

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ

MATHEUS ROCHA CAMARGO

**DESENVOLVIMENTO DA VISUALIZAÇÃO DE OBJETOS DE APRENDIZAGEM
NA FERRAMENTA FARMA**

GUARAPUAVA

2025

MATHEUS ROCHA CAMARGO

**DESENVOLVIMENTO DA VISUALIZAÇÃO DE OBJETOS DE APRENDIZAGEM
NA FERRAMENTA FARMA**

Development of Visualizationalization of Learning Objects at FARMA

Projeto de Trabalho de Conclusão de Curso de Graduação apresentado como requisito para obtenção do título de Tecnólogo em Tecnologia em Sistemas para Internet do Curso Superior de Tecnologia em Sistemas para Internet da Universidade Tecnológica Federal do Paraná.

Orientador: Prof. Dr. Diego Marczal

Coorientador: Prof^a. Dr^a. Renata Luiza Stange e
Prof. Dr. Alex Sandro De Castilho

GUARAPUAVA

2025



[4.0 Internacional](https://creativecommons.org/licenses/by/4.0/)

Esta licença permite compartilhamento, remixe, adaptação e criação a partir do trabalho, mesmo para fins comerciais, desde que sejam atribuídos créditos ao(s) autor(es). Conteúdos elaborados por terceiros, citados e referenciados nesta obra não são cobertos pela licença.

LISTA DE ABREVIATURAS E SIGLAS

Siglas

API	Application Programming Interface
FARMA	Ferramenta de Autoria para a Remediação de Erros com Mobilidade na Aprendizagem
HTML	HyperText Markup Language
JSON	JavaScript Object Notation
OA	Objeto de Aprendizagem
Rails	Ruby on Rails
TCC	Trabalho de Conclusão de Curso
VSCode	Visual Studio Code

SUMÁRIO

1	INTRODUÇÃO	4
1.1	Objetivos	4
1.1.1	Objetivo geral	5
1.1.2	Objetivos específicos	5
1.2	Justificativa	5
2	FARMA	7
2.1	Funcionalidades	9
2.2	O Papel do OA na FARMA	9
3	DOMÍNIO DO PROBLEMA	11
3.1	Defasagem Tecnológica	11
3.2	Visualização dos OAs	14
3.3	Representação visual dos OAs	15
4	MATERIAIS E MÉTODOS	19
4.1	Materiais	19
4.2	Métodos	20
4.2.1	Levantamento de requisitos	20
4.2.2	Prototipação da interface	21
4.2.3	Planejamento e organização do desenvolvimento	21
4.2.4	Implementação da visualização dos OAs	22
4.2.5	Verificação e validação	22
4.2.6	Entrega e Implantação	23
5	ANÁLISE E PROJETO	24
5.1	Histórias de Usuário	24
5.1.1	Visualização como Aluno	24
5.1.2	Visualização como Professor	26
5.1.3	Visualização como Visitante	28
5.2	Interfaces	29
5.2.1	Interface de Visualização do Aluno	30
5.2.2	Interface de Visualização do Professor	36
5.2.3	Interface de Visualização do Visitante	37

6	CONSIDERAÇÕES FINAIS	38
	REFERÊNCIAS	39

1 INTRODUÇÃO

Atualmente, diversas ferramentas digitais buscam aprimorar o processo de aprendizagem, dentre elas videoaulas, plataformas de ensino e e-books. Muitas dessas ferramentas oferecem recursos para que os alunos aprendam com seus próprios erros, mesmo que de forma indireta, ao proporcionar revisões e repetições de conteúdo. No entanto, não é comum ver materiais didáticos que tratem o erro como elemento central do processo de aprendizagem, tratando-o como um dado pedagógico ao invés de falhas a serem corrigidas.

Objetos de Aprendizagem (OA) são materiais didáticos assistidos por computador que podem ser aplicados em diversos contextos educacionais. A Ferramenta de Autoria para a Remediação de Erros com Mobilidade na Aprendizagem (FARMA) é uma ferramenta de autoria que permite a criação de OAs com uma característica distinta: a capacidade de utilizar os erros como elementos ativos no processo de aprendizagem através do mecanismo de retroação. Dessa forma, os OAs desenvolvidos com a FARMA possibilitam que professores personalizem suas abordagens didáticas voltadas para o ensino de conceitos matemáticos e que alunos recebam um *feedback* individualizado para corrigir desvios na formação de conceitos, promovendo um ciclo de aprendizado mais iterativo (MARCZAL, 2014).

Desde sua concepção, a ferramenta FARMA tem sido aprimorada gradualmente por meio de diversos trabalhos acadêmicos desenvolvidos ao longo dos anos. Entre os avanços realizados estão: a implementação da visualização de objetos de aprendizagem (RAMOS, 2019), o desenvolvimento do módulo de estatística da FARMA (SANTOS, 2022), o desenvolvimento da área de criação de OAs (VITEK, 2023), a construção de uma Application Programming Interface (API) para comunicação com esses objetos (BONIN, 2024) e a implementação de uma nova interface gráfica para criação do Objeto de Aprendizagem (OA) em uma arquitetura monolítica (RIBAS, 2024). A migração para arquitetura monolítica proposta por Ribas (2024) baseou-se na análise das dificuldades enfrentadas em trabalhos anteriores que adotaram sistemas distribuídos, onde a complexidade de manutenção e integração entre componentes resultou em sobrecarga de desenvolvimento sem benefícios proporcionais.

Considerando os avanços já realizados, este projeto propõe a continuidade do trabalho desenvolvido por Bonin (2024) e o aperfeiçoamento do trabalho de Ramos (2019), ambos com foco na visualização dos OAs na ferramenta FARMA. Para isso, serão elaborados e implementados os protótipos das interfaces necessárias para a visualização das diferentes etapas de um OA, aproveitando a base monolítica estabelecida por Ribas (2024).

1.1 Objetivos

A seguir é apresentado o objetivo geral do trabalho, bem como os objetivos específicos que vão orientar os resultados esperados.

1.1.1 Objetivo geral

Projetar e desenvolver interfaces gráficas responsáveis pela visualização e interação com Objetos de Aprendizagem (OA) na ferramenta FARMA, utilizando a base de arquitetura monolítica existente e implementando soluções modernas de desenvolvimento web.

1.1.2 Objetivos específicos

- Desenvolver interfaces gráficas para exibição de conteúdos teóricos e exercícios dos OAs;
- Implementar a navegação entre etapas dos OAs, com indicadores visuais de progresso do aluno;
- Exibir mensagens de acerto, erro e dicas conforme o desempenho do aluno;
- Validar as interfaces implementadas com os professores que vão fazer o uso da ferramenta.

1.2 Justificativa

A justificativa para desenvolver este trabalho se dá pela necessidade de implementar a visualização dos OAs na ferramenta FARMA, considerando que essa funcionalidade é necessária para que a plataforma cumpra seu papel pedagógico. Embora a proposta de visualização tenha sido inicialmente abordada por Jefferson Henrique Ramos (RAMOS, 2019), observa-se que o foco do trabalho estava mais concentrado na construção de um teclado virtual, enquanto o projeto atual foca na implementação da visualização dos OAs, utilizando um teclado mais simples e adequado às necessidades do projeto.

Trabalhos subsequentes, como o da Isabela Taques Vitek (VITEK, 2023), adotaram uma arquitetura baseada na separação entre *frontend* e *backend*, fazendo o uso de uma API para comunicação entre esses serviços. Embora essa abordagem esteja alinhada a práticas comuns de desenvolvimento, observou-se na prática que essa separação resultou em uma complexidade adicional e poucas vantagens. A manutenção e integração entre as partes exigiram um esforço significativo, sem um retorno proporcional.

No trabalho de Douglas Vinicius Bonin (BONIN, 2024), foi desenvolvida uma API para a visualização e interação com os OAs. Embora essa implementação tenha avançado em conceitos relacionados a interação com os OAs, este projeto optará por aproveitar essas ideias dentro de uma arquitetura monolítica. Em vez de retornar dados utilizando o formato JavaScript Object Notation (JSON), como em uma API tradicional, a visualização será feita diretamente por meio de páginas que utilizam a linguagem de marcação HyperText Markup Language (HTML) e serão

geradas no servidor, mantendo o controle da interface e da lógica de apresentação em um único sistema.

Essa decisão arquitetural é ainda mais justificada considerando que a implementação original de Ramos (2019) utiliza tecnologias que evoluíram significativamente desde sua criação. A modernização completa da base tecnológica existente representaria um esforço desproporcional aos benefícios que seriam obtidos, justificando a decisão de implementar uma nova versão que preserve os conceitos pedagógicos da FARMA, mas utilize padrões tecnológicos atuais.

A decisão por manter uma arquitetura monolítica se baseia no perfil da própria FARMA, que possui um número reduzido de usuários e não requer um nível elevado de escalabilidade. Os trabalhos anteriormente citados indicam que a separação de responsabilidades em múltiplos sistemas causa sobrecarga no desenvolvimento, dificultando a evolução contínua da ferramenta. Dessa forma, a opção por uma aplicação monolítica favorece a manutenção e a continuidade do projeto.

2 FARMA

A FARMA é um ambiente digital criado com o objetivo de auxiliar no ensino da Matemática por meio da criação e disponibilização de OAs. Sua concepção teve origem na tese de doutorado de Marczał (2014), que identificou a necessidade de ferramentas capazes de acompanhar o processo de ensino e aprendizagem por meio da análise de tentativas e erros.

A proposta central da FARMA é oferecer aos professores um ambiente que permite criar, gerenciar e disponibilizar OAs, proporcionando uma experiência de estudo personalizada aos estudantes. Isso permite a criação de conteúdos teóricos e exercícios práticos, a ferramenta também possibilita o acompanhamento do desempenho dos alunos, destacando seus erros e dificuldades.

Ao longo dos anos, a FARMA vem sendo tema de pesquisas em nível de mestrado, doutorado e trabalhos de conclusão de curso, dando origem a diferentes versões da ferramenta, como a *FARMA Calc*¹, a *FARMA ALG*² e a *FARMA Reborn*, sendo esta última um *redesign* e refatoração completa tanto na arquitetura quanto na interface, visando modernizar a ferramenta e torná-la mais intuitiva.

Para ilustrar essa trajetória da ferramenta, uma linha do tempo foi desenvolvida por Ribas (2024) que destaca a evolução da FARMA, a qual pode ser observada na cronologia abaixo:

- 2009 • Melhor Artigo do Workshop de Informática na Escola (WIE) - Sociedade Brasileira de Computação (SBC) (2º Lugar) ³
- 2010 • Concurso de Teses, Dissertações e Trabalhos de Conclusão de Curso em Informática na Educação (CTD-IE) - Sociedade Brasileira de Computação (SBC) (3º Colocado cat. Dissertação) ⁴
- 2010 • Um arcabouço que enfatiza a retroação a contextos de erro durante o acesso a conteúdos educacionais (MARCZAL, 2010)
- 2011 • Classificação automática de erros de aprendizes humanos do processo de indução analítica (BAZZO, 2011)
- 2013 • Arquitetura para remediação de erros baseada em múltiplas representações externas (LEITE, 2013)
- 2014 • Farma: uma ferramenta de autoria para objetos de aprendizagem de conceitos matemáticos (MARCZAL, 2014)
- 2015 • Sequenciamento adaptativo de exercícios baseado na correspondência entre a dificuldade da solução e o desempenho dinâmico do aprendiz (SILVA, 2015)

¹ *FARMA Calc*: versão da ferramenta focada no aprendizado do cálculo diferencial e integral

² *FARMA ALG*: versão da ferramenta desenvolvida para apoiar o ensino e a aprendizagem de programação e lógica de algoritmos

³ Disponível em <https://tophotels.com/tecnologia-da-informacao-no-sistema-educacional-do-brasil.pt-br>

⁴ Disponível em https://www.proativa.virtual.ufc.br/sbie2010/Anais_do_XXI_SBIE/CTD-IE-MS.html

- 2015 • Informática educacional e a mediação do erro na educação: um estudo teórico-crítico e uma proposta de instrumento computacional (KUTZKE, 2015)
- 2015 • Avaliação do impacto da retroação na aprendizagem apoiada por uma ferramenta educacional (MOURA, 2015)
- 2015 • Concurso de Teses, Dissertações e Trabalhos de Conclusão de Curso em Informática na Educação (CTD-IE) - Sociedade Brasileira de Computação (SBC) (1º colocado na cat. Tese)⁵
- 2015 • FARMA: Uma ferramenta de autoria para objetos de aprendizagem de conceitos matemáticos (MARCZAL, 2015)
- 2015 • Otimizando o processo de Ensino e Aprendizagem com a Arquitetura para Desenvolvimento de Objetos de Aprendizagem - ADOA (LEITE, 2015)
- 2016 • Metodologia e Software Educacional para a Investigação e Remediação de Erros Conceituais em Matemática (MARCZAL, 2016)
- 2017 • ATAUDIW - Uma ferramenta de autoria para auxiliar o uso da lousa digital interativa (COSTA, 2017)
- 2017 • EduTraceSys como objeto de aprendizagem na remediação de erros (TRIBECK, 2017)
- 2018 • Adaptabilidade de Objetos de Aprendizagem usando Calibragem e Sequenciamento Adaptativo de Exercícios (SILVA, 2018)
- 2018 • Projeto de uma nova interface gráfica para a ferramenta de autoria de objetos de aprendizagem matemáticos FARMA (SANTOS, 2018)
- 2019 • Objetos de Aprendizagem com Feedbacks para a Autorregulação da Aprendizagem de Conceitos Matemáticos Necessários para o Cálculo Diferencial e Integral (CASTILHO, 2019)
- 2019 • Visualização de Objetos de Aprendizagem para a Ferramenta de autoria FARMA (RAMOS, 2019)
- 2021 • Mudanças nas ideias prévias sobre ponto, reta e plano por meio da interação com a ferramenta FARMA (ARRUDA, 2021)
- 2022 • Desenvolvimento do Módulo de Estatísticas da Ferramenta de Autoria de Objetos de Aprendizagem FARMA (SANTOS, 2022)
- 2023 • Desenvolvimento da Área Destinada à Criação de Objetos de Aprendizagem para Ferramenta de Autoria FARMA (VITEK, 2023)
- 2024 • Implementação De Uma Api Para Visualização E Interação Com Objetos De Aprendizagem Para Ferramenta De Autoria Farma (BONIN, 2024)

Atualmente, a FARMA não permite apenas a criação de OAs, mas também oferece aos estudantes um ambiente para interação com esses materiais. Por meio de recursos como dicas, *feedbacks* automáticos e monitoramento de desempenho, a ferramenta foca em transformar erros em oportunidades de aprendizagem.

⁵ Disponível em <https://tophotels.com/tecnologia-da-informacao-no-sistema-educacional-do-brasil.pt-br>

2.1 Funcionalidades

Para promover a aprendizagem por meio da análise dos erros dos estudantes, a ferramenta oferece um conjunto de funcionalidades que servem tanto no processo de criação de conteúdos quanto o acompanhamento do desempenho dos alunos.

Entre as principais funcionalidades, destacam-se:

- **Criação do OA:** permite que os professores criem conteúdos compostos por introduções teóricas, exercícios e dicas. A interface oferece recursos para organizar os materiais de forma sequencial e estruturada.
- **Visualização do OA:** possibilita que os OAs sejam acessados pelos diferentes perfis de usuários, cada um com funcionalidades específicas:
 - **Área do aprendiz:** apresenta o conteúdo de forma sequencial, com acompanhamento do progresso, exibição de dicas e *feedback* após cada tentativa.
 - **Área do professor:** permite ao docente visualizar como o OA será exibido para os alunos, possibilitando revisar os conteúdos e acompanhar os registros de interação dos estudantes vinculados às suas turmas.
 - **Área pública (visitante):** oferece a possibilidade de acessar os OAs sem vínculo a uma turma. As interações realizadas nessa modalidade não são registradas no sistema.
- **Retroação a erros:** funcionalidade central da ferramenta, permite que o aluno retome o contexto no qual ocorreu um erro, revise introduções, utilize dicas e refaça exercícios. Essa abordagem visa promover a aprendizagem a partir da análise e compreensão dos próprios erros.
- **Gestão de turmas e usuários:** permite que os professores criem turmas, associem alunos e organizem quais OAs serão disponibilizados para cada grupo.

2.2 O Papel do OA na FARMA

Os OAs servem como meio para estruturar e organizar os conteúdos didáticos. Eles foram concebidos para promover a aprendizagem a partir da análise dos erros cometidos pelos alunos durante a resolução de exercícios, possibilitando a remediação desses erros de forma sistemática.

Na prática, um OA na FARMA é composto por diferentes elementos que se complementam, formando uma sequência pedagógica que orienta o estudante desde a introdução teórica até a resolução do exercício. Cada OA é construído pelo professor e pode ser vinculado a uma ou mais turmas, permitindo o acompanhamento individualizado do progresso dos alunos.

A estrutura de um OA inclui:

- **Introduções teóricas:** oferecem uma base conceitual necessária para que o aluno compreenda o conteúdo que será abordado nos exercícios. Podem incluir textos, imagens, vídeos e outros recursos de apoio.
- **Exercícios:** são as atividades propostas pelo professor, cada exercício pode ser composto por uma ou mais questões, e cada questão pode ser dividida em passos.
- **Dicas e orientações:** podem ser configuradas pelo professor para serem disponibilizadas ao aluno após um determinado número de tentativas incorretas, funcionando como suporte adicional no processo de aprendizagem.

Diante do entendimento das funcionalidades e do papel dos OAs na FARMA, será abordado o domínio do problema, destacando os desafios da defasagem tecnológica.

3 DOMÍNIO DO PROBLEMA

Pelo avanço constante das tecnologias utilizadas no desenvolvimento de software, realizar a manutenção em sistemas que não acompanham essas mudanças é cada vez mais desafiador. Projetos que permanecem longos períodos sem atualizações tendem a sofrer com acúmulo de débitos técnicos, bibliotecas sem contribuições há meses ou anos, falhas de segurança e dificuldades de integração com novas ferramentas atualizadas do mercado. Todos esses pontos não apenas dificultam o desenvolvimento, como também impactam negativamente a qualidade, a usabilidade e a manutenibilidade do sistema a longo prazo.

3.1 Defasagem Tecnológica

Segundo Li (2014), a defasagem tecnológica de um projeto acontece por conta de aspectos invisíveis relacionados à evolução do ecossistema dos softwares. Como tecnologias que caem em desuso, correções e melhorias que são aplicadas naquelas que já existem, troca por outras tecnologias mais seguras e modernas e assim por diante. Esses aspectos fazem com que os projetos que não recebem manutenção contínua fiquem defasados e percam sua compatibilidade com ferramentas modernas. Além disso, novos desenvolvedores que integram sistemas já em andamento têm um esforço cognitivo maior para entender o escopo de uma tecnologia que já não é mais tão utilizada e possuem maior risco de introdução de erros durante manutenções corretivas ou evolutivas.

No contexto do desenvolvimento web, esse problema se agrava pela velocidade com que frameworks JavaScript evoluem. O React, por exemplo, recebe atualizações frequentes que mudam a forma como componentes são escritos. A versão utilizada no trabalho de Ramos (2019) era baseada no React 16, onde a estrutura de componentes de classe ainda era o padrão na época, e muitos dos recursos modernos, como hooks, ainda estavam começando.

Abaixo, é extraído o componente App, componente raiz e ponto inicial de uma aplicação React, diretamente do repositório do projeto FARMA Reborn, uma reimplementação da ferramenta de autoria FARMA. A seguir, é feita uma comparação e mostro como esse mesmo componente poderia ser reescrito hoje, com React 19.

```
1 import React, { Component } from 'react';
2 import PropTypes from 'prop-types';
3 import store from './store';
4 import Router from './router';
5 import { loadExercise } from './store/actions/index';
6
7 class App extends Component {
8   constructor (props) {
9     super(props);
10    let { exerciseId } = this.props;
11    store.dispatch(loadExercise(exerciseId));
12  }
13
14  render () {
15    return <Router store={store} />;
16  }
17 }
18
19 App.propTypes = {
20   exerciseId: PropTypes.number.isRequired,
21 }
22
23 export default App;
```

Listing 3.1 – Código retirado do projeto Farma Reborn (React 16 - 2019)

Observações relacionadas a esse código comparado a versões atuais:

- Utiliza classe, enquanto o React moderno utiliza funções;
- Instancia e injeta a store de forma manual (atualmente usam-se 'Providers');
- Usa PropTypes (ainda é válido, mas TypeScript é mais comum atualmente);
- O router é utilizado como componente, tendo a store sendo passada como parâmetro e essa abordagem não é mais necessária com o uso de useSelector/useDispatch.

```

1  import { useEffect } from 'react';
2  import { useDispatch } from 'react-redux';
3  import { loadExercise } from './store/actions';
4  import Router from './router';
5
6  function App({ exerciseId }) {
7      const dispatch = useDispatch();
8
9      useEffect(() => {
10         dispatch(loadExercise(exerciseId));
11     }, [dispatch, exerciseId]);
12
13     return <Router />;
14 }
15
16 export default App;

```

Listing 3.2 – Sugestão de reescrita com React 19 (2025), Redux e hooks

Analisando a possível reescrita, podemos concluir que:

- A classe foi substituída por uma função com `useEffect`, que executa o carregamento quando o componente é montado;
- A store não é mais importada nem enviada como props, aqui assume-se que o `<Provider store=store>` está definido na raiz da aplicação;
- O Router também não precisa receber manualmente a store, já que os componentes internos acessam o estado com o uso da função `useSelector`.

A estrutura atual simplifica o código, melhora a legibilidade e permite maior reutilização de lógica com os hooks personalizados. Atualizar um projeto que foi escrito inteiramente com base em componentes de classe requer não apenas reescrever partes significativas do código, mas também adaptar bibliotecas, ajustando comportamentos de ciclo de vida, gerenciamento de estado e integração com outras dependências modernas como o React Router e o Redux. Assim, a defasagem tecnológica se torna um dos principais impedimentos para a evolução de projetos como o FARMA Reborn, dificultando sua reutilização como base para novas funcionalidades.

Embora o React moderno simplifique a estrutura dos componentes e promova uma organização mais clara do código, ele ainda depende de ser criado como um projeto à parte, o que pode introduzir uma complexidade desnecessária quando o objetivo não é criar interfaces altamente dinâmicas.

Ainda no aspecto tecnológico, o projeto original utilizava o framework Ruby on Rails na versão 5, que, embora moderno na época, hoje várias versões com novos recursos foram lançadas. Desde então, o framework passou por importantes atualizações, lançando sua versão 8 em dezembro de 2024. Essa evolução trouxe melhorias em termos de segurança, performance, suporte a arquiteturas modernas como APIs RESTful, e um modelo de comunicação assíncrono mais robusto baseado em WebSockets.

Utilizando o Hotwire, um conjunto de ferramentas mantido pela equipe do Rails, a renderização parcial da interface pode ser feita no lado do servidor, utilizando recursos como Turbo Streams e Turbo Frames. Essa solução reduz o acoplamento entre front-end e back-end, permitindo a atualização dinâmica da interface sem a necessidade do uso de JavaScript. A comunicação em tempo real, antes dependia de soluções externas, agora, ela pode ser realizada de forma integrada. Isso significa que manter a estrutura baseada no Rails 5 impede o uso nativo desses novos recursos.

Ao implementar a funcionalidade de visualização dos OAs com tecnologias atualizadas, espera-se não apenas modernizar a ferramenta, mas também contribuir com o campo educacional ao aplicar conceitos de remediação de erros em um ambiente digital.

O projeto contribui para o conhecimento acadêmico ao integrar aspectos de desenvolvimento web, design centrado no usuário e educação matemática, oferecendo uma abordagem prática sobre como o erro pode ser explorado pedagogicamente em sistemas interativos. Do ponto de vista técnico, o trabalho serve como um estudo sobre atualização tecnológica de sistemas, abordando os impactos da defasagem tecnológica de frameworks e a importância da manutenção contínua. Espera-se que assim, este trabalho sirva como base para futuras pesquisas e extensões da ferramenta, fomentando a continuidade do projeto e sua adoção mais ampla na prática docente.

3.2 Visualização dos OAs

A funcionalidade de visualização OAs é um dos componentes principais do sistema. É por meio dessa interface que o estudante acessa os conteúdos teóricos, interage com os exercícios e recebe mensagens conforme seu desempenho.

Uma primeira implementação dessa visualização foi proposta por Ramos (2019), com foco na construção de um teclado virtual, e chegou a apresentar uma estrutura inicial para exibição dos OAs. No entanto, ao longo dos anos, essa versão foi se tornando obsoleta devido às novas versões lançadas das tecnologias utilizadas na plataforma. Trabalhos posteriores voltaram sua atenção à criação dos OAs, como é o caso do trabalho de Vitek (2023) e Bonin (2024), porém sem retomar ou atualizar a visualização iniciada anteriormente.

Este projeto propõe implementar a funcionalidade de visualização de Objetos de Aprendizagem na nova versão da ferramenta FARMA. Isso será realizado utilizando tecnologias mais

atuais e compatíveis com o ecossistema Ruby on Rails, que possibilitem maior integração entre os componentes do sistema e menor dependência de bibliotecas descontinuadas.

3.3 Representação visual dos OAs

A nova implementação não será construída desde o início, mas sim como uma continuidade dos trabalhos realizados por Bonin (2024), que estruturou as rotas de API para a visualização dos OAs, e Ramos (2019), que idealizou o primeiro modelo de visualização. O presente trabalho terá como foco principal a reformulação das interfaces responsáveis pela exibição dos conteúdos de um OA, que incluem:

- Introduções teóricas;
- Navegação entre os passos de resolução dos exercícios;
- Indicadores de progresso do aluno;
- Exibição de mensagens de acerto e erro (retroação);
- Exibição de dicas.

Essa visualização será baseada nas interfaces do projeto FARMA Reborn. Essas interfaces representam o funcionamento esperado da visualização de um OA na prática. A seguir, são descritas as principais telas com suas respectivas funcionalidades que compõem essa visualização.

A Figura 1 exibe a estrutura de um exercício de um OA. O conteúdo é organizado em blocos que apresentam um problema, seguido de uma questão que propõe a resolução com base nos dados fornecidos. Esse tipo de apresentação busca promover a compreensão gradativa do conceito trabalhado.

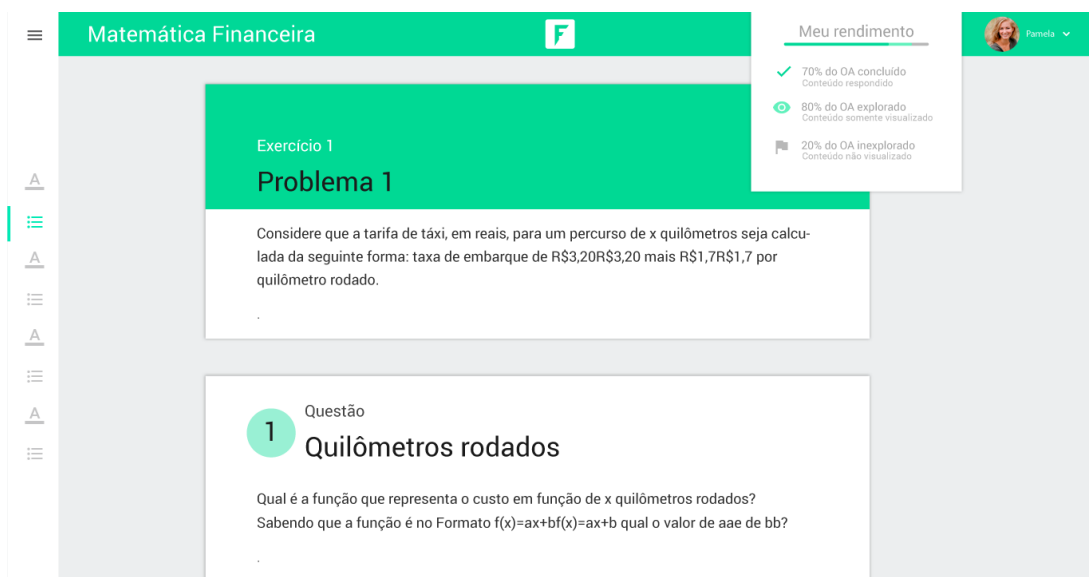


Figura 1 – Tela de exercício com enunciado e questão

A Figura 2 demonstra a introdução teórica de um conteúdo. Nesse exemplo, o conceito de função é apresentado com apoio de um texto explicativo e observações organizadas por tópicos. Essa introdução tem o papel de preparar o estudante para os exercícios, reforçando o entendimento teórico necessário.

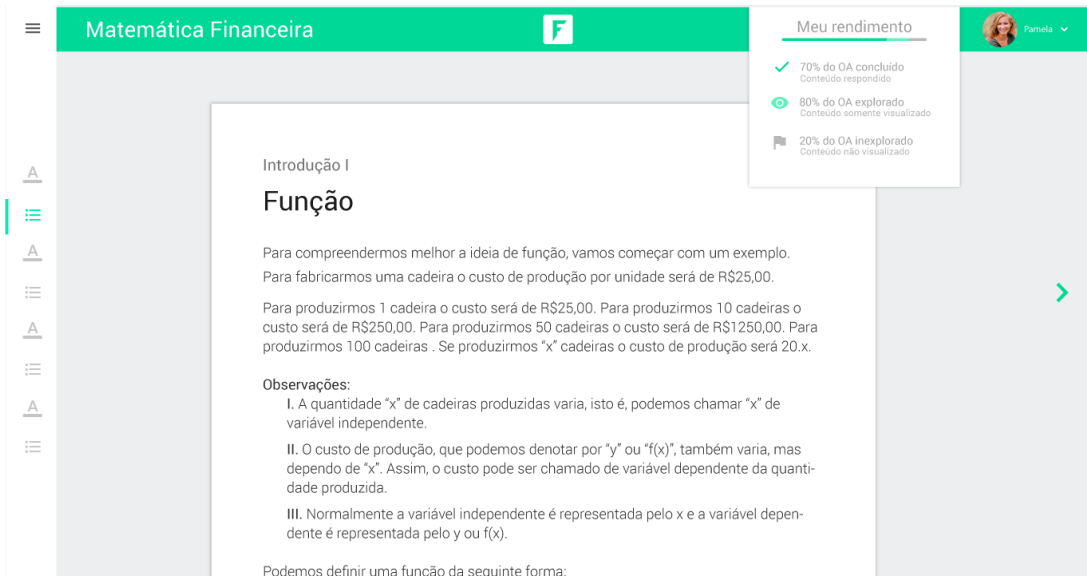


Figura 2 – Tela de introdução teórica de um OA

Na Figura 3, é possível visualizar a navegação entre os elementos do OA, incluindo introduções e exercícios organizados por etapas. Cada cor dentro do item de exercício representa o status daquela questão, sendo vermelho quando o aluno erra a questão, amarelo quando visualizado e não respondido e verde quando respondida de forma correta. Essa estrutura permite ao estudante visualizar o seu progresso e acessar os conteúdos de forma lógica. No canto superior direito, um painel de desempenho mostra estatísticas relacionadas ao completamento do OA pelo estudante.



Figura 3 – Navegação entre introduções e exercícios, com painel de desempenho do aluno

As Figuras 4 e 5 ilustram a tela de resposta do exercício, onde o aluno interage com o sistema por meio de um teclado virtual. Esse teclado disponibiliza operadores e símbolos matemáticos organizados por categorias como funções, matrizes e trigonometria. A interface também inclui um campo de resposta e botões para envio e limpeza, promovendo uma interação direta com o exercício proposto.

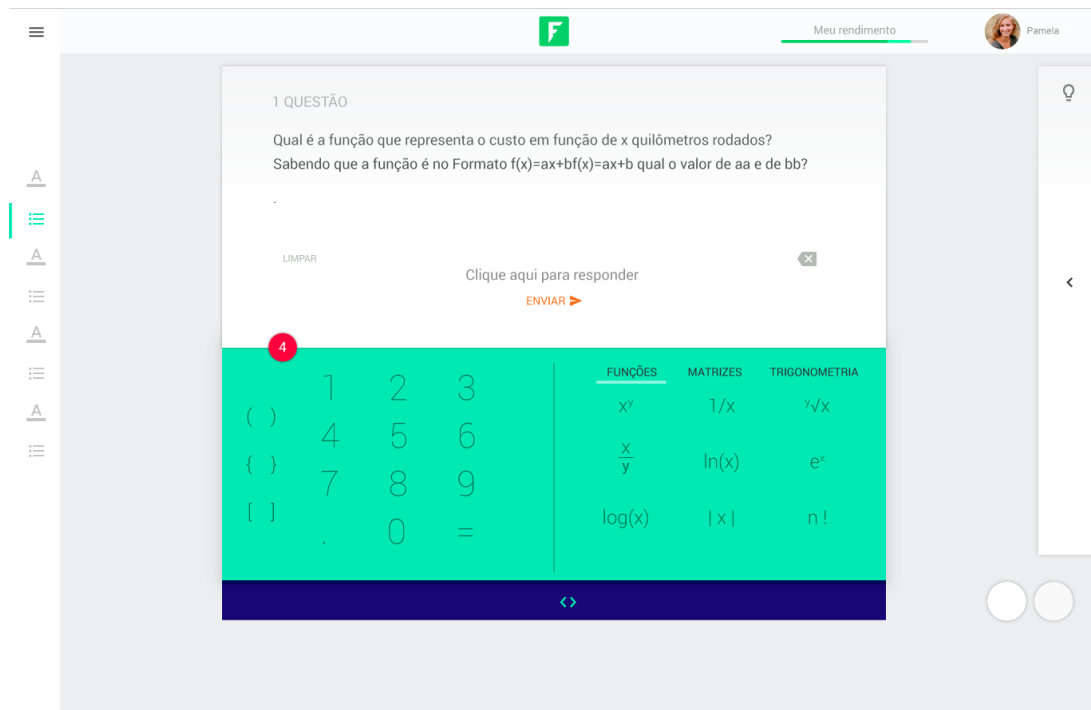


Figura 4 – Tela de resposta com teclado virtual (categoria: Funções)

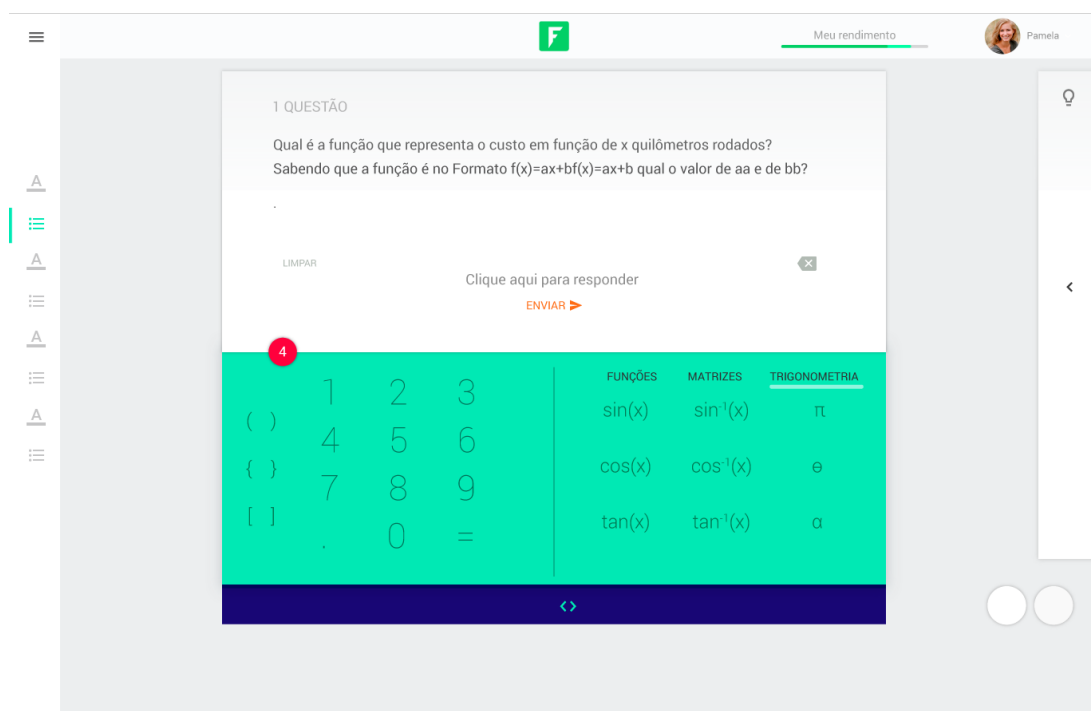


Figura 5 – Tela de resposta com teclado virtual (categoria: Trigonometria)

Por fim, a Figura 6 mostra o painel lateral de dicas. As dicas são exibidas ao lado da questão, possibilitando ao estudante consultar orientações definidas previamente pelo professor, sem sair do ambiente da questão. Essa funcionalidade está diretamente ligada à proposta de retroação, pois permite que o aluno receba orientações baseadas em seus erros ou dificuldades.

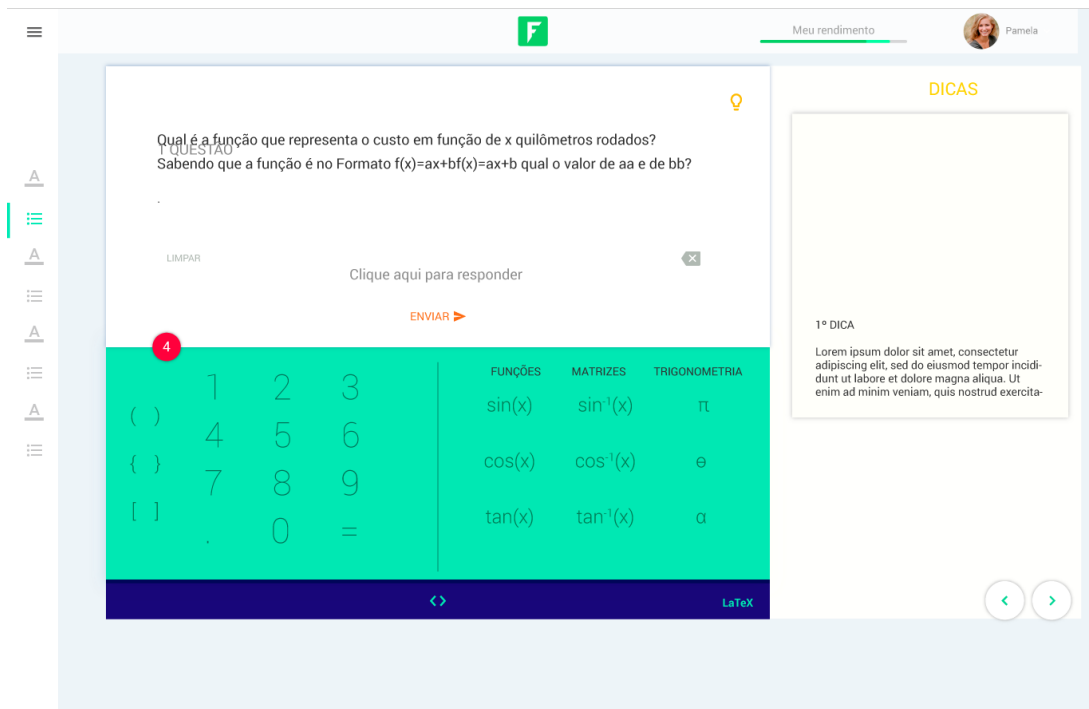


Figura 6 – Exibição de dica contextual ao lado da questão

Esses exemplos representam os principais elementos que compõem a visualização de um OA: introduções, exercícios, estrutura de navegação, interação via teclado e exibição de dicas. Este trabalho se propõe em implementar essas visualizações utilizando tecnologias atualizadas, conforme discutido na seção anterior, eliminando o impedimento causado pela defasagem tecnológica.

4 MATERIAIS E MÉTODOS

Este capítulo descreve os materiais utilizados no desenvolvimento deste trabalho, bem como os métodos adotados para alcançar os objetivos propostos.

4.1 Materiais

Para o desenvolvimento deste trabalho, serão utilizadas ferramentas que abrangem desde a prototipação das telas até o desenvolvimento das interfaces. A seguir, estão listados os principais materiais que serão utilizados:

- **Visual Studio Code (VSCode)**¹: editor de código-fonte gratuito e leve, desenvolvido pela Microsoft. Será utilizado no desenvolvimento por sua compatibilidade com sistemas Linux, Windows e macOS, além de oferecer suporte nativo a linguagens como JavaScript. Conta ainda com um vasto ecossistema de extensões para diversas linguagens e funcionalidades, como depuração de código (debugging) e integração com sistemas de controle de versão como o Git.
- **Ruby on Rails (Rails)**²: framework de código aberto utilizado no desenvolvimento de aplicações web baseado na linguagem Ruby, que será utilizado na construção da visualização dos OAs.
- **Hotwire**³: conjunto de tecnologias integrado ao Rails que permitirá a construção de uma interface reativa e dinâmica, dispensando bibliotecas externas de JavaScript. Será utilizado principalmente para implementar a navegação e interações da visualização dos OAs;
- **Hypertext Markup Language (HTML)**⁴: linguagem de marcação utilizada para estruturar o conteúdo de páginas web
- **TailwindCSS**⁵: framework utilitário de CSS que será empregado para estilização das páginas. Fornece um conjunto de classes predefinidas que podem ser combinadas para estilizar diretamente elementos HTML;
- **PostgreSQL**⁶: sistema gerenciador de banco de dados relacional, conhecido pela sua confiabilidade, manter a integridade dos dados, e seu conjunto extenso de ferramentas. Será utilizado para armazenar e consultar os dados relacionados aos OAs;

¹ <https://code.visualstudio.com/>

² <https://rubyonrails.org/>

³ <https://hotwired.dev/>

⁴ <https://html.spec.whatwg.org/>

⁵ <https://tailwindcss.com/>

⁶ <https://www.postgresql.org/>

- **Docker**⁷: ferramenta de containerização que será utilizada para padronizar o ambiente de desenvolvimento e facilitar a implantação da aplicação;
- **ClickUp**⁸: plataforma de gestão de projetos que será empregada para organização das tarefas e acompanhamento do progresso, por meio de quadros Kanban e listas de tarefas;
- **Git e GitHub**⁹: ferramentas utilizadas para versionamento de código e colaboração, com o apoio de issues, branches e pull requests, integradas ao fluxo de desenvolvimento contínuo;
- **Figma**¹⁰: plataforma de prototipação que será utilizada para projetar e validar visualmente os fluxos das interfaces de visualização antes do desenvolvimento.

4.2 Métodos

Para a realização deste trabalho, será seguido um conjunto de métodos que abrangem desde o levantamento dos requisitos até a validação do sistema desenvolvido. As etapas previstas estão descritas a seguir.

4.2.1 Levantamento de requisitos

Inicialmente, será realizado o levantamento de requisitos com base na análise da ferramenta FARMA, buscando compreender como se dá atualmente a visualização dos (OAs) e quais são suas limitações. Essa etapa contribuirá para identificar as funcionalidades esperadas e os pontos de melhoria.

Para organizar os requisitos e acompanhar o progresso do desenvolvimento, será utilizado um quadro Kanban estruturado na plataforma ClickUp. O Kanban é uma metodologia visual de gerenciamento de tarefas que facilita a organização e o acompanhamento do trabalho em equipe.

Cada tarefa será inserida inicialmente na coluna *Backlog*, que representa a lista de funcionalidades e melhorias identificadas, mas que ainda não foram iniciadas. À medida que forem avançando, as tarefas passarão pelas seguintes etapas:

- **A Fazer (Ready)**: tarefas já priorizadas e prontas para serem desenvolvidas;
- **Em Andamento (In Progress)**: tarefas que estão em processo de implementação;

⁷ <https://www.docker.com/>

⁸ <https://clickup.com/pt-BR>

⁹ <https://git-scm.com/> e <https://github.com/>

¹⁰ <https://www.figma.com/pt-br/>

- **Em Revisão (In Review):** tarefas que já foram implementadas e aguardam a revisão técnica e validação do orientador;
- **Concluído (Done):** tarefas finalizadas e aprovadas;

Essa estrutura permitirá um maior controle sobre o andamento do projeto, facilitando o acompanhamento por parte do orientador. A divisão dos requisitos em tarefas menores também contribui para o acompanhamento durante o processo de desenvolvimento.

4.2.2 Prototipação da interface

Com base nos requisitos levantados, serão desenvolvidos protótipos das telas da visualização dos OAs utilizando a ferramenta Figma. Os protótipos permitirão antecipar a estrutura e o fluxo de navegação da interface, possibilitando ajustes antes do início da implementação. A paleta de cores e a tipografia seguirão os padrões visuais previamente adotados na FARMA.

4.2.3 Planejamento e organização do desenvolvimento

O desenvolvimento seguirá um fluxo baseado no GitLab Flow (Figura 7), com ramificações específicas para diferentes tipos de tarefa. Serão utilizadas as seguintes *branches*:

- **Main:** branch principal do repositório, onde será mantida a versão mais estável do código. Servirá como base para criação de novas funcionalidades e correções (não significa que está disponível para o usuário acessar);
- **Staging:** ambiente intermediário para testes e validações. As funcionalidades implementadas nas branches *feature* ou *bugfix* serão integradas nesta branch antes de serem encaminhadas para produção.
- **Production:** representa a versão estável da aplicação, que está implantada no ambiente de produção e acessível aos usuários finais. Apenas funcionalidades validadas e testadas serão integradas nesta branch.
- **Feature:** criada a partir da branch *main* para o desenvolvimento de novas funcionalidades. Após sua implementação e validação, será integrada nas branches *main* e *staging*.
- **BugFix:** criada a partir da branch *staging* para corrigir falhas menores identificadas durante a fase de testes. Após a correção, será feito o merge na branch *staging* e, se necessário, na *main*.

- **HotFix:** utilizada para corrigir erros críticos detectados em produção. É criada a partir da branch *production* e, uma vez finalizada, é integrada simultaneamente às branches *production*, *staging* e *main*, garantindo a consistência entre os ambientes.

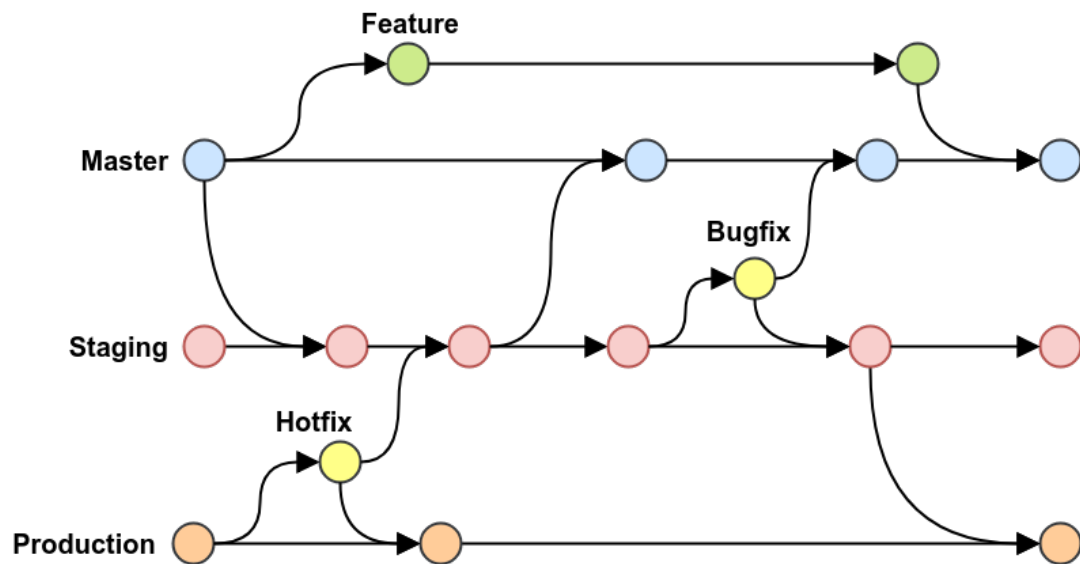


Figura 7 – GitLab Flow

Fonte: <https://blog.programster.org/git-workflows>.

Cada tarefa de desenvolvimento será representada por uma *issue*¹¹ associada a uma *pull request*¹², contendo descrição e os critérios de aceite.

4.2.4 Implementação da visualização dos OAs

Nesta etapa, serão implementadas as telas responsáveis por exibir os conteúdos dos OAs, de acordo com os protótipos validados previamente.

4.2.5 Verificação e validação

A verificação do sistema será realizada por meio de testes de funcionalidade e revisão de código. Os testes incluirão verificações manuais e automatizadas, sempre que possível, com o objetivo de assegurar que os requisitos implementados estejam em conformidade com o esperado.

A validação ocorrerá por meio de testes em ambiente controlado, simulando o uso da visualização dos OAs e avaliando se as interações e fluxos estão adequados.

¹¹ Uma *issue* é um registro de tarefa, sugestão ou erro no repositório, utilizada para organizar e acompanhar o progresso do desenvolvimento.

¹² *Pull Requests* são solicitações de revisão e integração de mudanças feitas em uma branch ao repositório principal, permitindo controle e colaboração no código

4.2.6 Entrega e Implantação

Será adotada a metodologia de Integração Contínua e Entrega Contínua, do inglês *Continuous Integration / Continuous Delivery (CI/CD)*. Essa prática possibilita que novas funcionalidades e melhorias desenvolvidas no projeto sejam testadas, integradas e implantadas de forma automatizada e incremental.

A integração contínua (*CI*) diz respeito à automatização do processo de integração das alterações no código, incluindo a execução de testes automatizados sempre que novas atualizações forem enviadas ao repositório. Essa abordagem permite identificar erros rapidamente, assegurando que o sistema continue funcionando conforme o esperado.

Já a entrega contínua (*CD*) consiste na automatização da publicação do sistema em ambientes específicos após a aprovação das alterações. As funcionalidades validadas passam inicialmente pelo ambiente de testes (*staging*) e, se aprovadas, são implantadas no ambiente final de produção. Essa separação garante que apenas versões estáveis e revisadas sejam disponibilizadas aos usuários finais.

5 ANÁLISE E PROJETO

Neste capítulo serão apresentados a análise e o projeto da funcionalidade de visualização dos OAs. Inicialmente, são descritas as histórias de usuário, que representam os requisitos funcionais para o desenvolvimento da visualização. As histórias foram elaboradas considerando as necessidades de diferentes tipos de usuários na plataforma. Essas histórias servem como base para a criação dos protótipos, que serão apresentados na sequência deste capítulo.

5.1 Histórias de Usuário

As histórias de usuário representam os requisitos funcionais do sistema sob a perspectiva dos diferentes perfis de usuários que interagem com a visualização dos OAs na ferramenta FARMA. Cada história detalha comportamentos específicos do sistema e cenários de uso, servindo como guia para o desenvolvimento e validação das funcionalidades.

Considerando as particularidades de cada tipo de usuário, as histórias foram organizadas em três categorias principais: professor, aluno e visitante. Essa separação permite abordar as necessidades específicas de cada perfil, desde funcionalidades de acompanhamento pedagógico até a exibição de dicas e acesso público aos conteúdos.

5.1.1 Visualização como Aluno

As histórias de usuário apresentadas nesta seção descrevem as funcionalidades específicas para estudantes que acessam a ferramenta FARMA com o objetivo de interagir com os OAs disponibilizados em suas turmas. Essas histórias contemplam desde o acesso inicial aos conteúdos até as interações específicas com exercícios, incluindo o registro de progresso e o sistema de retroação baseado em erros. Cada história detalha o comportamento esperado do sistema durante a navegação e resolução de atividades, considerando as particularidades do perfil de aluno autenticado na plataforma.

Feature: Visualizar OAs disponíveis na turma

Como aluno **quero** visualizar os OAs disponíveis na minha turma **para** que eu possa selecionar qual conteúdo desejo estudar.

Dado que estou autenticado como aluno

Quando acesso uma turma em que estou matriculado

Então o sistema deve exibir a lista de OAs daquela turma com título, descrição e status de conclusão.

Feature: Acessar um Objeto de Aprendizagem

Como aluno **quero** selecionar e acessar um OA específico **para** iniciar o estudo do conteúdo.

Dado que visualizo a lista de OAs da minha turma

Quando clico em um OA disponível

Então o sistema deve carregar a interface de navegação do OA com introduções e exercícios organizados sequencialmente.

Feature: Navegar entre introduções teóricas

Como aluno **quero** acessar as introduções teóricas do OA **para** estudar os conceitos antes dos exercícios.

Dado que estou visualizando a navegação do OA

Quando clico em uma introdução teórica

Então o sistema deve exibir o conteúdo da introdução e marcá-la como visualizada.

Feature: Acessar exercícios do OA

Como aluno **quero** selecionar um exercício **para** visualizar o enunciado e começar a resolver as questões.

Dado que estou navegando no OA

Quando clico em um exercício disponível

Então o sistema deve exibir o enunciado do exercício e a primeira questão disponível.

Feature: Receber mensagem de acerto ou erro ao submeter uma resposta

Como aluno **quero** receber *feedback* imediato após enviar minha resposta **para** saber se está correta ou incorreta.

Dado que estou respondendo uma questão

Quando preencho o campo de resposta e clico em enviar

Então o sistema deve validar a resposta e exibir uma mensagem indicando acerto ou erro.

Feature: Visualizar dicas após atingir o número de tentativas incorretas

Como aluno **quero** receber dicas quando erro várias vezes **para** me ajudar a compreender como resolver a questão.

Dado que respondi incorretamente uma questão pelo número de tentativas definido pelo professor

Quando este limite for atingido

Então o sistema deve exibir a dica cadastrada para esta questão.

Feature: Visualizar progresso no OA

Como aluno **quero** acompanhar meu progresso no OA **para** saber quais questões já respondi corretamente.

Dado que estou navegando no OA

Quando acesso o painel de progresso

Então o sistema deve mostrar o status das questões e estatísticas de conclusão.

Feature: Retomar OA em progresso

Como aluno **quero** continuar um OA que iniciei anteriormente **para** prosseguir de onde parei sem perder meu progresso.

Dado que já interagi anteriormente com um OA

Quando acesso novamente este OA

Então o sistema deve carregá-lo mantendo todo o progresso anterior e histórico de respostas.

5.1.2 Visualização como Professor

O professor possui acesso diferenciado à visualização dos OAs, com funcionalidades específicas voltadas ao acompanhamento pedagógico e validação dos conteúdos criados. Essas funcionalidades permitem ao docente verificar como os OAs são apresentados aos alunos, acompanhar o desempenho das turmas e validar a estrutura pedagógica dos materiais desenvolvidos.

Feature: Visualizar lista de OAs criados

Como professor **quero** acessar a lista dos OAs que criei **para** selecionar qual desejo visualizar ou revisar.

Dado que estou autenticado como professor

Quando acesso a seção de meus OAs

Então o sistema deve exibir todos os OAs que criei com título, descrição e data de criação.

Feature: Visualizar OA como seria apresentado ao aluno

Como professor **quero** visualizar meu OA da mesma forma que será apresentado aos alunos **para** verificar se o conteúdo está adequado.

Dado que selecionei um OA que criei

Quando clico na opção de visualizar

Então o sistema deve carregar o OA com a mesma interface e navegação disponível para os alunos.

Feature: Navegar pelas introduções criadas

Como professor **quero** acessar as introduções teóricas do meu OA **para** revisar o conteúdo e verificar a apresentação.

Dado que estou visualizando meu OA

Quando clico em uma introdução teórica

Então o sistema deve exibir o conteúdo da introdução conforme configurado.

Feature: Testar exercícios criados

Como professor **quero** acessar e testar os exercícios do meu OA **para** verificar se estão funcionando corretamente.

Dado que estou visualizando meu OA

Quando clico em um exercício

Então o sistema deve exibir o exercício permitindo que eu teste as respostas e validações.

Feature: Verificar funcionamento das dicas

Como professor **quero** testar as dicas configuradas nos exercícios **para** confirmar que serão exibidas adequadamente aos alunos.

Dado que estou testando um exercício com dicas configuradas

Quando respondo incorretamente pelo número de tentativas definido

Então o sistema deve exibir a dica conforme configurada para aquela questão.

Feature: Validar respostas dos exercícios

Como professor **quero** testar as respostas corretas cadastradas **para** garantir que a validação está funcionando adequadamente.

Dado que estou respondendo uma questão do meu OA

Quando insiro a resposta correta cadastrada

Então o sistema deve reconhecer como acerto e permitir prosseguir para a próxima questão.

5.1.3 Visualização como Visitante

As histórias de usuário desta seção descrevem as funcionalidades específicas para visitantes que acessam a ferramenta FARMA sem necessidade de autenticação. Estes usuários podem visualizar apenas os OAs disponibilizados publicamente pelos professores, sem vínculo com turmas específicas. A interação do visitante com o OA segue o mesmo padrão das histórias do aluno para navegação entre introduções teóricas, acesso a exercícios, submissão de respostas e visualização de dicas. A principal diferença é no fato de que o sistema não registra nem persiste o progresso do visitante, sendo todas as interações descartadas ao encerrar a interação com o OA.

Feature: Visualizar OAs disponíveis publicamente

Como visitante **quero** acessar a lista de OAs públicos **para** selecionar conteúdos disponíveis sem necessidade de cadastro.

Dado que acesso a ferramenta sem autenticação

Quando navego para a seção de OAs públicos

Então o sistema deve exibir todos os OAs disponibilizados como públicos pelos professores.

Feature: Acessar OA público tendo a mesma visualização de um aluno

Como visitante **quero** visualizar um OA público **para** explorar seu conteúdo

Dado que selecionei um OA público

Quando acesso sua visualização

Então o sistema deve carregar a interface de navegação e permitir que eu navegue assim como um aluno.

Feature: Interagir com exercícios sem persistência de dados

Como visitante **quero** responder exercícios dos OAs públicos **para** testar meus conhecimentos sabendo que os dados não serão salvos.

Dado que estou visualizando um exercício como visitante

Quando submeto uma resposta

Então o sistema deve validar e exibir *feedback* sem registrar a tentativa no banco de dados.

Feature: Receber feedback temporário

Como visitante **quero** receber dicas e mensagens de erro durante a interação com o OA **para** auxiliar na compreensão dos exercícios.

Dado que estou respondendo questões como visitante

Quando atinjo o número de tentativas incorretas

Então o sistema deve exibir as dicas configuradas, mantendo as informações apenas durante o acesso atual.

5.2 Interfaces

As interfaces apresentadas nesta seção foram desenvolvidas com base nas histórias de usuário definidas na seção 5.1. Cada elemento visual foi projetado para atender necessidades específicas do processo de aprendizagem, conforme detalhado nas subseções a seguir.

5.2.1 Interface de Visualização do Aluno

A interface do aluno foi projetada para apresentar informações de progresso e permitir a navegação direta pelos conteúdos. As principais características desta interface são apresentadas a seguir.

A tela inicial do aluno (Figura 8) implementa as histórias de usuário *"Visualizar OAs disponíveis na turma"* e *"Retomar OA em progresso"*. Cada OA é exibido em formato de card contendo título, descrição resumida e um indicador circular de progresso com porcentagem.

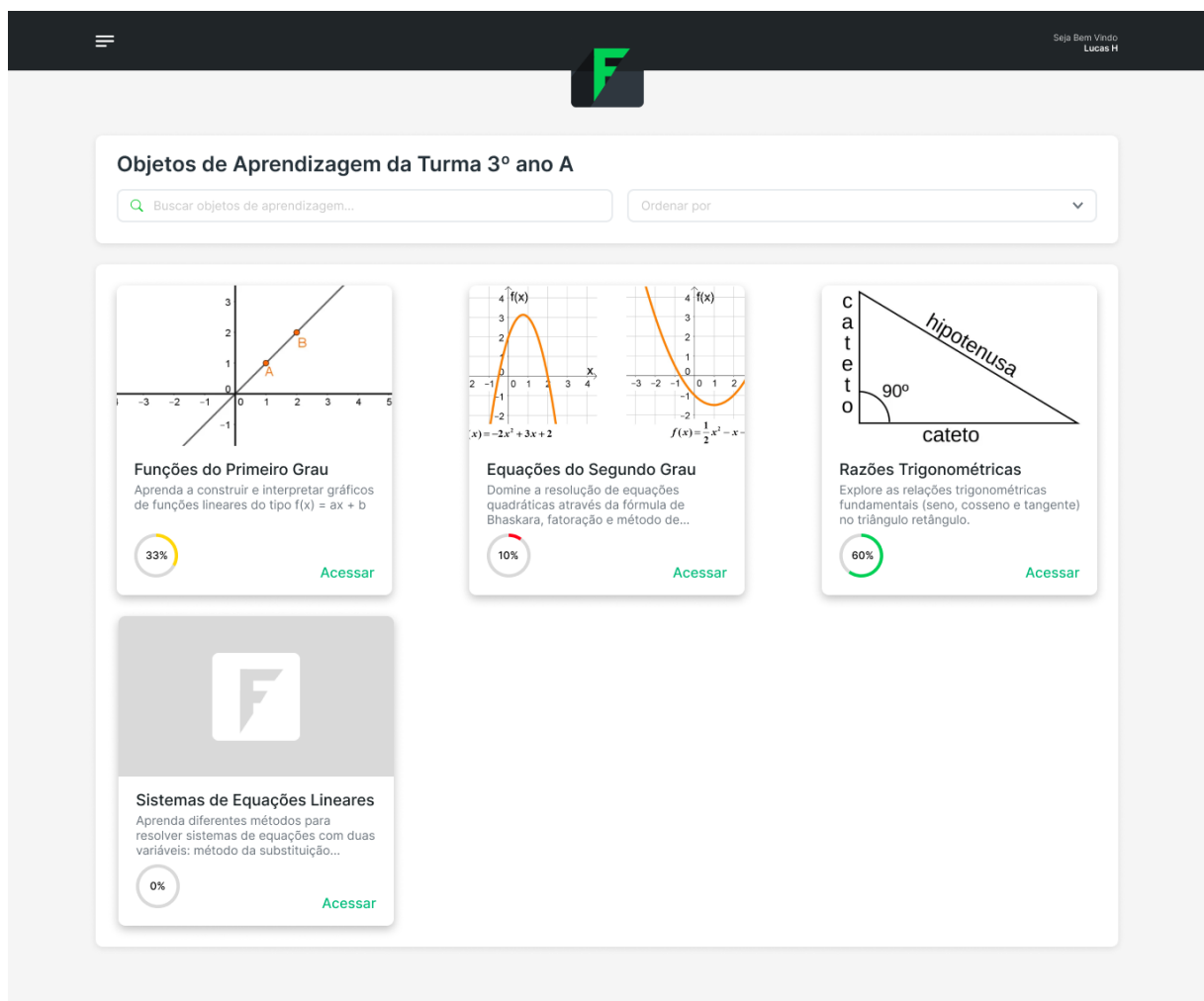


Figura 8 – Tela de Listagem dos Objetos de Aprendizagem

A decisão de tornar o progresso percentual proeminente responde a uma necessidade pedagógica fundamental: o aluno precisa compreender imediatamente sua situação em cada OA para tomar decisões informadas sobre seus estudos. O indicador circular com cores diferenciadas (verde para progresso significativo, laranja para progresso inicial) oferece um feedback visual instantâneo que promove o engajamento e a autorregulação da aprendizagem.

O campo de busca e filtros foram posicionados no topo da interface porque, em turmas com múltiplos OAs, para o aluno localizar rapidamente um conteúdo específico, isso reduz o tempo gasto em navegação, permitindo que o estudante dedique mais tempo ao aprendizado.

Página de Introdução: A Figura 9 implementa a história de usuário *"Navegar entre introduções teóricas"*. A separação da introdução em página em uma página isolada permite que o aluno concentre-se exclusivamente na assimilação dos conceitos fundamentais, conforme especificado na história *"Acessar um Objeto de Aprendizagem"* que estabelece a necessidade de "carregar a interface de navegação do OA com introduções e exercícios organizados sequencialmente". O cabeçalho verde identifica claramente o tipo de conteúdo, criando uma associação visual que auxilia na organização mental do estudante.

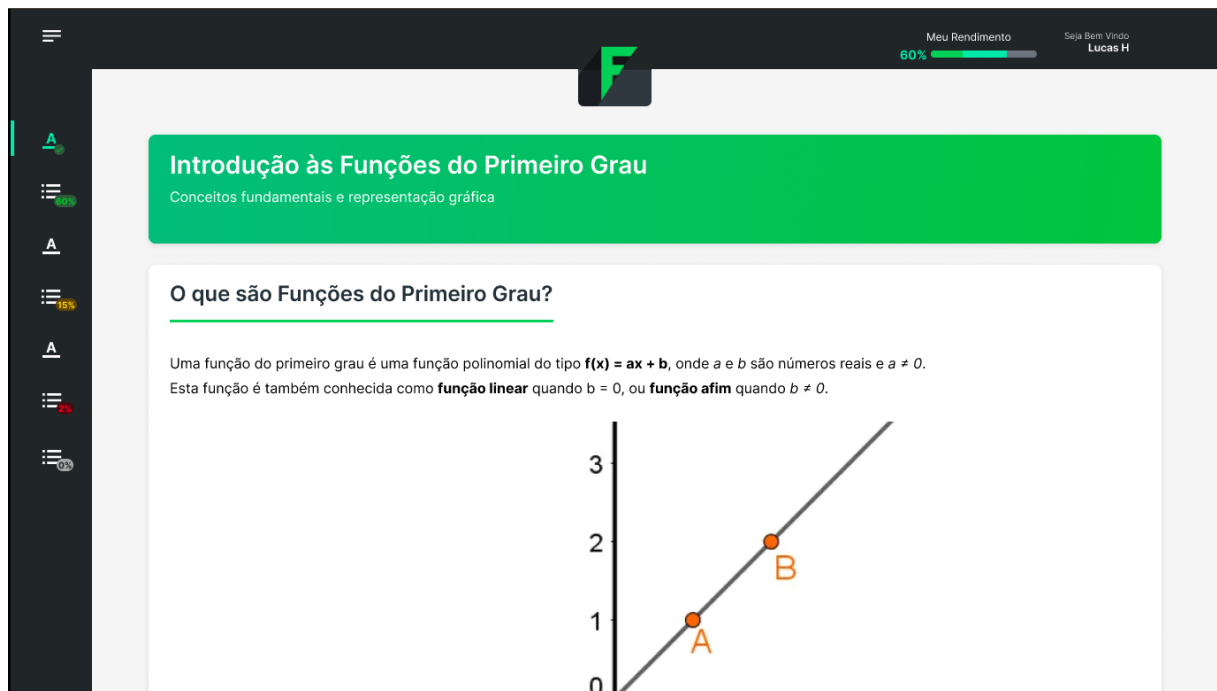


Figura 9 – Visualização da Introdução no OA

Separação em Páginas: Introdução, Exercício e Questões

A arquitetura de navegação foi projetada para separar cada etapa do processo de aprendizagem em páginas distintas. Esta decisão fundamenta-se na proposta pedagógica da FARMA de promover aprendizagem através da progressão sequencial entre conteúdos teóricos e práticos.

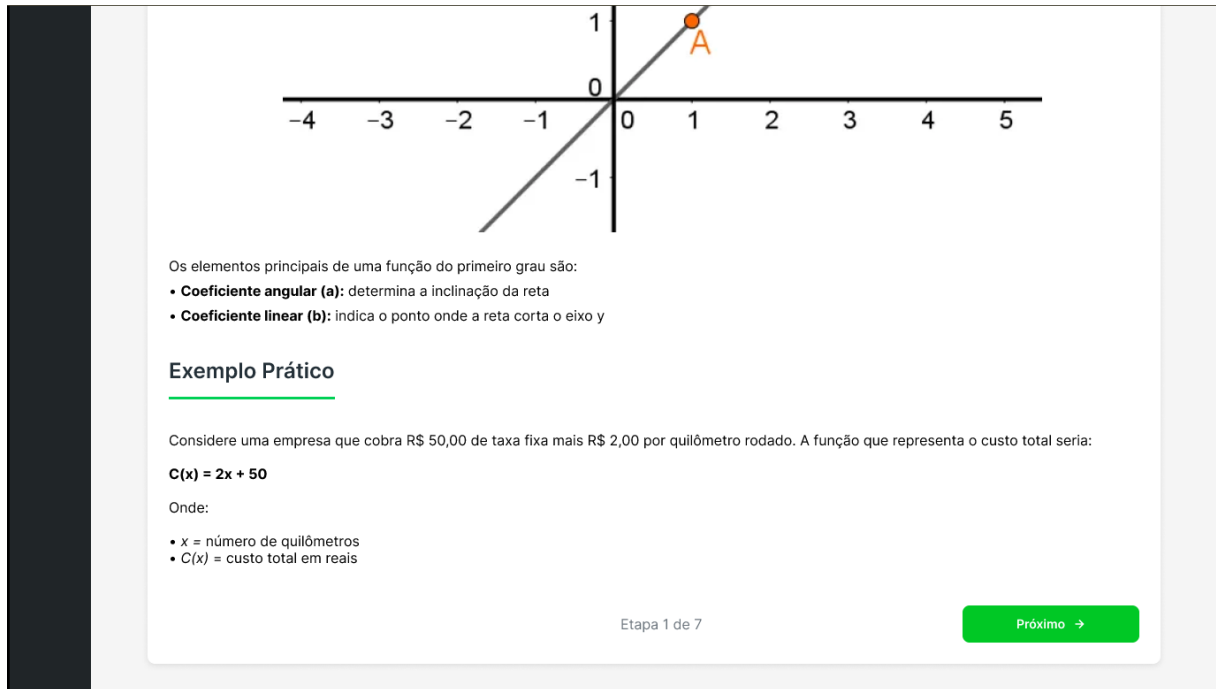


Figura 10 – Navegação Entre Etapas de Um OA

Página de Exercício: Conforme ilustrado na Figura 11, os exercícios implementam a história de usuário "Acessar exercícios do OA", apresentando o ambiente com bordas coloridas que delimitam cada questão. Esta separação cria um contexto visual específico para resolução de problemas, atendendo ao requisito especificado na história de "exibir o enunciado do exercício e a primeira questão disponível".

Meu Rendimento 60% Seja Bem Vindo Lucas H

Introdução às Funções do Primeiro Grau

Conceitos fundamentais e representação gráfica

Exercício 1 - Aplicando os Conceitos

Problema: Análise de uma Função

Uma fábrica produz peças com um custo fixo de R\$ 200,00 e um custo variável de R\$ 5,00 por peça produzida. Determine a função que representa o custo total de produção e analise as questões a seguir.

1

Escreva a função $C(x)$ que representa o custo total de produção, onde x é o número de peças produzidas.

[Clique aqui para responder](#)

Digite sua resposta no formato: $C(x)=...$

Figura 11 – Visualização de Um Exercício de Um OA

Modal de Questões: As questões individuais abrem em modais (Figura 12) implementando múltiplas histórias de usuário simultaneamente. O modal atende à história *"Receber mensagem de acerto ou erro ao submeter uma resposta"* a partir do momento em que o usuário submete sua resposta e tem o *feedback* baseado em se a resposta foi correta ou não. A funcionalidade de dicas (Figura 13) implementa a história *"Visualizar dicas após atingir o número de tentativas incorretas"*, exibindo as orientações em painel destacado após o limite de tentativas ser atingido. O histórico de tentativas anteriores (Figura 14) implementa a história de usuário correspondente, permitindo ao aluno revisar todas as suas submissões daquele exercício.

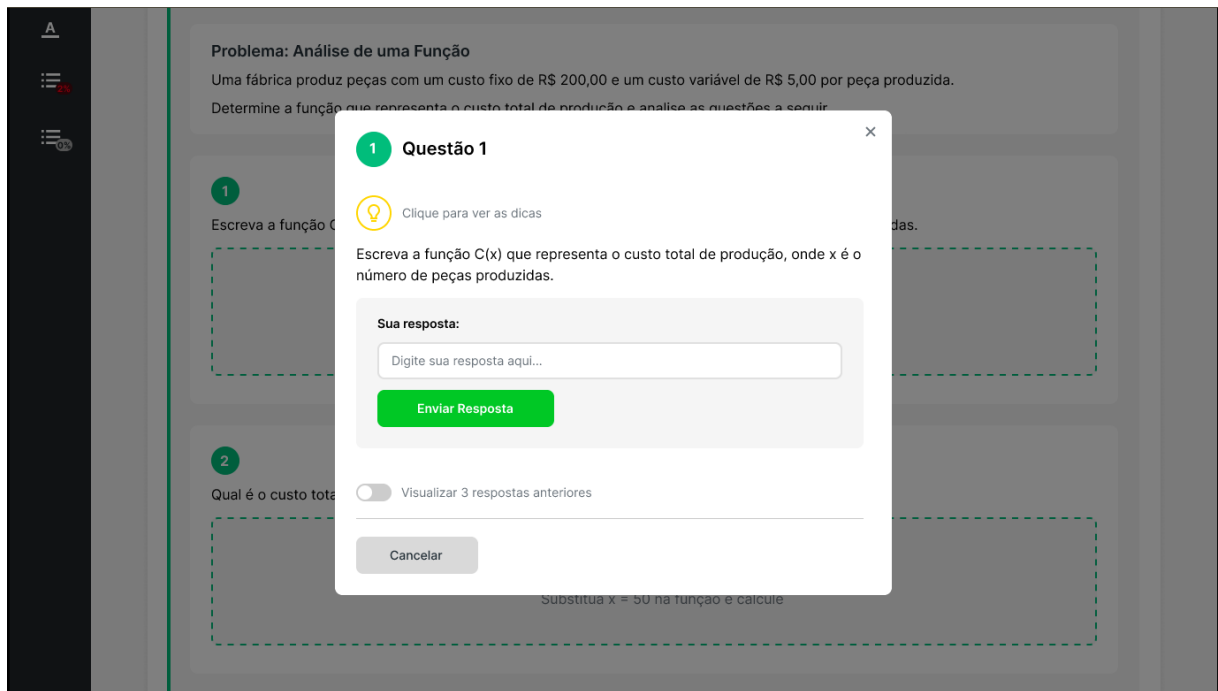


Figura 12 – Modal para Inserção de Resposta do Exercício

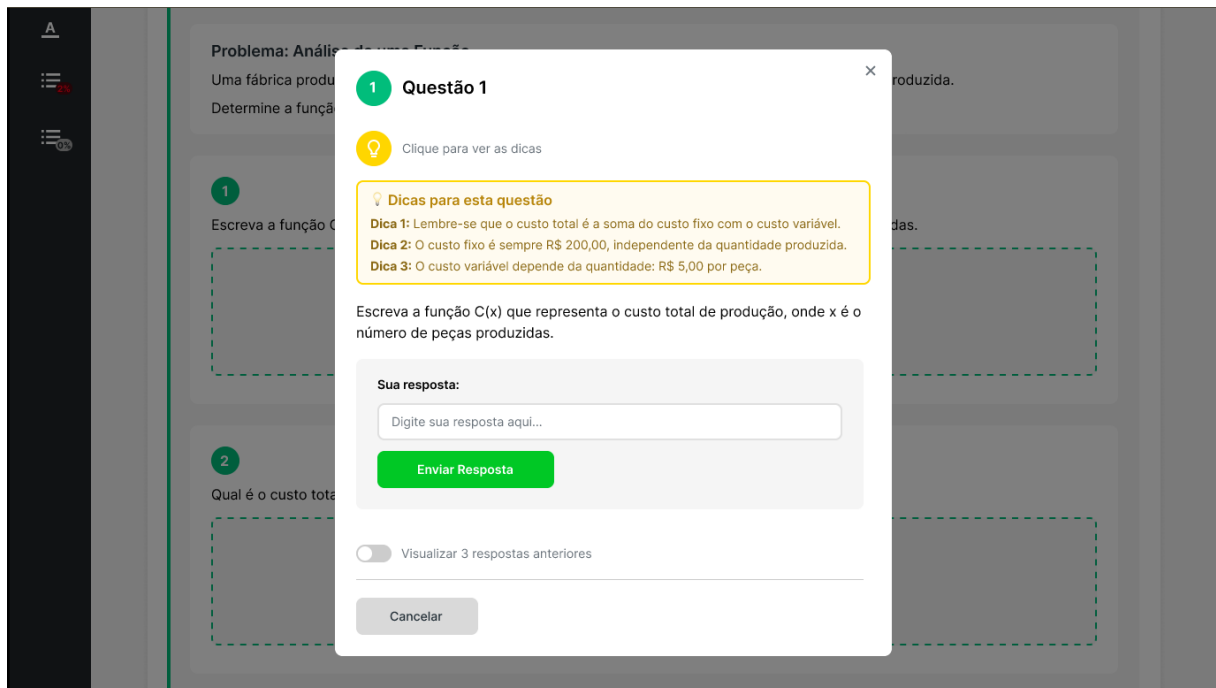


Figura 13 – Visualização das Dicas no Modal de Resposta

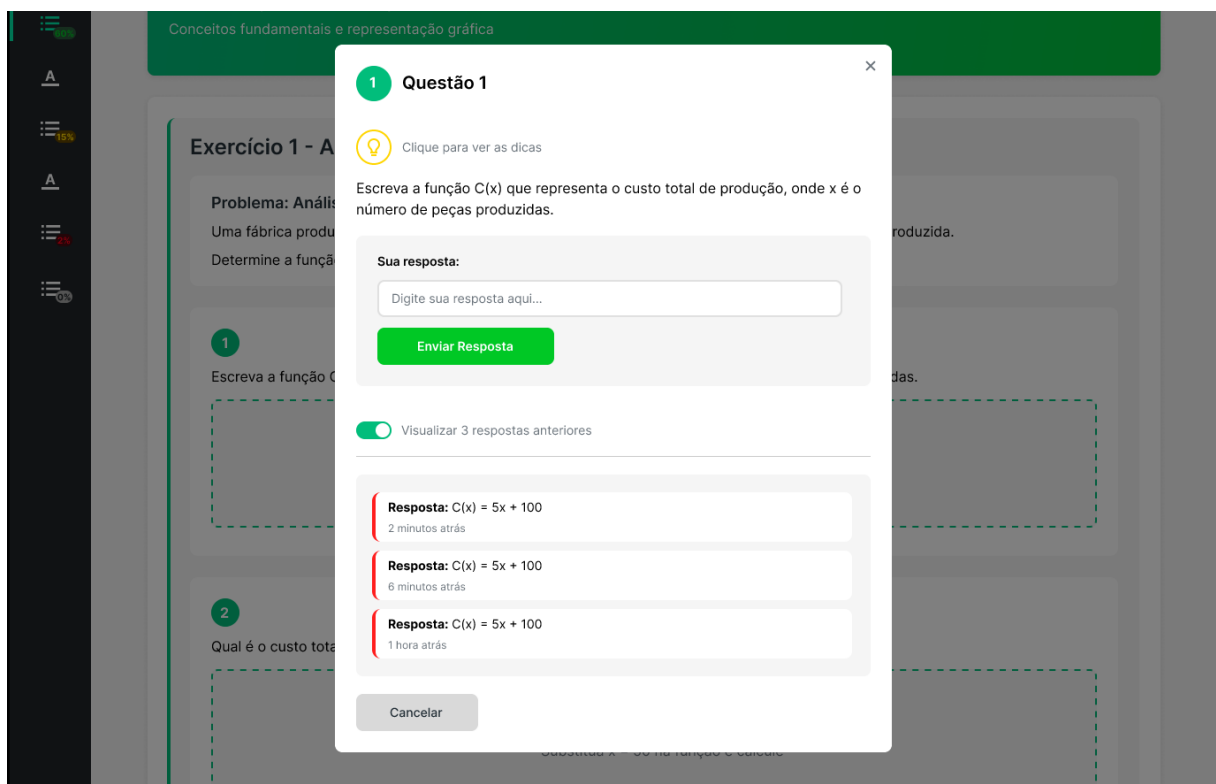


Figura 14 – Visualização das Respostas Anteriores e Seus Resultados

A barra lateral (Figura 15) implementa a história de usuário "*Visualizar progresso no OA*", apresentando uma navegação e monitoramento que atende ao requisito de "mostrar o status das questões e estatísticas de conclusão". Cada seção (introduções e exercícios) possui indi-

cadadores visuais de status usando um sistema de cores que realiza a necessidade expressada na história de usuário de permitir ao aluno "saber quais questões já respondi corretamente":

- Verde: conteúdo concluído com sucesso
- Amarelo: conteúdo visualizado mas com respostas incorretas
- Vermelho: tentativas realizadas com erros
- Cinza: conteúdo ainda não acessado

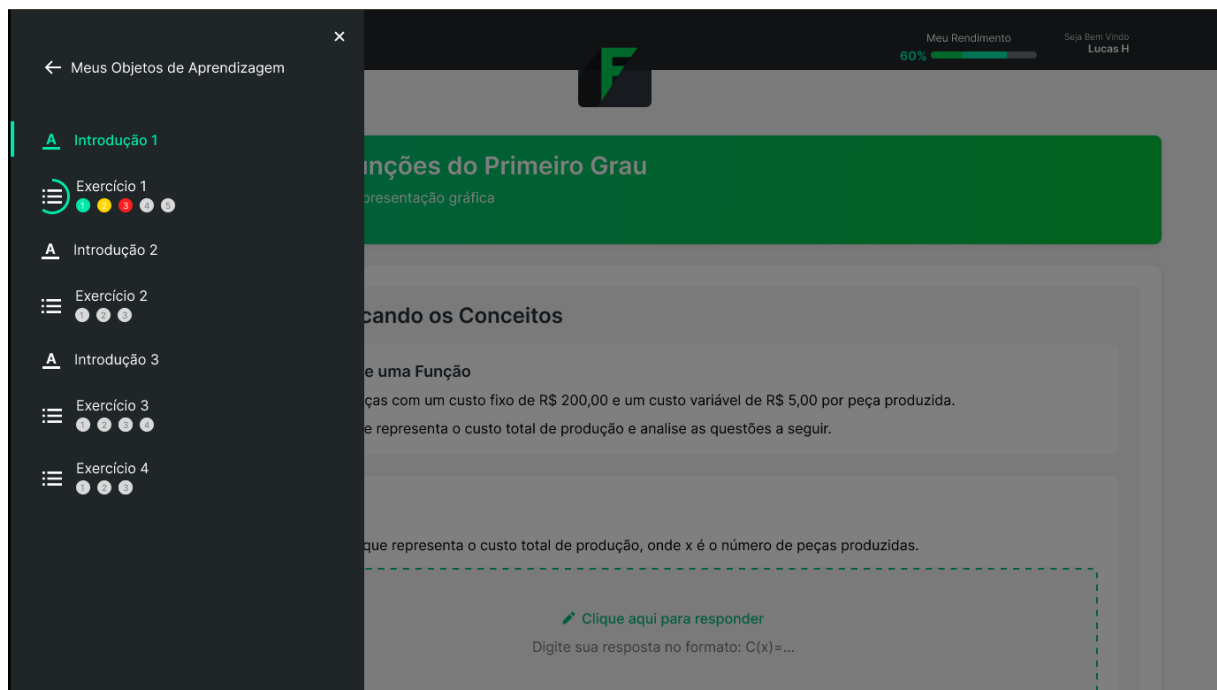


Figura 15 – Sistema de Progresso Detalhado na Barra Lateral

Esta codificação por cores atende ao apontamento do professor sobre a necessidade de o aluno "saber quais questões já respondeu corretamente" de forma imediata. O sistema permite que o estudante identifique onde precisa concentrar seus esforços de estudo, direcionando o tempo de estudo de acordo com seu desempenho individual.

Para os exercícios, círculos menores representam cada questão individual, permitindo que o aluno veja não apenas o progresso geral do exercício, mas também o status de cada questão. Esta organização de informação permite que o estudante planeje sua sessão de estudos com base em dados sobre seu desempenho.

O sistema de dicas (Figura 13) foi projetado para aparecer em momentos pedagogicamente apropriados, seguindo o número de tentativas configurado pelo professor. As dicas são apresentadas em um painel destacado visualmente, com fundo amarelo que as distingue claramente do conteúdo principal da questão.

A decisão de não mostrar as dicas imediatamente reflete a abordagem pedagógica da FARMA: o aluno deve primeiro tentar resolver o problema utilizando seus conhecimentos, desenvolvendo autonomia intelectual através da tentativa e erro. As dicas servem como um apoio que aparece quando o estudante demonstra necessidade de suporte adicional.

O recurso "Visualizar tentativas anteriores"(Figura 14) permite que o aluno acesse seu histórico completo de interações com cada questão, incluindo há quanto tempo o aluno realizou essa tentativa. Esta funcionalidade atende diretamente ao princípio central da FARMA de aprendizagem através da análise de erros, permitindo que o estudante revise seu processo de raciocínio e identifique padrões em suas dificuldades.

No canto superior direito, o indicador "Meu Rendimento: 60%" fornece uma visão consolidada do desempenho geral do aluno no OA. Este elemento responde à necessidade identificada pelo professor de apresentar o progresso de forma visível, permitindo que o aluno tenha uma percepção do seu avanço geral sem precisar navegar por menus adicionais.



Figura 16 – Dropdown Contendo o Progresso do Aluno no OA

5.2.2 Interface de Visualização do Professor

Conforme destacado por Ramos (2019), a área do professor tem como objetivo apresentar uma pré-visualização do OA durante sua construção, agilizando o processo de desenvolvimento de conteúdo.

A interface do professor apresenta as mesmas características visuais da interface do aluno, permitindo que o docente experimente a mesma jornada de aprendizagem que seus estudantes vivenciarão. Esta consistência visual permite que o professor avalie a experiência pedagógica projetada.

No entanto, existem funcionalidades específicas disponíveis apenas para o professor: acesso completo a todas as dicas configuradas independentemente do número de tentativas, visualização de estatísticas de desempenho da turma, e capacidade de testar todas as respostas corretas para validar o funcionamento adequado dos exercícios. Estas funcionalidades adicionais não alteram a apresentação visual básica do OA, mantendo a autenticidade da experiência de pré-visualização.

O professor também pode acessar dados de desempenho individual dos alunos e estatísticas consolidadas da turma, permitindo identificar questões que geram maior dificuldade e ajustar estratégias pedagógicas com base em dados concretos de desempenho. Esta capacidade de monitoramento transforma a ferramenta de um meio de entrega de conteúdo em instrumento de avaliação formativa continuada.

5.2.3 Interface de Visualização do Visitante

A interface do visitante segue o mesmo padrão visual e funcional da interface do aluno, permitindo exploração completa dos OAs disponibilizados publicamente pelos professores. Conforme estabelecido por Ramos (2019), a área pública oferece possibilidade de acessar os OAs sem vínculo a uma turma, sendo que as interações realizadas nesta modalidade não são registradas no sistema.

O visitante pode navegar por todas as introduções teóricas, resolver exercícios, receber feedback de erro ou acerto, e acessar dicas após o número configurado de tentativas incorretas. Esta abrangência permite que usuários não cadastrados explorem o potencial pedagógico da ferramenta e avaliem a qualidade dos conteúdos disponibilizados.

A principal diferença reside na ausência de persistência de dados: todo o progresso do visitante é mantido apenas em memória durante a sessão ativa. Quando o visitante encerra sua interação com o OA, todas as informações de progresso, tentativas e navegação são descartadas. Esta abordagem permite demonstrar as capacidades da ferramenta sem comprometer a integridade dos dados do sistema ou criar sobrecarga desnecessária no armazenamento.

6 CONSIDERAÇÕES FINAIS

O projeto se destaca por dar continuidade ao desenvolvimento de uma ferramenta já consolidada no contexto acadêmico, com múltiplos estudos, Trabalho de Conclusão de Curso (TCC)s e doutorados realizados ao longo dos anos. Ao aprimorar a visualização dos OAs, o projeto busca proporcionar ao aluno um ambiente mais didático, onde ele possa acompanhar seu próprio desempenho e utilizar seus erros como parte do processo de construção do conhecimento, sem que isso represente uma barreira. Além disso, a adoção de tecnologias atualizadas fortalece a base técnica do sistema e contribui para sua manutenção a longo prazo.

Espera-se que os resultados obtidos com este trabalho contribuam tanto para o avanço técnico da ferramenta quanto para sua aplicação prática em contextos reais de ensino. A nova visualização poderá ser utilizada por alunos e professores no dia a dia, além de servir como base para futuras contribuições acadêmicas, incluindo trabalhos voltados à acessibilidade, testes de usabilidade, ou extensão do sistema para outras áreas do conhecimento além da matemática.

REFERÊNCIAS

- ARRUDA, S. d. M. Mudanças nas ideias prévias sobre ponto, reta e plano por meio da interação com a ferramenta farma. **Revista Brasileira de Ensino de Ciência e Tecnologia**, 2021.
- BAZZO, G. **CLASSIFICAÇÃO AUTOMÁTICA DE ERROS DE APRENDIZES HUMANOS DO PROCESSO DE INDUÇÃO ANALÍTICA**. 2011. Tese (Doutorado) — Universidade Federal do Paraná, Curitiba - Paraná, 2011.
- BONIN, D. **IMPLEMENTAÇÃO DE UMA API PARA VISUALIZAÇÃO E INTERAÇÃO COM OBJETOS DE APRENDIZAGEM PARA FERRAMENTA DE AUTORIA FARMA**. 2024. Disponível em: https://tcc.tsi.pro.br/uploads/academic_activity/pdf/273/GP_COINT_2024_1_DOUGLAS_VINICIUS_CALDAS_BONIN_MONOGRAFIA.pdf.
- CASTILHO, A. S. d. **Objetos de Aprendizagem com Feedbacks para a Autorregulação da Aprendizagem de Conceitos Matemáticos Necessários para o Cálculo Diferencial e Integral**. 2019. Tese (Doutorado) — Universidade Federal do Paraná, Curitiba - Paraná, 2019.
- COSTA, D. B. B. d. **ATAUDIW - Uma Ferramenta de Autoria para Auxiliar o Uso da Lousa Digital Interativa**. 2017. Tese (Doutorado) — Universidade Federal de Goiás - Regional Jataí Instituto de Ciências Exatas e Tecnológicas (ICET), Goiás, 2017.
- KUTZKE, A. R. **Informática educacional e a mediação do erro na educação : um estudo teórico-crítico e uma proposta de instrumento computacional**. 2015. Tese (Doutorado) — Universidade Federal do Paraná, Curitiba - Paraná, 2015.
- LEITE, M. D. **Arquitetura para remediação de erros baseada em múltiplas representações externas**. 2013. Tese (Doutorado) — Universidade Federal do Paraná, Curitiba - Paraná, 2013.
- LEITE, M. D. Otimizando o processo ensino e aprendizagem com a arquitetura para desenvolvimento de objetos de aprendizagem - (adoa). **Anais do XXVI Simpósio Brasileiro de Informática na Educação (SBIE 2015)**, 2015.
- LI, Z. **A systematic mapping study on technical debt and its management**. 2014. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0164121214002854>. Acesso em: 05 may. 2025.
- MARCZAL, D. **UM ARCABOUÇO QUE ENFATIZA A RETROAÇÃO**. 2010. Tese (Doutorado) — Universidade Federal do Paraná, Curitiba - Paraná, 2010. Disponível em: <https://github.com/dmarczal/FARMA/blob/master/app/assets/files/ctd-ie-artigo-2010.pdf>.
- MARCZAL, D. **FARMA: UMA FERRAMENTA DE AUTORIA PARA OBJETOS DE APRENDIZAGEM DE CONCEITOS MATEMÁTICOS**. 2014. Tese (Doutorado) — Universidade Federal do Paraná, Curitiba - Paraná, 2014. Disponível em: <https://acervodigital.ufpr.br/handle/1884/37363>.
- MARCZAL, D. Farma: Uma ferramenta de autoria para objetos de aprendizagem de conceitos matemáticos. **C3SL – Departamento de Informática – Universidade Federal do Paraná(UPFR)**, 2015.
- MARCZAL, D. Metodologia e software educacional para a investigação e remediação de erros conceituais em matemática. **Revista Brasileira de Informática na Educação**, 2016.

MOURA, V. A. B. d. **AVALIAÇÃO DO IMPACTO DA RETROAÇÃO NA APRENDIZAGEM APOIADA POR UMA FERRAMENTA EDUCACIONAL**. 2015. Tese (Doutorado) — Universidade Federal do Paraná, Curitiba - Paraná, 2015.

RAMOS, J. **VISUALIZAÇÃO DE OBJETOS DE APRENDIZAGEM PARA A FERRAMENTA DE AUTORIA FARMA**. 2019. Disponível em: https://tcc.tsi.pro.br/uploads/academic_activity/pdf/19/GP_COINT_2019_2_JEFFERSON_HENRIQUE_RAMOS_MONOGRAFIA.pdf.

RIBAS, G. H. **DESENVOLVIMENTO DA INTERFACE GRÁFICA PARA CRIAÇÃO DE OBJETOS DE APRENDIZAGEM NA FERRAMENTA FARMA**. 2024. Disponível em: https://tcc.tsi.pro.br/uploads/academic_activity/pdf/329/GP_COINT_2024_2_GABRIEL_HENRIQUE_RIBAS_PROJETO.pdf. Acesso em: 06 may. 2025.

SANTOS, A. G. D. **DESENVOLVIMENTO DO MÓDULO DE ESTATÍSTICAS DA FERRAMENTA DE AUTORIA DE OBJETOS DE APRENDIZAGEM FARMA**. 2022. Disponível em: https://tcc.tsi.pro.br/uploads/academic_activity/pdf/131/GP_COINT_2022_1_AGUINALDO_GOUDINHO_DOS_SANTOS_MONOGRAFIA.pdf.

SANTOS, E. L. d. **Projeto de uma nova interface gráfica para a ferramenta de autoria de objetos de aprendizagem matemáticos FARMA**. 2018. Tese (Doutorado) — Universidade Tecnológica Federal do Paraná, Guarapuava - Paraná, 2018.

SILVA, R. C. **SEQUENCIAMENTO ADAPTATIVO DE EXERCÍCIOS BASEADO NA CORRESPONDÊNCIA ENTRE A DIFICULDADE DA SOLUÇÃO E O DESEMPENHO DINÂMICO DO APRENDIZ**. 2015. Tese (Doutorado) — Universidade Federal do Paraná, Curitiba - Paraná, 2015.

SILVA, R. C. Adaptabilidade de objetos de aprendizagem usando calibragem e sequenciamento adaptativo de exercícios. **Revista Brasileira de Informática na Educação – RBIE**, 2018.

TRIBECK, P. M. d. A. Edutracsys como objeto de aprendizagem na remediação de erros. **Espacios**, 2017.

VITEK, I. **DESENVOLVIMENTO DA ÁREA DESTINADA À CRIAÇÃO DE OBJETOS DE APRENDIZAGEM PARA FERRAMENTA DE AUTORIA FARMA**. 2023. Disponível em: https://tcc.tsi.pro.br/uploads/academic_activity/pdf/231/GP_COINT_2023_2_ISABELA_TQUES_VITEK_MONOGRAFIA.pdf.