode selecionar um hábito sugerido ou optar por criar um novo hábito manualmente, definindo nome, descrição, meta diária, ícone e ativação.

Figura 5 - Conjunto de telas dos hábitos



Fonte: Elaborado pelo autor.

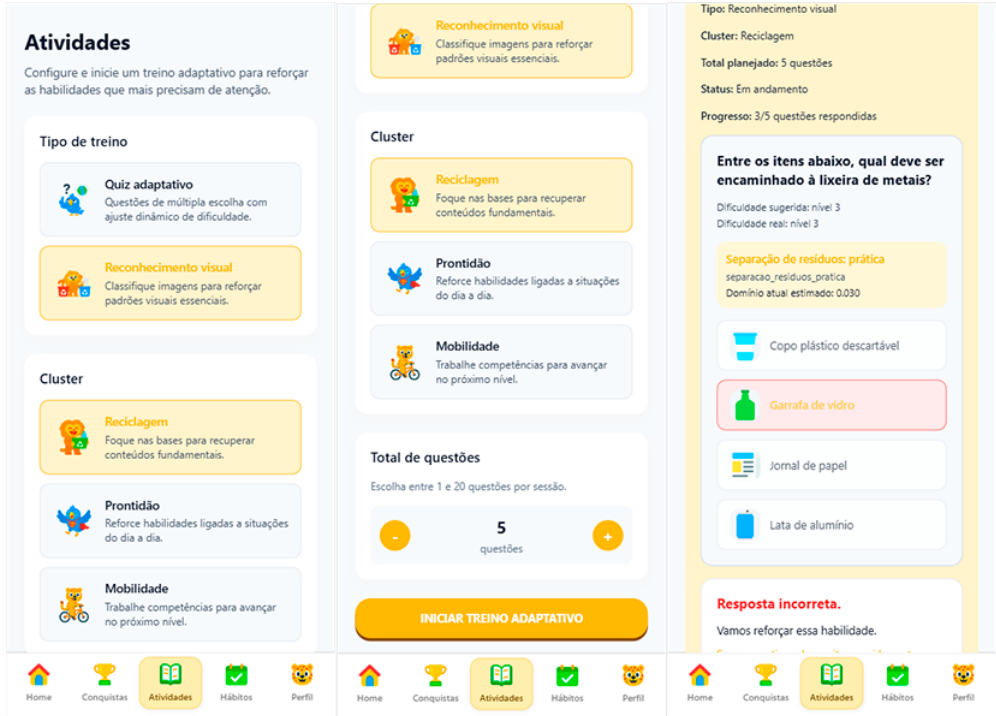
Após a criação, o hábito passa a ser exibido na lista principal, com opções para registrar a conclusão diária, editar, remover ou desfazer marcações. O sistema acompanha o progresso do hábito, indicando estatísticas como streak atual, recorde e total de registros.

O sistema de atividades conta com dois tipos de quizzes educativos: o quiz adaptativo, composto por questões de múltipla escolha com ajuste dinâmico de dificuldade, e o reconhecimento visual, que propõe a classificação de imagens para reforçar padrões visuais essenciais. Ambos são considerados quizzes do tipo educativo e se diferenciam do quiz diagnóstico inicial, que serve exclusivamente para avaliar o domínio inicial das habilidades. Cada questão cadastrada pode ser associada a um tipo específico de quiz e a um nível de dificuldade, variando entre 1 e 4.

Durante a execução dos quizzes, o sistema ajusta o domínio da habilidade vinculada com base no desempenho do usuário. O algoritmo de atualização aplica incrementos ou decrementos conforme a dificuldade da questão: para dificuldade 4 ou superior, o domínio é ajustado em +0,05 em caso de acerto e 0,04 em caso de erro; para dificuldade 3, os valores são +0,03 e 0,02, respectivamente; e para dificuldades 1 ou 2, os ajustes são +0,02 e 0,01. Esses valores são somados ao domínio atual da habilidade na tabela de habilidades do usuário.

O domínio inicial de cada habilidade é definido a partir do resultado do quiz diagnóstico. Caso o usuário não tenha realizado o diagnóstico, o sistema adota um valor padrão de 0,50 como ponto de partida. A Figura 6 mostra a seleção de tipo de treino e cluster, bem como a execução de uma sessão de reconhecimento visual no cluster “Reciclagem”, com feedback imediato após a resposta do usuário.

Figura 6 - Conjunto de telas das atividades



Fonte: Elaborado pelo autor.

O aplicativo também conta com uma seção dedicada às conquistas, criada para reforçar o engajamento do usuário. As conquistas foram planejadas tanto para hábitos quanto para quizzes, e são liberadas automaticamente pelo sistema a partir de verificações em rotas específicas. Ao atingir as condições definidas, a conquista é desbloqueada e exibida na tela para o usuário, funcionando como incentivo ao progresso contínuo.

A implementação do aplicativo foi pensada para ser modular, flexível e orientada a dados. As funcionalidades de cadastro, quizzes, hábitos e atividades foram desenvolvidas com base em modelos parametrizáveis, permitindo adaptar o conteúdo conforme o progresso do usuário. Isso garante escalabilidade e facilita a inclusão de novos recursos, mantendo o aplicativo dinâmico e personalizado.

2.6 GAMIFICAÇÃO

A gamificação busca promover mudanças de comportamento, desenvolver habilidades e estimular a inovação, permitindo que os participantes alcancem suas metas e, como consequência, que organizações também atinjam as suas (Burke, 2015).

Jogos educacionais em formato de quiz podem aumentar o envolvimento, apoiar a retenção de informações e estimular a busca ativa por conhecimento, ao combinar desafio e diversão (Souza et al., 2023). De forma mais ampla, experiências com jogos bem elaborados favorecem a assimilação de conteúdos e o desenvolvimento de competências, inclusive quando títulos de entretenimento são inseridos em contextos pedagógicos, sobretudo aqueles de maior complexidade que demandam estratégias, colaboração e tempo de domínio (Studart, 2022).

No Progresso, a gamificação aparece de maneira objetiva: símbolos de conquista (insígnias e marcos), mecânicas de memória (recordar padrões, regras ou dicas nos quizzes) e feedback imediato após cada ação. Esses elementos funcionam como reforçadores do comportamento desejado e manutenção do interesse do usuário (Hoffmann, 2023).

3 RESULTADOS E DISCUSSÃO

Nos testes funcionais realizados com a API REST do Progresso, todos os endpoints responderam conforme o esperado, retornando os códigos HTTP corretos e os dados previstos. A criação de usuários foi validada por chamadas ao endpoint de cadastro, que retornaram 201 (Created) com o recurso gerado. Cenários de erro também foram testados manualmente via Postman, como requisições sem token, que retornaram 403 (Forbidden), e exclusões validadas pelo retorno 204 (No Content). Esses testes confirmam a aderência aos requisitos funcionais e à arquitetura proposta, com separação entre controladores, serviços e repositórios, além da segurança por token. A integração entre o aplicativo móvel (React Native) e a API (Spring Boot) se mostrou estável e coerente com os fluxos definidos.

A avaliação de usabilidade baseada nas 10 heurísticas de Nielsen revelou que o sistema Progresso atende em grande medida aos princípios clássicos de interação, embora tenha evidenciado alguns pontos passíveis de melhoria. Conforme sintetizado na Tabela 2, a maioria dos princípios heurísticos foi plenamente atendida pelo sistema, com poucas violações ou inconsistências observadas.

Tabela 2 - Heurísticas de Nielsen e suas implementações no aplicativo Progresso.

Heurística	Implementação no Progresso
Visibilidade do status do sistema	Exibe status e progresso em tempo real (feedback visual imediato ao usuário).
Correspondência entre o sistema e o mundo real	Usa linguagem simples e termos familiares ao usuário, alinhados ao seu cotidiano.
Controle e liberdade para o usuário	Disponibiliza função “Desfazer” para reverter ações e corrigir enganos rapidamente.
Consistência e padronização	Adota design consistente: barra de navegação inferior fixa e ícones padronizados em todas as telas.
Prevenção de erros	Fluxos curtos e opções claras minimizam ambiguidades, prevenindo erros de uso.
Reconhecimento em vez de memorização	Mantém opções visíveis (menu fixo com ícones e rótulos), reduzindo a necessidade de memorizar comandos.
Flexibilidade e eficiência de uso	Fluxos ágeis e design responsivo garantem eficiência para diversos perfis de usuário.
Estética e design minimalista	Apresenta interface limpa e minimalista, exibindo apenas informações essenciais em cada tela.
Ajudar usuários a reconhecer, diagnosticar e recuperar-se de erros	Em caso de erro, exibe mensagens claras e destaca o problema para facilitar a correção (ex.: campo obrigatório em destaque).
Ajuda e documentação	Não oferece ajuda ou documentação interna, apostando na intuitividade da interface.

Fonte: Adaptado de Usability Inspection Methods (Nielsen; Mack, 1994; Nielsen, 1994).

Por sua vez, a Tabela 3 apresenta os principais problemas de usabilidade identificados durante a avaliação, relacionando cada problema à heurística de Nielsen correspondente e ao seu nível de severidade. Essas tabelas evidenciam que, apesar de o sistema demonstrar boa aderência geral às heurísticas de usabilidade, existem áreas específicas que requerem ajustes para aprimorar ainda mais a experiência do usuário.

Tabela 3 - Avaliação heurística com base nas 10 heurísticas de usabilidade e na escala de severidade de Nielsen (0-4).

Heurística	Observação	Sev.
Visibilidade do status do sistema	Feedback visual imediato após ações do usuário (ex.: ícone de check verde confirma conclusão).	0
Correspondência entre o sistema e o mundo real	Textos e opções utilizam linguagem clara e familiar, sem jargões técnicos.	0
Controle e liberdade para o usuário	Botão “Desfazer” permite reverter ações equivocadas (corrige check-in indevido, por exemplo).	0
Consistência e padronização	Design consistente entre telas; barra de navegação fixa e ícones uniformes em toda a interface.	0
Prevenção de erros	Opções bem definidas e fluxos simples evitam equívocos do usuário durante as tarefas.	0
Reconhecimento em vez de memorização	Ícones com rótulos e menu permanente facilitam o reconhecimento das funções, dispensando memorização.	0
Flexibilidade e eficiência de uso	Não há recursos de atalho ou modos avançados para usuários experientes (fluxo igual para todos os perfis).	2
Estética e design minimalista	Interface minimalista foca no essencial, sem poluição visual nas telas.	0
Ajudar usuários a reconhecer, diagnosticar e recuperar-se de erros	Campos inválidos são destacados e mensagens de erro objetivas orientam na correção imediata.	0
Ajuda e documentação	Ausência de ajuda contextual ou manual do usuário no aplicativo.	2

Fonte: Adaptado de Usability Inspection Methods (Nielsen; Mack, 1994; Nielsen, 1994).

Optou-se por adiar a implementação de atalhos avançados e de uma documentação extensa, priorizando a simplicidade de uso e a baixa carga cognitiva no onboarding. De acordo com as diretrizes da Nielsen Norman Group, aceleradores devem beneficiar usuários experientes sem comprometer a experiência dos iniciantes, sendo mais apropriado introduzi-los gradualmente conforme a familiaridade com o sistema evolui (Laubheimer, 2020).

Com o objetivo de situar o Progresso no contexto de pesquisas

recentes, realizou-se uma análise comparativa com aplicações de gamificação ambiental, como EkoPlay, Recycle Quiz, REHI e GoBeEco. Esses estudos oferecem contribuições relevantes para o engajamento em práticas sustentáveis, mas, em geral, carecem de mecanismos de acompanhamento contínuo e de personalização do conteúdo ao perfil do usuário.

O diferencial do GoBeEco (Novo et al., 2024) está na validação em múltiplas etapas, com questionários e grupos focais que demonstraram a conversão de conscientização em ação e identificaram preferências dos usuários por estatísticas, exemplos práticos, elementos lúdicos e tarefas diárias. Computacionalmente, trata-se de uma aplicação web com missões, desafios e tarefas, gamificação inspirada no framework Octalysis e camada social. Inclui ainda um handbook para educadores. Em contraste, o Progresso tem foco desde o início em personalização e acompanhamento contínuo. A principal lição metodológica a ser incorporada é o uso de um modelo de avaliação pré e pós com abordagem qualitativa, apoiado nas métricas já disponíveis, como adesão a hábitos, streaks e taxa de retorno.

Recycle Quiz (Souza et al., 2023) apresenta um app educacional em JavaScript/React Native, com portabilidade Android/iOS. O conteúdo é organizado em quatro níveis temáticos, com pontuação fixa, metas progressivas e feedback imediato, funcionando como ferramenta de alfabetização ambiental via quiz. A interface é simples, colorida e voltada ao público infantil, mas o estudo não inclui avaliação formal de usabilidade. Na comparação, o Progresso atua em uma camada distinta, com personalização e monitoramento contínuo, complementando o escopo inicial do Recycle Quiz com foco em uso prolongado e persistência.

O REHI (Betancourt; Torres; Rozo, 2024) é um jogo de tabuleiro presencial, estruturado com base no framework MDA e desenvolvido por meio de pesquisa orientada por design. As mecânicas envolvem missões, papéis em equipe, eventos e tempo limitado de jogo, com apoio de materiais físicos como tabuleiro, cartas e atividades. A avaliação foi feita com 106 estudantes usando um instrumento com 30 variáveis, indicando ganhos em comunicação, motivação e compreensão sobre pegada hídrica. Para o Progresso, o REHI oferece um roteiro temático e um instrumento avaliativo adaptável, que pode ser aplicado em testes piloto para medir impactos no uso individual e contínuo.

EkoPlay (Martins et al., 2024) combina métodos quali-quantitativos com 70 crianças e 7 gestores, mapeando lacunas conceituais e barreiras de adoção escolar. O app traz quizzes por dificuldade, dicas, ecopontos,

notícias e canais de contato. Utiliza React Native com Expo, prototipação em Figma e fluxo colaborativo com GitHub. A gamificação é centrada em questionários com progressão por níveis. Embora acessível ao público infantil, o estudo aponta limitações como resistência institucional e orçamento restrito. A contribuição prática para o Progresso é incorporar materiais de apoio a docentes, avaliando como personalização e acompanhamento contínuo influenciam adesão e persistência.

4 CONCLUSÃO

Este trabalho teve como objetivo empregar a gamificação no desenvolvimento de um aplicativo móvel voltado à promoção da conscientização ambiental dos usuários. O protótipo resultante, composto por diagnóstico inicial, quizzes (incluindo o reconhecimento visual), módulo de hábitos com acompanhamento diário, sistema de conquistas e backend estruturado em arquitetura REST, demonstrou-se funcional e coerente com a proposta metodológica e de engenharia de software adotada.

Do ponto de vista técnico e de design, destacam-se três contribuições principais: (i) a arquitetura modular e orientada a dados, que permite a evolução do conteúdo sem necessidade de alterar o código-fonte; (ii) a combinação entre jogos educativos e registro de hábitos, criando um ciclo contínuo de diagnóstico, prática, feedback e recompensa; e (iii) a aderência às heurísticas de Nielsen, com foco em feedback imediato, consistência de navegação e prevenção ou recuperação de erros. Em conjunto, esses elementos diferenciam o Progresso de abordagens restritas a quizzes, ao incorporar o monitoramento contínuo e recomendações personalizadas como mecanismos de apoio à mudança comportamental.

A principal limitação do estudo é a ausência de testes com usuários finais, o que impossibilita, por ora, avaliar aspectos como atratividade, engajamento sustentado e impacto comportamental em contextos reais. Diante disto, propõe-se como trabalhos futuros: (i) a realização de estudos piloto controlados, com métricas de uso, satisfação e aprendizagem; (ii) a expansão das camadas de feedback sensorial, incluindo animações e efeitos sonoros para reforço motivacional; e (iii) o desenvolvimento de um ambiente administrativo para cadastro e curadoria de questões, habilidades e hábitos, permitindo maior escalabilidade e adaptações regionais ou educacionais. Essas frentes poderão ampliar a validade externa da aplicação e favorecer sua adoção em contextos formais e informais de ensino.

REFERÊNCIAS

BETANCOURT, C. M.; TORRES, P. P.; ROZO, Y. Y. P. **Gamificación como estrategia para incluir la educación ambiental en el contexto universitario: caso rehi**. Revista científica, 2024, scieloco. ISSN 0124-2253. Disponível em: <http://www.scielo.org.co/scielo.php?script=sci_arttext&pid=S0124-22532024000100013&nrm=iso>. Acesso em: 21 set. 2024.

Brasil. **Lei n.º 9.795, de 27 de abril de 1999**. 1999. <http://www.planalto.gov.br/ccivil_03/leis/19795.htm>. Disponível em: <http://www.planalto.gov.br/ccivil_03/leis/19795.htm>. Acesso em: 21 set. 2024.

BURKE, B. **Gamificar: Como a gamificação motiva as pessoas a fazerem coisas extraordinárias**. 1. ed. São Paulo: DVS Editora, 2015. Tradução de Sieben Gruppe. ISBN 978-85-8289-107-0.

HOFFMANN, S. A. **Introdução à Gamificação na Educação: como aplicar os elementos do jogos em contextos de aprendizagem**. [S.l.]: Independente, 2023. 459 p. EBook Kindle, ASIN: B0CHGCRZ4C. Disponível em: <<https://www.amazon.com.br/dp/B0CHGCRZ4C>>. Acesso em: 9 out. 2024.

IBM. **What is Java Spring Boot?** 2025. Disponível em: <<https://www.ibm.com/br-pt/think/topics/java-spring-boot>>. Acesso em: 25 out. 2025.

LAUBHEIMER, P. **Flexibility and Efficiency of Use (Usability Heuristic #7)**. 2020. Disponível em: <<https://www.nngroup.com/articles/flexibility-efficiency-heuristic/>>.

MARTINS, D. et al. **Construindo consciência ambiental desde a infância: um estudo sobre sustentabilidade e gamificação**. 2024. ISSN 2525-8729.

Meta Open Source. **React Native Documentation: Getting Started**. [S.l.], 2025. Disponível em: <<https://reactnative.dev/docs/getting-started>>. Acesso em: 5 nov. 2025.

NIELSEN, J. **10 Usability Heuristics for User Interface Design**. 1994. Atualizado em 2024-01-30. Disponível em: <<https://www.nngroup.com/articles/ten-usability-heuristics/>>.

NIELSEN, J.; MACK, R. L. **Usability Inspection Methods**. New York: John Wiley & Sons, 1994.

NOVO, C. et al. **The use of gamification and web-based apps for sustainability education**. Sustainability, 2024, v. 16, n. 8, p. 3197. Open Access Article. Disponível em: <<https://doi.org/10.3390/su16083197>>. Acesso em: 12 abr. 2025.

Programa das Nações Unidas para o Meio Ambiente PNUMA. **Global Waste Management Outlook 2024**. 2024. Disponível em: <<https://www.unep.org/resources/global-waste-management-outlook-2024>>. Acesso em: 21 set. 2024.

SILVA, G. F. da; SOUSA, S. M. de. **Tecnologia e meio ambiente: levantamento de aplicativos móveis voltados a temas ambientais**. Brazilian Journal of Development, 2020, v. 6, n. 6, p. 42060–42071. Disponível em: <<https://ojs.brazilianjournals.com.br/ojs/index.php/BRJD/article/view/16564>>. Acesso em: 21 set. 2024.

SOUZA, C. C. d. et al. **Recycle quiz: Uma abordagem educacional para a conscientização ambiental**. Mosaico - Revista Multidisciplinar de Humanidades, 2023, Vassouras, v. 14, n. 3, p. 47–55.

STUDART, N. **A gamificação como design instrucional**. Revista Brasileira de Ensino de Física, 2022, Sociedade Brasileira de Física, v. 44, p. e20210362. ISSN 1806-1117. Disponível em: <<https://doi.org/10.1590/1806-9126-RBEF-2021-0362>>. Acesso em: 10 maio 2025.

WOOD, J. **This is what stops people from recycling more, finds a global survey**. 2021. Disponível em: <<https://www.weforum.org/agenda/2021/11/barriers-to-recycling-sustainability-survey/>>. Acesso em: 21 set. 2024.

ZEZZO, L. V.; COLTRI, P. P. **Educação em mudanças climáticas no contexto brasileiro: uma revisão integrada**. Terrae Didatica, 2022, v. 18, n. 00, p. e022039. Disponível em: <<https://periodicos.sbu.unicamp.br/ojs/index.php/td/article/view/8671305>>. Acesso em: 9 out. 2024.