

CENTRO PAULA SOUZA

Etec Professora Maria Cristina Medeiros

**Ensino Médio com Habilitação Profissional de Técnico em
Informática para Internet**

Amanda Kaori Shinohara

Ângelo Gabriel Araújo de Souza

Cecília Santiago Faria

Denner Pereira dos Santos

Enzo Mobile de Oliveira

**OTTO: Plataforma para o ensino de lógica de programação para
crianças**

Ribeirão Pires

2025

Amanda Kaori Shinohara
Ângelo Gabriel Araújo de Souza
Cecília Santiago Faria
Denner Pereira dos Santos
Enzo Mobile de Oliveira

**OTTO: Plataforma para o ensino de lógica de programação para
crianças**

Trabalho de Conclusão de Curso apresentado ao Curso Ensino Médio com Habilitação Profissional de Técnico em Informática para Internet da ETEC Professora Maria Cristina Medeiros, orientado pela Profª Ma Suely dos Santos Sousa, como requisito parcial para obtenção do título de Técnico em Informática para Internet.

Ribeirão Pires
2025

FICHA CATALOGRÁFICA
CATALOGAÇÃO CENTRALIZADA
Biblioteca da ETEC Prof.^a Maria Cristina Medeiros

O91

OTTO: Plataforma para o ensino de lógica de programação para crianças / Amanda Kaori Shinohara; Ângelo Gabriel Araújo de Souza; Cecília Santiago Faria; Denner Pereira dos Santos; Enzo Mobile de Oliveira. – Ribeirão Pires (SP): ETEC MCM, 2025. Monografia. 202 fls.

Formato PDF/A. Requisitos de sistema: Adobe Acrobat Reader

Trabalho de Conclusão de Curso (Técnico) – Centro Paula Souza, ETEC Prof.^a Maria Cristina Medeiros, Eixo Informática e Comunicação – Ensino Médio com Habilitação Técnica em Informática para Internet.

Orientador (a): Prof.^a Ma. Suely dos Santos Souza

Depósito: Repositório Institucional do Conhecimento do Centro Paula Souza

Modo de acesso: <https://ric.cps.sp.gov.br>

1. Lógica de Programação 2. Gamificação 3. Inclusão Digital 4. Educação Tecnológica 5. Programação em Blocos

I. Título II. SHINOHARA, Amanda Kaori III. SOUZA, Ângelo Gabriel Araújo de
IV FARIA, Cecília Santiago V. SANTOS, Denner Pereira dos VI. OLIVEIRA, Enzo
Mobile de

CDD 005.36

Elaborado Por: Patricia Cordeiro da Silva Farias – CRB-8/7510

**Amanda Kaori Shinohara
Ângelo Gabriel Araújo de Souza
Cecília Santiago Faria
Denner Pereira dos Santos
Enzo Mobile de Oliveira**

OTTO

Plataforma para o ensino de lógica de programação para crianças

Trabalho de Conclusão de Curso apresentado ao Curso Ensino Médio com Habilitação Profissional de Técnico em Informática para Internet da ETEC Professora Maria Cristina Medeiros, como requisito parcial para obtenção do título de Técnico em Informática para Internet.

Aprovado em: ____/____/____

BANCA EXAMINADORA:

Prof. Ma Suely dos Santos Sousa - Orientador(a)

Leticia Lira da Silva - Examinador(a)

Prof. Ricardo Moreira - Examinador(a)

Prof. Rogério Carneiro Leal - Examinador(a)

RESUMO

Este trabalho apresenta o desenvolvimento da plataforma OTTO, um site interativo para o ensino de lógica de programação destinado a crianças de 8 a 12 anos. A proposta surge da necessidade de enfrentar a exclusão digital e o desinteresse de jovens pela área de tecnologia, causados principalmente pelo alto custo de plataformas educacionais e pela percepção de complexidade das disciplinas de exatas. O objetivo geral é promover o acesso inclusivo e lúdico à programação em blocos e incorporar elementos de gamificação e interatividade. A fundamentação teórica aborda conceitos de lógica de programação, pensamento computacional, construcionismo e tecnologias educacionais. A metodologia envolve o uso de ferramentas de design, prototipagem e inteligência artificial para o desenvolvimento da plataforma. Como resultados esperados, espera-se que a plataforma beneficie crianças da região do ABC Paulista, promovendo o desenvolvimento de raciocínio lógico, pensamento crítico e criatividade, além de contribuir para o Objetivo de Desenvolvimento Sustentável (ODS) 04 da ONU, que visa garantir educação inclusiva e de qualidade.

Palavras-Chave: Lógica de Programação; Gamificação; Inclusão Digital; Educação Tecnológica; Programação em Blocos.

ABSTRACT

This paper presents the development of the OTTO platform, an interactive website for teaching programming logic to children aged 8 to 12. The proposal arises from the need to address digital exclusion and young people's lack of interest in technology, caused mainly by the high cost of educational platforms and the perception of complexity in science subjects. The overall objective is to promote inclusive and playful access to block-based programming and incorporate elements of gamification and interactivity. The theoretical foundation addresses concepts of programming logic, computational thinking, constructionism, and educational technologies. The methodology involves the use of design, prototyping, and artificial intelligence tools for the development of the platform. As expected results, the platform is expected to benefit children in the ABC Paulista region, promoting the development of logical reasoning, critical thinking, and creativity, in addition to contributing to the UN Sustainable Development Goal (SDG) 04, which aims to ensure inclusive and quality education.

Keywords: Programming Logic; Block Programming; Gamification; Digital Inclusion; Technology Education.

LISTA DE ILUSTRAÇÕES

Figura 1 - Interface de criação do Scratch.....	18
Figura 2 - Fase 01 da Plataforma Otto.....	18
Figura 3 - Protótipo do figma.....	31
Figura 4 - ODS 04.....	35
Figura 5 - Fase 1 da plataforma OTTO com o desafio resolvido.....	37
Figura 6 - Fase 2 da plataforma OTTO com o desafio resolvido.....	37
Figura 7 - Fase 3 da plataforma OTTO com o desafio resolvido.....	38
Figura 8 - Fase 4 da plataforma OTTO com o desafio resolvido.....	39
Figura 9 - Fase 5 da plataforma OTTO com o desafio resolvido.....	40
Figura 10 - Fase 6 da plataforma OTTO com o desafio resolvido.....	40
Figura 11 - Fase 7 da plataforma OTTO com o desafio resolvido.....	41
Figura 12 - Fase 8 da plataforma OTTO com o desafio resolvido.....	42
Figura 13 - Fase 9 da plataforma OTTO com o desafio resolvido.....	43
Figura 14 - Fase 10 da plataforma OTTO com o desafio resolvido.....	43
Figura 15 - Logotipo.....	45
Figura 16 - Personagem Otto.....	45
Figura 17 - Distribuição por faixa etária dos respondentes.....	47
Figura 18 - Uso de plataformas de programação.....	48
Figura 19 - Interesse da criança por novas tecnologias.....	48
Figura 20 - Percepção sobre o benefício de plataformas de programação interativa.....	49
Figura 21 - Avaliação geral da experiência na Plataforma Otto.....	51
Figura 22 - Curiosidade e vontade de continuar jogando/aprendendo na Plataforma.....	51
Figura 23 - Contribuição das atividades para o entendimento da lógica de programação.....	52
Figura 24 - Contribuição percebida da Plataforma Otto para a ODS 4 (Educação de Qualidade).....	52

Figura 25 - Percepção sobre a acessibilidade da Plataforma para diferentes perfis de usuários.....	53
Figura 26 - Avaliação do visual da Plataforma (cores, personagens, blocos, etc.)....	53
Figura 27 - Entendimento da lógica de programação ensinada em blocos para crianças.....	54
Figura 28 - Intenção de reutilização e recomendação da Plataforma Otto.....	54

SUMÁRIO

1 INTRODUÇÃO	10
1.1 Tema	11
1.2 Problema.....	11
1.3 Justificativa	12
1.4 Objetivo.....	13
1.4.1 Objetivo geral	13
1.4.2 Objetivos específicos	13
1.5 Resultados esperados	13
2 FUNDAMENTAÇÃO TEÓRICA	14
2.1 Lógica de Programação.....	14
2.1.1 Definição e importância.....	14
2.1.2 Lógica como base para o pensamento computacional.....	14
2.2 Ensino de Programação para Crianças	15
2.2.1 Abordagem pedagógica: Construtivismo e Construcionismo.....	15
2.2.2 Faixa etária	17
2.2.3 Scratch.....	17
2.3 Tecnologias Educacionais	19
2.3.1 Gamificação e interatividade.....	19
3 METODOLOGIA	22
3.1 Pesquisa	22
3.1.1 Pesquisa de campo da plataforma.....	22
3.1.2 Análise de resultados da plataforma	23
3.2 Etapas do desenvolvimento da plataforma OTTO	23
3.2.1 Prototipagem	23
3.2.2 Front-end	24
3.2.2.1 HyperText Markup Language	25
3.2.2.2 Cascading Style Sheets	25
3.2.2.3 Linguagem de Programação JavaScript	26
3.2.3 Back-end.....	27

3.2.3.1 Framework web Django	27
3.2.4 Banco de dados	28
3.2.4.1 Gerenciador de banco de dados (PostgreSQL)	28
3.2.4.2 HeidiSQL	29
3.2.5 API e Integrações	29
3.2.5.1 Blockly	29
3.2.5.2 Google Identity Services.....	30
3.3 Ferramentas e tecnologias utilizadas	30
3.3.1 Figma	30
3.3.2 ChatGPT.....	31
3.3.3 Canva	31
3.3.4 Visual Studio Code.....	32
3.3.5 Github	32
3.3.6 Git.....	33
3.3.7 Microsoft Azure	33
4 DESENVOLVIMENTO DA PLATAFORMA OTTO.....	35
4.1 Objetivos da plataforma	35
4.2 Jogos e Habilidades.....	36
4.2.1 Bom Dia, Polvo!	36
4.2.2 Hora do Café do Polvo	37
4.2.3 Otto na Escola	38
4.2.4 Dança do Polvo Feliz	38
4.2.5 Missão Mercado: Polvo Organizado	39
4.2.6 Hora do Polvo Limpinho	40
4.2.7 Restaurante Tentáculos.....	41
4.2.8 Super Treino do Polvo	41
4.2.9 Aventuras na Cidade do Otto!	42
4.2.10 Projeto final	43
4.3 Funcionalidades principais, interface e usabilidade para crianças	44
4.3.1 Funcionalidades Principais	44
4.3.2 Identidade Visual.....	44
4.3.3 Acessibilidade	46

5 AVALIAÇÃO E RESULTADOS	47
5.1 Análise dos resultados obtidos: Pesquisa de Campo da plataforma	47
5.2 Feedback de usuários.....	50
6 CONSIDERAÇÕES FINAIS	56
REFERÊNCIAS BIBLIOGRÁFICAS	59
APÊNDICE A – PESQUISA SOBRE A PLATAFORMA OTTO	64
APÊNDICE B – ANÁLISE DE RESULTADOS DA PLATAFORMA OTTO.....	66

1 INTRODUÇÃO

Cada vez mais, as habilidades relacionadas à área de tecnologia estão sendo requisitadas, impulsionadas pelo atual cenário de modernização digital. Segundo relatório da Brasscom (2025), o setor de Tecnologia, Informação e Comunicação (TIC) no Brasil deve gerar entre 30 mil e 147 mil empregos formais até o final de 2025, sendo 57% desses postos diretamente ligados à área de tecnologia. Esse crescimento evidencia a crescente demanda por profissionais qualificados na área. No entanto, ainda existem diversos obstáculos para que esse conhecimento tecnológico seja efetivamente acessível e incorporado por toda a população.

Atualmente, muitas das plataformas disponíveis são pagas ou necessitam de equipamentos avançados, gerando uma barreira significativa para pessoas com baixa renda que dificulta o acesso a recursos de qualidade e inserção ao mercado de trabalho. A necessidade de plataformas envolventes e a visão social negativa sobre exatas também são desafios a serem superados (AGÊNCIA BRASIL, 2023; INSTITUTO DE PESQUISA ECONÔMICA APLICADA, 2024).

Além da resolução dessas barreiras, a plataforma tem o intuito de que os usuários possam ter a possibilidade de desenvolver habilidades como lógica, pensamento crítico e resolução de problemas do cotidiano. Elas podem ter a visão do funcionamento de tecnologias do dia a dia, já que os usuários têm contato em aplicações do cotidiano, logo, tem-se uma abertura de aprendizado.

Tendo em vista, este trabalho visa a criação de um site interativo para ensinar lógica de programação de forma divertida a crianças. Tendo como objetivo incentivar o interesse e o aprendizado de programação através da Programação em Blocos e a linguagem de programação Python.

1.1 Tema

Este trabalho trata do desenvolvimento de um site interativo para ensinar lógica de programação de forma divertida e acessível para público infantil de 8 a 12 anos. Com o principal objetivo de incentivar o público alvo a ter conhecimento e interesse nessa área. A plataforma trabalha por meio da Programação em Blocos por meio do Scratch, linguagem de programação visual desenvolvida pelo Instituto de Tecnologia de Massachusetts (MIT).

1.2 Problema

Um dos problemas presentes é a exclusão digital, que limita o acesso a recursos educacionais de qualidade em Tecnologia da Informação (TI), especialmente para jovens de baixa renda, segundo a pesquisa TIC Kids Online Brasil, do Comitê Gestor da Internet no Brasil (CGI.br), divulgada em 26 de outubro de 2023. Embora 95% das crianças e adolescentes entre 9 e 17 anos tenham acesso à internet, o estudo aponta que mais de 580 mil jovens dessa faixa etária nunca acessaram a rede. A predominância de plataformas pagas e a exigência de equipamentos avançados restringem o aprendizado e aprofundam disparidades sociais.

Adicionalmente, outro desafio significativo é o desinteresse na área de Tecnologia observado na experiência escolar de crianças e adolescentes. Essa indiferença é alimentada pela percepção de que a área é complexa, inacessível e difícil, o que pode desmotivar o aprendizado desde o início. Essa crença pode afastá-los da exploração dessas disciplinas, limitando o desenvolvimento de habilidades essenciais (AGÊNCIA BRASIL, 2023; INSTITUTO DE PESQUISA ECONÔMICA APLICADA, 2024).

1.3 Justificativa

O ensino de programação desde a infância é uma maneira para a ampliação do acesso ao conhecimento na área de Tecnologia, além de desenvolver habilidades como lógica, pensamento crítico e resolução de problemas. Ao aprender a programar, os jovens não só adquirem aprendizagens, mas também desmistificam a ideia de que a tecnologia é algo complexo.

Uma das maiores barreiras que muitos enfrentam ao lidar com áreas ligadas a exatas e tecnologia é a percepção de que são difíceis ou inacessíveis. O ensino de programação, entretanto, pode quebrar essa barreira. Ao ensinar programação de forma didática e interativa é despertado o interesse de jovens na área.

O computador é uma ferramenta que propicia à criança as condições de entrar em contato com algumas das mais profundas ideias em ciência, matemática e criação de modelos. Segundo a filosofia Logo, o aprendizado acontece através do processo de a criança inteligente 'ensinar' o computador burro, ao invés de o computador inteligente ensinar a criança burra. (PAPERT, 1985, p. 9).

De acordo com uma pesquisa feita entre 3 e 9 de maio de 2024 pelo Senai (Serviço Nacional de Aprendizagem Industrial) mostra que 43% dos jovens brasileiros têm pouco ou nenhum interesse em fazer um curso técnico, ou profissionalizante. Esse dado revela a necessidade de despertar o interesse dos jovens pela área tecnológica, tanto para quem deseja seguir uma carreira no setor, quanto para aqueles que buscam adquirir um conhecimento para a vida.

Diante desse contexto, este trabalho propõe o desenvolvimento de um site interativo de ensino a programação em blocos, conseqüentemente incentivando o interesse pela área. A fim de auxiliar na disseminação de conhecimento do âmbito tecnológico. Dessa maneira, a plataforma contribui com o Objetivo de Desenvolvimento Sustentável 04, que visa garantir a educação de qualidade.

1.4 Objetivo

1.4.1 Objetivo geral

Desenvolver uma plataforma lúdica para ensinar lógica de programação. O site tem como propósito incentivar o interesse e a compreensão da área de Tecnologia desde a infância. De modo que contribua com a ODS 04, que visa garantir uma educação de qualidade, inclusiva e equitativa.

1.4.2 Objetivos específicos

- ✓ Atingir jovens entre 8 a 12 anos;
- ✓ Ampliar o acesso à educação tecnológica, promovendo a inclusão digital de forma equitativa e democratizada;
- ✓ Desenvolver habilidades nos usuários de resolução de problemas, raciocínio lógico e criatividade por meio de desafios e atividades lúdicas.

1.5 Resultados esperados

Espera-se que a plataforma atenda, em sua fase inicial, crianças e jovens localizados na região do ABC Paulista, com prioridade em escolas e instituições sociais. Estima-se alcançar, nesta primeira etapa, cerca de 1.000 jovens dessa população, de acordo com um levantamento de dados da Fundação Seade, gerenciada pelo governo do Estado, a região em 2025, que abrange sete municípios: Santo André, São Bernardo do Campo, São Caetano do Sul, Diadema, Mauá, Ribeirão Pires e Rio Grande da Serra, totaliza aproximadamente 486.820 de crianças.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 Lógica de Programação

2.1.1 Definição e importância

A lógica de programação é a capacidade crucial de organizar instruções de maneira clara e coerente. É o processo mental de converter um problema em algoritmos capazes de guiar o computador na execução de suas funções. O raciocínio lógico permite transformar uma necessidade complexa em uma solução sistemática, utilizando comandos desde a entrada de dados até a obtenção de resultados (MANZANO; OLIVEIRA, 2019).

Esse raciocínio ultrapassa uma forma de linguagem de programação específica. De acordo com Milani, Dutra e Machado (2024), o foco não está em decorar a sintaxe, mas sim de como as etapas para resolver um problema são organizadas. Seja com variáveis, estruturas de repetição, condições ou funções, a lógica oferece controle sobre o fluxo do programa. A capacidade de pensar passo a passo permite criar soluções complexas.

No ensino de programação, representações como fluxogramas são úteis. Elas auxiliam a criança a criar o raciocínio algorítmico antes de usar linguagens mais complexas. Plataformas visuais como o Scratch também auxiliam nesse aprendizado. Com blocos intuitivos, o jovem visualiza a lógica dos comandos e entende como as ações geram resultados. Essas ferramentas tornam a lógica de programação acessível, preparando o estudante para solucionar problemas na área de tecnologia. A lógica de programação, por sua vez, é a habilidade de sequenciar esses algoritmos de uma maneira que o computador possa executar (MANZANO; OLIVEIRA, 2019).

2.1.2 Lógica como base para o pensamento computacional

A lógica é o alicerce do pensamento computacional, pois permite organizar o raciocínio em etapas claras e ordenadas. É por meio dela que problemas complexos podem ser divididos em partes menores e mais fáceis de compreender, dando origem à formulação de algoritmos. Segundo Vargas, Da Silva e Finger (2022), o ensino da

lógica matemática é indispensável para que os estudantes desenvolvam competências de abstração e raciocínio algorítmico, elementos fundamentais para o pensamento computacional.

No contexto do ensino de programação, a lógica atua como um elo intermediário entre a formulação do problema e sua implementação em código. Milani, Dutra e Machado (2024) destacam que o uso de pseudocódigo, fluxogramas e exercícios desplugados contribui para que os alunos expressem suas soluções de maneira estruturada, o que facilita a transição para linguagens de programação. Dessa forma, a lógica não se restringe à formalização matemática, mas funciona como um processo que aproxima o raciocínio humano da execução computacional.

Além disso, Paulino Júnior e Kelsen (2024) defendem que o pensamento computacional pode ser empregado como metodologia alternativa no ensino de lógica de programação, tornando-o mais dinâmico e acessível. Essa abordagem favorece o desenvolvimento não apenas do raciocínio lógico, mas também do pensamento crítico e criativo, ampliando as possibilidades de aprendizagem.

Portanto, a lógica pode ser compreendida como o pilar do pensamento computacional, uma vez que sustenta as habilidades de decomposição, reconhecimento de padrões, abstração e construção de algoritmos, essenciais tanto para o aprendizado de programação quanto para a resolução de problemas do cotidiano.

2.2 Ensino de Programação para Crianças

2.2.1 Abordagem pedagógica: Construtivismo e Construcionismo

O Construtivismo, fundamentado principalmente nas ideias de Jean Piaget e aprofundado por outros pensadores como Lev Vygotsky, entende que o aprendizado ocorre de maneira mais eficiente quando o indivíduo constrói seu conhecimento a partir da interação com o meio e com outras pessoas. Diferente de uma aprendizagem passiva, essa abordagem coloca o aluno como um agente ativo do processo de aprendizagem, enquanto o professor atua como facilitador da construção do conteúdo.

O construtivismo propõe que o aluno participe ativamente do próprio aprendizado, mediante a experimentação, a pesquisa em grupo, o

estímulo a dúvida e o desenvolvimento do raciocínio, entre outros procedimentos. A partir de sua ação, vai estabelecendo as propriedades dos objetos e construindo as características do mundo. As escolas que usam o método construtivista têm características de estrutura diferentes das escolas que usam outros métodos. (POZO, 1994, p. 24).

O Construcionismo, proposto por Seymour Papert, baseia-se na ideia de que o aprendizado ocorre de forma mais eficaz quando o indivíduo está envolvido na construção de algo significativo. Diferente da simples transmissão de conteúdo, essa abordagem valoriza a experimentação, a exploração e a criação, colocando o aluno como protagonista do processo de aprendizagem.

[...] quando a criança aprende a programar, o processo de aprendizagem é transformado. [...] A criança faz alguma coisa com ele. O novo conhecimento é fonte de poder e é experienciado como tal a partir do momento que começa a se formar na mente da criança. (PAPERT, 1985, p.37).

A utilização de uma plataforma interativa baseada em programação em blocos estabelece um diálogo com as abordagens do Construtivismo e do Construcionismo. Enquanto o Construtivismo destaca a importância da construção ativa do conhecimento pelo estudante, o Construcionismo amplia essa perspectiva ao valorizar a aprendizagem por meio da criação de produtos concretos e significativos. Nesse sentido, ao desenvolver projetos na plataforma, as crianças aplicam conceitos de lógica de programação, como sequências e repetições, de maneira lúdica e contextualizada, unindo teoria e prática. Além disso, ambas as abordagens reconhecem o erro como parte essencial do processo de aprendizagem: ao experimentar, reorganizar blocos de código e testar hipóteses, os alunos refletem sobre suas escolhas, buscam novas estratégias e fortalecem sua autonomia. Assim, mais do que aprender programação, os estudantes desenvolvem competências cognitivas, criativas e socioemocionais fundamentais para sua formação integral e para a vida em sociedade.

2.2.2 Faixa etária

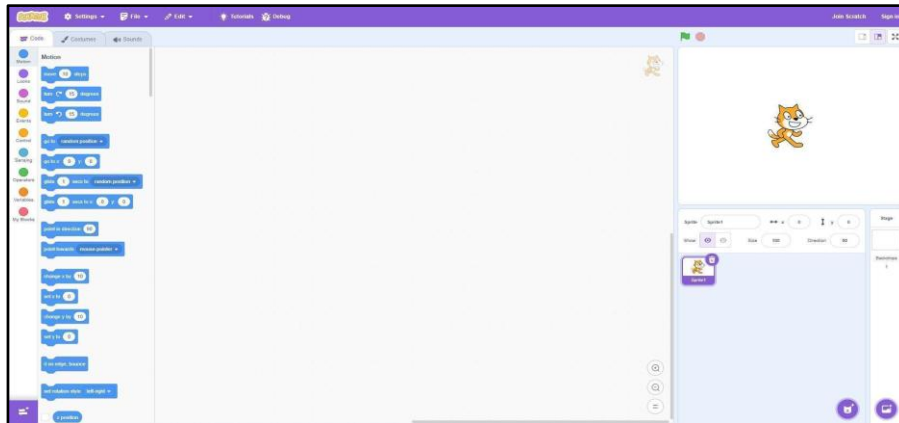
A plataforma é voltada para crianças de 8 a 12 anos, faixa etária em que os estudantes apresentam raciocínio lógico inicial, capacidade de resolver problemas estruturados e curiosidade natural por novas tecnologias. Nessa fase, é possível despertar o interesse das crianças por áreas ainda pouco exploradas, como a programação, estimulando nelas a vontade de aprender e de se envolver com o universo tecnológico. Ao concentrar nesse público, a plataforma busca criar um ambiente de aprendizado acessível, lúdico e motivador, que não apenas ensine conceitos de lógica de programação, mas também incentive o apreço pela área. Por meio do lúdico, a criança canaliza suas energias, vence suas dificuldades, modifica sua realidade, propicia condições de liberação da fantasia e a transforma em uma grande fonte de prazer (PINTO; TAVARES, 2010, p. 226-235,).

2.2.3 Scratch

Para alcançar crianças de forma mais leve e atrativa, este projeto utiliza o Scratch como base, já que trata de uma plataforma de programação visual desenvolvida pelo MIT Media Lab, que foi criada justamente para facilitar o contato inicial com a programação, principalmente entre crianças e jovens, de maneira acessível e divertida. Por ser gratuita e funcionar direto do navegador, ela também se mostra uma opção viável para escolas públicas ou projetos com poucos recursos tecnológicos.

O Scratch possibilita aos usuários aprender por meio de várias competências como: Raciocínio Lógico, Criatividade, Pensamento Sistêmico, Resolução de Problemas, de uma maneira divertida e utilizando a tecnologia, ainda é possível trabalhar de forma colaborativa compartilhando os projetos no site do programa. Os alunos que usam o Scratch aprendem a encaixar blocos como um quebra-cabeça ou um jogo de lego de maneira lógica. (CASTRO, 2017, p.39).

Figura 1 - Interface de criação do Scratch.

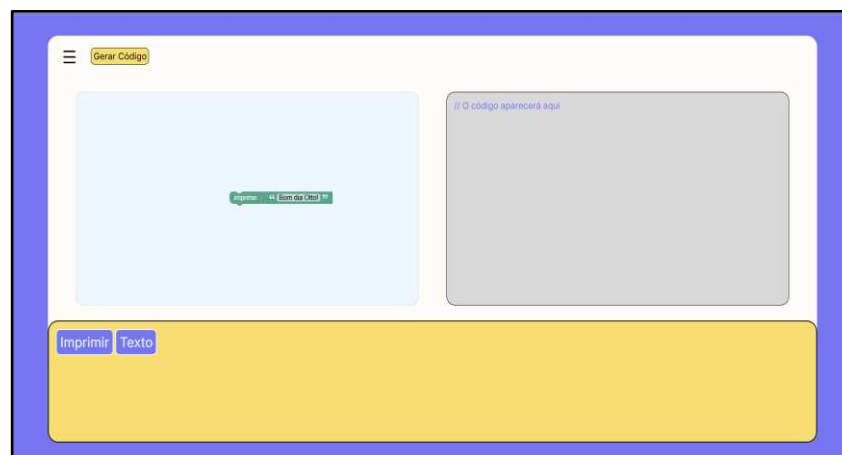


Fonte: Autoria própria, 2025.

Como mostrado na imagem, o funcionamento do Scratch é bem intuitivo. Ao invés de digitar códigos, o usuário apenas arrasta blocos coloridos com comandos e encaixa um no outro para criar sequências. Esses blocos controlam personagens, sons, movimentos e outros elementos dentro do projeto.

No próprio site do Scratch, é explicado que a plataforma foi feita para que crianças possam criar projetos ao mesmo tempo em que aprendem a resolver problemas, usar a lógica e explorar a criatividade (SCRATCH, 2025). Além disso, existe uma comunidade online onde é possível compartilhar os projetos, ver ideias de outras pessoas e até modificar (remixar) trabalhos de colegas, o que contribui para o aprendizado colaborativo.

Figura 2 - Fase 01 da Plataforma Otto.



Fonte: Autoria própria, 2025.

A plataforma também oferece diversos recursos extras, como tutoriais, atividades guiadas, o que facilita sua aplicação tanto em sala de aula quanto em projetos independentes.

No estudo de Bach et al. (2020), os autores destacam que o uso de blocos visuais ajuda crianças em fase de alfabetização a entenderem melhor os conceitos por trás da programação, sem se prenderem à escrita de código. Isso torna o processo mais natural e acessível, principalmente nos primeiros contatos com a lógica computacional.

Pereira et al. (2025) também mostram resultados positivos com o uso do Scratch. Em sua pesquisa, os alunos que aprenderam com a plataforma demonstraram mais motivação, interesse e desempenho melhor quando comparados a métodos mais tradicionais de ensino da programação.

Essa integração vai ao encontro da abordagem STEAM (Ciência, Tecnologia, Engenharia, Artes e Matemática), que valoriza uma aprendizagem mais conectada e criativa. Assim, a plataforma mostra uma aliada não apenas no ensino da tecnologia, mas também no desenvolvimento de competências importantes para o século XXI.

O STEAM é uma proposta de ensino baseada em projetos que relaciona os conteúdos disciplinares a partir de problemas reais, para que, integrados à estrutura de conhecimento do indivíduo, assumam significado em uma situação concreta. Para além da informação lógica, sequencial, estruturada e estática, na sociedade da informação e da comunicação, é preciso aprender novas formas de conhecer, de se comunicar e de integrar aspectos humanos, tecnológicos, individuais e sociais (MAURI, 2009).

2.3 Tecnologias Educacionais

2.3.1 Gamificação e interatividade

A gamificação é uma estratégia pedagógica que utiliza elementos de jogos em contextos de aprendizado para tornar o processo educativo mais envolvente e motivador. Segundo o relatório “The 2019-2024 Global Game-based Learning Market”, da Metaari, elaborado por uma consultoria de análise de mercado especializada em identificar oportunidades para fornecedores de tecnologias de aprendizagem, a estimativa é de que a gamificação no ensino tenha um crescimento de 15,4% de 2019 a 2024. Esse crescimento pode ser explicado pela sua capacidade de transformar o

ensino da lógica de programação em uma experiência lúdica, permitindo que crianças se sintam motivadas a explorar conceitos complexos de forma divertida. Elementos como personalização do Avatar, recompensas e níveis estimulam a continuidade do aprendizado, promovendo o engajamento.

A interatividade é outro ponto central do projeto, pois permite que os usuários aprendam de maneira ativa e prática. Diferentemente de métodos tradicionais de ensino, a plataforma oferece exercícios que exigem participação direta da criança, como arrastar e conectar blocos de programação no Scratch para criar uma parte da história. Esse tipo de abordagem prática contribui para a fixação dos conceitos de lógica computacional.

A interatividade é um dos principais pilares da aprendizagem ativa, pois permite que os alunos se envolvam diretamente no processo de construção do conhecimento. Ao manipular elementos digitais e realizar atividades práticas, os estudantes desenvolvem habilidades cognitivas e metacognitivas que facilitam a compreensão e aplicação dos conteúdos. (SANTANA, 2024).

A junção entre gamificação e interatividade na plataforma cria um ambiente de aprendizado dinâmico, onde o erro é encarado como parte do processo e não como falha. Ao permitir que a criança teste diferentes soluções e receba feedback imediato, a plataforma promove o desenvolvimento de habilidades cognitivas e de resolução de problemas de forma divertida. Essa combinação também auxilia na construção da motivação e curiosidade, incentivando o interesse contínuo pela programação e preparando o público-alvo para futuros estudos na área de tecnologia de maneira lúdica e acessível.

A gamificação insere-se na plataforma como um recurso estratégico para potencializar o engajamento e a motivação dos usuários. A partir de mecanismos como a progressão por níveis, a conquista de emblemas vinculados ao cumprimento de objetivos e a personalização de avatares, estabelece-se um ambiente que valoriza tanto o esforço individual quanto a interação coletiva. Esses elementos contribuem para a construção de uma experiência mais imersiva e significativa, promovendo a permanência na plataforma e favorecendo processos contínuos de aprendizado e participação.

A gamificação no ensino de ciências é uma abordagem pedagógica que utiliza elementos e mecânicas de jogos para melhorar a aprendizagem e o engajamento dos/as alunos/as no estudo das disciplinas científicas. Essa estratégia combina princípios de design de jogos com objetivos educacionais, criando uma experiência de aprendizagem mais envolvente e motivadora (1994, apud SANTANA, 2024).

Dessa forma, unir gamificação e interatividade torna o aprendizado muito mais atrativo e significativo para as crianças. Enquanto os elementos de jogo ajudam a manter o interesse e a motivação, as atividades interativas permitem que elas aprendam “fazendo”, explorando possibilidades e testando suas próprias ideias. Dessa forma, a plataforma cria um ambiente em que aprender programação deixa de ser algo difícil ou distante e passa a ser uma experiência divertida, criativa e envolvente. Ao transformar o erro em parte natural do processo e incentivar a curiosidade, essa combinação contribui para que as crianças se sintam confiantes e animadas para continuar aprendendo e descobrindo o mundo da tecnologia.

3 METODOLOGIA

3.1 Pesquisa

3.1.1 Pesquisa de campo da plataforma

A pesquisa de campo foi realizada em junho de 2025, por meio de um questionário estruturado aplicado no Google Forms, com o objetivo de analisar o que incluir em uma plataforma de programação voltada para crianças de 8 a 12 anos. Esta investigação caracteriza-se como exploratória e descritiva, pois busca compreender o perfil e as percepções de adultos em relação ao ensino de programação infantil, permitindo levantar hipóteses e descrever características da população estudada (Raupp & Beuren, 2006).

Segundo Raupp e Beuren (2006), a pesquisa exploratória é aplicada quando há pouco conhecimento sobre o tema, possibilitando maior familiaridade e clareza quanto ao fenômeno estudado, além de permitir o levantamento de hipóteses. Já a pesquisa descritiva tem como finalidade principal descrever características de determinada população ou fenômeno, estabelecendo relações entre variáveis e permitindo a análise de comportamentos, opiniões e situações observadas.

Quanto à abordagem, a pesquisa foi mista, combinando elementos quantitativos e qualitativos. Perguntas fechadas, como idade, região de residência e experiência prévia com plataformas de programação, permitem quantificar dados e identificar tendências entre os participantes. As perguntas abertas, que solicitavam sugestões para aprimorar a atratividade e a eficácia da plataforma, fornecem dados qualitativos que ajudam a compreender de forma mais aprofundada as percepções, necessidades e expectativas do público-alvo.

Além disso, trata-se de uma pesquisa do tipo survey (levantamento de dados), realizada por meio de questionário estruturado. Essa abordagem permite coletar informações de um grupo específico de pessoas de forma rápida e eficiente, possibilitando identificar padrões e preferências que servirão como base para o desenvolvimento de soluções educacionais inovadoras voltadas ao ensino de programação infantil. O levantamento de dados caracteriza-se pela coleta direta de informações junto a um grupo de pessoas, geralmente por meio de questionários, com

o objetivo de descrever com precisão características de populações ou amostras específicas (Raupp & Beuren, 2006).

3.1.2 Análise de resultados da plataforma

O formulário destinado à análise da experiência dos usuários com a Plataforma Otto foi aplicado em novembro de 2025, por meio de um questionário estruturado no Google Forms. A pesquisa possui caráter quantitativo descritivo, permitindo examinar a percepção dos participantes quanto à usabilidade, ao design e ao potencial pedagógico da plataforma no ensino de lógica de programação para o público infantil.

De maneira geral, os resultados apontaram avaliações positivas sobre a experiência na plataforma, destacando seu potencial para despertar interesse, incentivar a continuidade do aprendizado e contribuir para o desenvolvimento do raciocínio lógico. Os participantes também reconheceram a relevância da ferramenta para o contexto educacional, especialmente por seu alinhamento com princípios de uma educação inclusiva e de qualidade.

Além disso, identificou-se uma boa aceitação quanto à proposta visual e metodológica baseada no uso de blocos de programação, considerada apropriada para o público-alvo. A análise do questionário ainda evidenciou que a Plataforma Otto apresenta potencial para engajar crianças e estimular o aprendizado de forma lúdica, sendo vista como uma solução promissora para o incentivo à tecnologia na educação básica.

3.2 Etapas do desenvolvimento da plataforma OTTO

3.2.1 Prototipagem

O protótipo da plataforma foi desenvolvido no Figma, com o objetivo de representar visualmente as telas e o fluxo de navegação da aplicação antes da fase de implementação. O projeto inclui uma landing page, que corresponde à página inicial de apresentação do sistema, utilizada para introduzir o usuário à plataforma e destacar suas principais funcionalidades e benefícios. Essa seção abrange telas como

“Inicial”, “Sobre o Projeto” e “Por que estudar com o Otto?”, funcionando como porta de entrada para a experiência do usuário.

Além disso, o protótipo contempla as demais interfaces da plataforma, como Cadastro, Login, Home, Fases, Progresso, Configurações e Meu Perfil, bem como telas complementares, entre elas Interface/Tema/Acessibilidade, Ajuda e Suporte, FAQ (Perguntas Frequentes), Jogo (ambiente de programação), Escolha de Fase Final, Fale Conosco, Agradecimento pelo Contato, Esqueceu a Senha?, Pré-jogo, Mapa do Jogo, Personalizar Avatar, Verificação de Código, Exclusão de Conta, Confirmação de Saída, etc. Em conjunto, essas telas possibilitam compreender o funcionamento e a navegação completa dentro da plataforma.

O processo de prototipagem iniciou-se com o mapeamento das funcionalidades e a definição da identidade visual, buscando uma estética lúdica, acessível e coerente com os objetivos educacionais do projeto. As telas foram elaboradas individualmente no Figma, observando princípios de usabilidade, consistência visual e acessibilidade.

Após a criação dos layouts, foram inseridas interações entre as páginas, simulando o fluxo real de navegação do usuário. Essa etapa possibilitou validar a estrutura e a experiência da interface, realizar ajustes visuais e funcionais e estabelecer uma base sólida para o desenvolvimento final da plataforma Otto.

3.2.2 Front-end

O front-end é a parte da aplicação responsável pela interface visual e pela interação direta com o usuário. De acordo com os estudos disponíveis nos repositórios da Universidade de Trás-os-Montes e Alto Douro (UTAD) e do Instituto Politécnico de Lisboa (IPL), o front-end corresponde à camada cliente do sistema, sendo aquela que “traduz as informações processadas no back-end em elementos gráficos e funcionais compreensíveis ao usuário final” (UTAD, 2021). Essa camada utiliza tecnologias como HTML, CSS e JavaScript para estruturar, estilizar e tornar as páginas interativas, proporcionando uma navegação intuitiva e responsiva. Ainda segundo os autores, “o front-end tem como principal objetivo garantir a melhor experiência de uso, aliando estética, desempenho e acessibilidade” (IPL, 2022). Assim, o front-end atua como a ponte entre o usuário e o sistema, transformando dados e funcionalidades técnicas em uma interface clara e funcional.

3.2.2.1 HyperText Markup Language

O HyperText Markup Language, popularmente conhecido como HTML, desempenhou um papel fundamental na estruturação do projeto, pois é a linguagem responsável por organizar e definir a disposição dos elementos em uma página da web. Ele funciona como a base sobre a qual todo o conteúdo é construído, determinando onde textos, imagens, vídeos e links serão exibidos. Segundo Aires (2024, p. 16), “HTML, abreviatura para HyperText Markup Language, traduzido Linguagem de Marcas de Hipertexto, não é considerada uma linguagem de programação, já que não pode criar funcionalidades dinâmicas. Ao invés disso, como o nome indica, é uma linguagem composta por diversas tags, estas definem elementos que dão forma às diferentes partes das páginas Web, como títulos, parágrafos, imagens, links e outros componentes de multimídia”.

No contexto do projeto, o HTML foi essencial para estabelecer a estrutura principal da interface, funcionando como um “esqueleto” que sustenta todo o site. É por meio dele que o navegador interpreta e exibe o conteúdo de forma organizada, garantindo que cada parte da página esteja em seu devido lugar. Como destaca o mesmo autor, “esta é a linguagem mais utilizada na construção de páginas na Web, compondo a maior parte das páginas da Internet e das aplicações online” (Aires, 2024, p. 16).

Assim, pode-se afirmar que o HTML foi a base estrutural do projeto, permitindo que o conteúdo fosse corretamente distribuído e visualmente coerente. A partir dessa fundação, outras tecnologias — como o CSS, responsável pela aparência visual, e o JavaScript, que adiciona interatividade — puderam ser integradas, completando o desenvolvimento e tornando o site mais dinâmico e funcional.

3.2.2.2 Cascading Style Sheets

O Cascading Style Sheets, conhecido como CSS, traduzido como Folhas de Estilo em Cascata, é uma linguagem utilizada para definir a aparência e a formatação das páginas da web. Segundo Aires (2024), “o CSS é a abreviatura para Cascading Style Sheet [...] e é utilizada para conceder estilo a elementos escritos em linguagens de marcas como é o exemplo do HTML” (Aires, 2024, p. 18).

O CSS permite modificar características visuais como cores, fontes, tamanhos e espaçamentos, além de possibilitar a criação de layouts e o ajuste de imagens. Conforme o autor explica, “o CSS separa o conteúdo da representação visual do Website” e “é possível alterar a cor do texto e do fundo, fonte e espaçamento entre parágrafos, para além de ser possível realizar variações de layouts e ajustar imagens” (Aires, 2024, p. 18).

Essa separação entre conteúdo e estilo torna o desenvolvimento web mais organizado e eficiente, facilitando a manutenção e atualização das páginas. Dessa forma, o CSS teve um papel essencial no projeto, sendo responsável por garantir uma identidade visual harmônica e um design coerente, sem interferir na estrutura definida pelo HTML.

3.2.2.3 Linguagem de Programação JavaScript

O JavaScript é uma linguagem de programação amplamente utilizada para tornar websites e aplicativos mais interativos e dinâmicos. Diferente de conteúdos estáticos, que apenas exibem informações, o JavaScript permite que as páginas respondam às ações do usuário, como cliques, preenchimento de formulários e movimentos do mouse. Segundo Biffi (2021), o JavaScript é fundamental para criar lógicas que tornam os sites mais funcionais e responsivos.

Além disso, o JavaScript facilita a implementação de estruturas de controle, como loops e condicionais, que são essenciais para a execução de tarefas de forma organizada e eficiente. Iepsen (2020) destaca que a linguagem auxilia os programadores na construção de algoritmos claros e objetivos, servindo também

como uma importante ferramenta para o aprendizado da lógica de programação.

Por sua simplicidade, versatilidade e ampla aplicação, o JavaScript se consolidou como uma das linguagens mais importantes no desenvolvimento web moderno, estando presente em sites, aplicativos e sistemas online que exigem interatividade e dinamismo.

3.2.3 Back-end

O back-end é a camada responsável pelo funcionamento interno de uma aplicação, atuando no servidor e gerenciando toda a lógica de negócios, a persistência de dados e a comunicação com o cliente por meio de APIs. Conforme apresentado no repositório do Instituto Politécnico de Lisboa, o back-end “é a parte da aplicação que não está visível para o utilizador mas é indispensável. Esta é responsável pela lógica, processamento de dados, comunicação com o servidor e dá apoio às ações do utilizador” (AIRES, 2024, p. xi).

Em outras palavras, o back-end é o motor invisível que faz o sistema funcionar, processando requisições, consultando e alterando bases de dados, e oferecendo serviços ao front-end de forma segura e eficiente. Embora não seja visível ao usuário, essa camada é essencial para garantir que operações como autenticação, armazenamento, controle de acesso e validação de regras de negócio ocorram corretamente. É a partir dela que o front-end obtém informações e executa ações, assegurando o equilíbrio e a coerência funcional de toda a aplicação.

3.2.3.1 Framework web Django

O Django é um framework web desenvolvido em Python que foi utilizado no projeto para realizar o controle geral do sistema, consultas ao banco de dados, comunicação entre interfaces, definição de rotas e autenticação de usuários. Ele fornece uma estrutura organizada que facilita o desenvolvimento de aplicações web complexas, permitindo que cada parte do sistema desempenhe sua função de forma

clara e eficiente. Segundo estudos publicados na ResearchGate (2021), o Django facilita o trabalho dos desenvolvedores, permitindo que eles criem aplicações seguras e escaláveis sem precisar escrever tudo do zero.

Em termos simples, o Django atua como o motor que mantém o site funcionando, processando informações, organizando dados e garantindo que as funcionalidades estejam disponíveis de maneira confiável. Sua arquitetura baseada no padrão Model-View-Template (MVT) separa a lógica do sistema da camada de apresentação, o que torna o código mais limpo e fácil de manter.

Além disso, o framework oferece recursos integrados de segurança, autenticação e gerenciamento de banco de dados, evitando repetições desnecessárias de código e otimizando o tempo de desenvolvimento. Por essas características, o Django é amplamente utilizado por pequenas e grandes organizações, sendo reconhecido pela sua robustez, escalabilidade e capacidade de sustentar aplicações web de alta

3.2.4 Banco de dados

3.2.4.1 Gerenciador de banco de dados (PostgreSQL)

O PostgreSQL é um Sistema Gerenciador de Banco de Dados (SGBD) relacional e orientado a objetos, desenvolvido como software livre e de código aberto. É amplamente utilizado por empresas e instituições acadêmicas por oferecer segurança, estabilidade e alta performance sem custos de licença. Segundo Souza, Amaral e Lizardo (2011), seus desenvolvedores são, em sua maioria, voluntários distribuídos pelo mundo, que colaboram continuamente pela internet. O PostgreSQL possui recursos avançados, como suporte a triggers, transações, chaves estrangeiras, controle de integridade referencial e as quatro propriedades ACID (atomicidade, consistência, isolamento e durabilidade), o que garante a confiabilidade dos dados (CARVALHO SANTOS, s.d.). Por sua robustez e flexibilidade, o PostgreSQL é considerado um dos SGBDs mais completos do mercado, sendo amplamente adotado em projetos acadêmicos e corporativos.

3.2.4.2 HeidiSQL

O HeidiSQL é um programa de computador utilizado para gerenciar bancos de dados de forma prática e visual, permitindo que o usuário acesse, visualize e edite informações em sistemas como MySQL, MariaDB, PostgreSQL e Microsoft SQL Server, sem a necessidade de escrever comandos manuais. Segundo Carvalho (2019), o software tem como objetivo ser fácil de aprender e utilizar, possibilitando a administração completa de bancos de dados mesmo por iniciantes. Baseado no modelo cliente-servidor, o HeidiSQL atua como o cliente, enviando comandos e exibindo resultados processados pelo servidor de banco de dados (TAKAI; ITALIANO; FERREIRA, 2005). Por ser software livre, pode ser baixado e utilizado gratuitamente, destacando-se por sua interface simples e intuitiva, o que o torna uma ferramenta amplamente usada em ambientes acadêmicos e empresariais para manipulação e gerenciamento eficiente de dados.

3.2.5 API e Integrações

3.2.5.1 Blockly

O Blockly é uma ferramenta desenvolvida pelo Google que permite criar programas de computador de forma visual e intuitiva, usando blocos que se encaixam uns nos outros em vez de escrever códigos complicados. Cada bloco representa uma ação ou comando do programa, como repetir tarefas, guardar informações ou fazer cálculos. Ao conectar os blocos na tela, o usuário constrói a lógica do programa, que pode ser executada ou transformada em código em linguagens como JavaScript, Python ou PHP. A ferramenta funciona diretamente no navegador, sem precisar de conexão constante com um servidor, e permite que desenvolvedores integrem o editor em seus próprios aplicativos. Dessa forma, mesmo pessoas sem experiência em programação podem aprender a construir algoritmos de maneira prática, segura e interativa, experimentando e visualizando como os programas funcionam.

3.2.5.2 Google Identity Services

A Google Identity Services é uma api da google que permite aos usuários se cadastrarem e fazerem login em nosso site usando a conta do google, sem precisar criar uma nova senha. Ela funciona como um atalho para facilitar a entrada do site sem uma necessidade real de criar uma outra conta. Ela elimina os formulários e cadastros e a simplifica um jeito sem senha. A verificação de identidade é feita diretamente pela empresa, protegendo contra bots e fraudes.

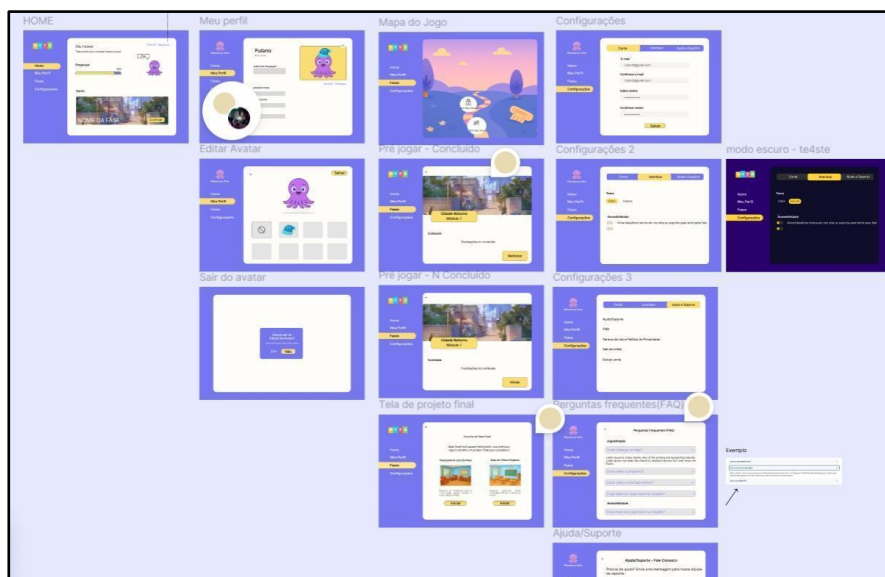
3.3 Ferramentas e tecnologias utilizadas

3.3.1 Figma

O Figma foi empregado como principal ferramenta de prototipagem e design de interface. Entre os benefícios do Figma, destacam-se a prototipagem interativa, que permite simular a navegação dentro da plataforma e realizar testes de usabilidade antes da implementação final, além da colaboração em tempo real, possibilitando que diferentes membros da equipe trabalhem simultaneamente no mesmo projeto. Dessa forma, o Figma possibilitou estruturar a interface da plataforma de maneira intuitiva, adequando a coerência visual em todas as páginas.

O Figma é uma ferramenta de design e prototipagem versátil que capacita designers e equipes a criar protótipos interativos, colaborar de forma contínua e dar vida às suas ideias criativas. Ele oferece uma ampla gama de recursos para projetar e prototipar interfaces para web e mobile, o que o torna valioso para profissionais e equipes de design que buscam otimizar seu fluxo de trabalho. (FIGMA, 2025).

Figura 3 - Protótipo do figma.



Fonte: Autoria própria, 2025.

3.3.2 ChatGPT

O ChatGPT foi utilizado como ferramenta de apoio tanto para o desenvolvimento de conteúdos textuais quanto para a criação das imagens utilizadas na plataforma. Por meio de prompts específicos, foi possível gerar imagens e elementos gráficos que integraram o design do projeto, permitindo rapidez e criatividade na produção de recursos visuais. Além disso, o ChatGPT contribuiu na revisão de textos e scripts, garantindo clareza e coerência na comunicação com os usuários. Essa ferramenta se mostrou essencial para unir conteúdo textual e visual de forma eficiente.

O ChatGPT é um modelo de linguagem natural criado pela OpenAI, empresa de desenvolvimento de pesquisa de Inteligência Artificial com sede em São Francisco, na Califórnia (EUA) e fundada em 2015. [...] Opera com base em redes neurais, treinadas com vastos conjuntos de dados. (RD STATION, 2025).

3.3.3 Canva

O Canva foi utilizado principalmente para a produção de elementos visuais que compõem a identidade da plataforma. Suas funcionalidades permitiram a criação de

logotipo, imagens ilustrativas e recursos gráficos, além do tratamento de imagens, como a remoção de fundos para integração em diferentes layouts.

Canva é uma plataforma de edição online gratuita que oferece diversas ferramentas de design para criar, editar e aprimorar modelos gráficos. Um dos objetivos do editor é oferecer um ambiente confiável para que os usuários desenvolvam a criatividade em seus projetos pessoais. (TECHTUDO, 2025).

3.3.4 Visual Studio Code

O Visual Studio Code (VS Code) é um editor de código-fonte desenvolvido pela Microsoft, utilizado para escrever, organizar e testar programas de computador. É uma ferramenta leve, mas poderosa, que ajuda programadores a desenvolver softwares de maneira mais eficiente. De acordo com Catarina (2023), o VS Code oferece suporte a múltiplas linguagens e recursos de depuração que facilitam o trabalho dos desenvolvedores.

Além disso, o VS Code permite personalização e instalação de extensões, que adicionam funcionalidades específicas, como integração com sistemas de controle de versão, suporte a novas linguagens de programação e ferramentas de acessibilidade. Lopes (2023) destaca que o editor possui recursos que tornam o ambiente de trabalho mais adaptável, inclusive para desenvolvedores com necessidades visuais especiais. Isso demonstra que o VS Code é uma ferramenta versátil, adequada tanto para iniciantes quanto para profissionais experientes no desenvolvimento de software.

3.3.5 GitHub

O GitHub é uma plataforma online criada para facilitar o trabalho em equipe no desenvolvimento de programas e projetos de software. Segundo Victor Maia de Sá (2023, p. 11), “O GitHub, fundado em 2008, é uma plataforma de hospedagem de códigos baseada em git, um sistema de controle de versionamento. Ele permite que desenvolvedores trabalhem em projetos de forma colaborativa, compartilhem seus códigos e controlem qualquer alteração feita no mesmo de forma clara e eficiente” (SÁ, 2023, p. 11). Em termos simples, ele funciona como um espaço onde várias pessoas podem editar um mesmo projeto sem perder informações ou causar

confusões, já que todas as versões ficam salvas e podem ser recuperadas a qualquer momento.

Além disso, o GitHub oferece ferramentas que vão além do armazenamento de código, tornando o trabalho mais organizado e interativo. O artigo destaca que “O GitHub também possui diversas ferramentas e recursos adicionais, como wikis para documentação de projetos, gerenciamento de projetos, rastreamento de problemas, integração com outras ferramentas de desenvolvimento” (Explorando o GitHub, 2023, p. 25). Assim, a plataforma se tornou um ambiente completo e acessível, que ajuda tanto iniciantes quanto profissionais a aprender, compartilhar conhecimento e colaborar de forma eficiente no mundo da tecnologia.

3.3.6 Git

O Git é uma ferramenta de controle de versão utilizada para registrar e gerenciar alterações em projetos, especialmente no desenvolvimento de software. Ele funciona como um histórico que armazena todas as modificações realizadas, permitindo retornar a versões anteriores caso necessário. Criado em 2005 por Linus Torvalds, o Git possibilita que várias pessoas colaborem em um mesmo projeto de forma simultânea e segura, mesmo em locais diferentes, sem risco de perda de dados ou substituição indevida de arquivos. Essa tecnologia tornou-se essencial em ambientes de desenvolvimento modernos por garantir organização, rastreabilidade e integridade das informações (LOELIGER; McCULLOUGH, 2012).

3.3.7 Microsoft Azure

Segundo a Microsoft (2025), o Azure é uma plataforma de computação em nuvem que disponibiliza serviços para hospedagem, armazenamento, processamento e execução de aplicações em datacenters da empresa, permitindo que organizações utilizem infraestrutura em nuvem com alta disponibilidade. Sua utilização elimina a necessidade de infraestrutura física própria, permitindo que sistemas sejam executados em datacenters da Microsoft com alta disponibilidade e segurança. Para a hospedagem do site, foi criada uma máquina virtual no Microsoft Azure, onde o

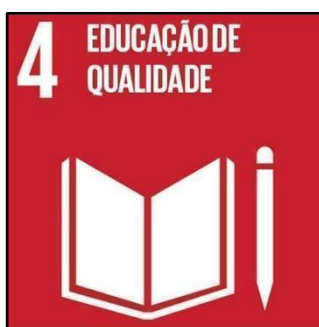
sistema foi configurado e colocado em funcionamento. Em seguida, foi liberada a porta de acesso necessária para tornar o site disponível ao público, atribuiu-se um endereço IP público à máquina virtual e, por fim, vinculou-se um domínio ao IP, possibilitando o acesso ao site por meio de um endereço personalizado.

4 DESENVOLVIMENTO DA PLATAFORMA OTTO

4.1 Objetivos da plataforma

A plataforma OTTO foi desenvolvida com o objetivo de incentivar o interesse e o conhecimento em tecnologia desde a infância, especificamente para jovens entre 8 e 12 anos. A iniciativa visa promover a educação tecnológica de forma inclusiva e equitativa, alinhando-se ao Objetivo de Desenvolvimento Sustentável (ODS) 04 da ONU, que busca assegurar a educação inclusiva, equitativa e de qualidade, e promover oportunidades de aprendizagem ao longo da vida para todos (PISCODELUZ.ORG, 2025). O recurso é gratuito, divertido e pode ser acessado em diversos dispositivos, como celulares, tablets ou computadores. A ferramenta contribui para que as crianças adquiram conhecimentos e habilidades essenciais, além de desmistificar a percepção de que a tecnologia é algo complexo e inacessível.

Figura 4 - ODS 04.



Fonte: Gtagenda2030, 2025.

O projeto se propõe a ensinar conceitos de programação de maneira lúdica, permitindo que as crianças não apenas aprendam a lógica por trás da codificação, mas também compreendam como a programação se aplica no mundo real. Por exemplo, por meio de atividades práticas, os usuários podem ter uma visão de como funcionam tecnologias do dia a dia, como a Internet das Coisas (IoT), já que têm contato com aplicações em seu cotidiano. Isso cria uma abertura para o aprendizado e a exploração de novas ideias.

“A Internet das Coisas, em poucas palavras, nada mais é que uma extensão da Internet atual, que proporciona aos objetos do dia-a-dia (quaisquer que sejam), mas com capacidade computacional e de comunicação, se conectarem à Internet. A conexão com a rede mundial de computadores viabilizará, primeiro, controlar remotamente os objetos e, segundo, permitir que os próprios objetos sejam acessados como provedores de serviços. Estas novas habilidades, dos objetos comuns, geram um grande número de oportunidades tanto no âmbito acadêmico quanto no industrial”. (SANTOS et al., 2016, p. 2).

A plataforma utiliza a Programação em Blocos, inspirada no Scratch, uma linguagem de programação visual desenvolvida pelo MIT Media Lab. Essa abordagem pedagógica está alinhada com a metodologia STEAM. Por meio de desafios e atividades interativas, o projeto busca desenvolver nos usuários habilidades como raciocínio lógico, pensamento crítico e criatividade, que são fundamentais para a resolução de problemas do cotidiano.

Além disso, o projeto também traz momentos de contato com a estrutura do Python, permitindo que os usuários façam a transição da lógica visual para a linguagem escrita. Dessa forma, as crianças percebem como os conceitos que praticam nos desafios em blocos podem ser representados em códigos reais, aproximando ainda mais o aprendizado do mundo da programação.

4.2 Jogos e Habilidades

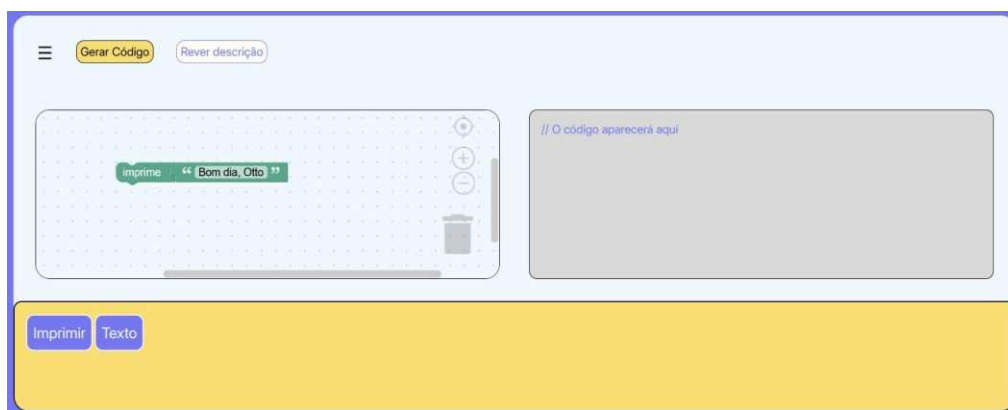
Na plataforma, cada nível representa uma parte da história do personagem Otto, criando uma narrativa envolvente que guia o aprendizado da programação. À medida que a criança avança de módulo em módulo, ela descobre novos desafios e aventuras de Otto, ao mesmo tempo em que pratica conceitos de lógica, programação em blocos e a sintaxe da linguagem de programação Python.

4.2.1 Bom Dia, Polvo!

No nível 1, os usuários iniciam o desenvolvimento de habilidades básicas de interação digital. Nesta fase, aprendem a combinar ações simples, como clicar em botões e associar eventos de impressão e texto, para gerar uma resposta visual e textual na tela. Ao utilizar o botão de imprimir junto ao de texto, o usuário faz com que

apareça a mensagem “Bom dia Otto!”, praticando a coordenação de comandos e a compreensão de sequência lógica de ações.

Figura 5 - Fase 1 da plataforma OTTO com o desafio resolvido.



Fonte: Autoria própria, 2025.

4.2.2 Hora do Café do Polvo

No Nível 2, os usuários aprofundam suas habilidades de pensamento lógico e tomada de decisão. Nesta fase, aprendem a utilizar variáveis e condições, elementos fundamentais da programação. O desafio consiste em criar a variável fruta, atribuir a ela o valor de texto “Maçã” e, em seguida, usar o comando “SE” para comparar o valor da variável. Ao verificar se fruta = Maçã, o usuário pratica a lógica condicional — imprimindo “Fruta certa!” quando a resposta estiver correta e “Fruta errada!” Caso contrário.

Figura 6 - Fase 2 da plataforma OTTO com o desafio resolvido.

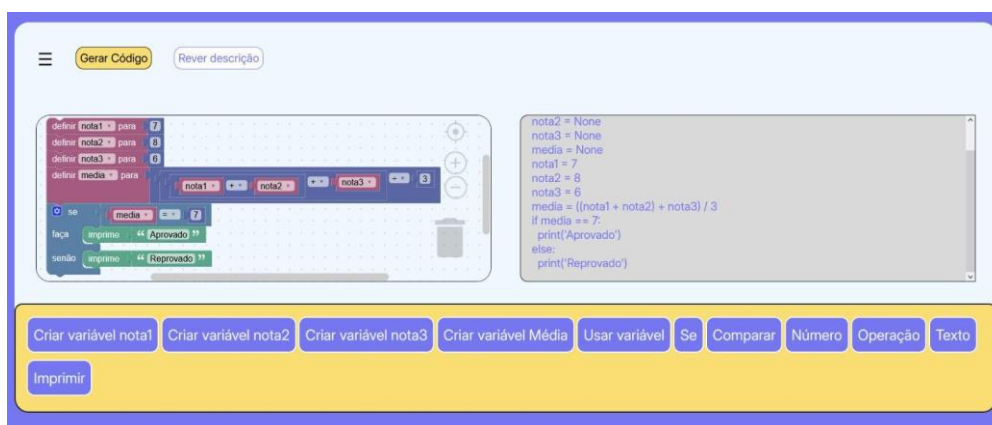


Fonte: Autoria própria, 2025.

4.2.3 Otto na Escola

No Nível 3, os usuários ampliam suas habilidades lógicas ao trabalhar com operações matemáticas e estruturas condicionais. Nesta fase, o desafio é ajudar o OTTO a calcular sua média de matemática, utilizando as notas 7, 8 e 6. O usuário deve criar as variáveis nota1, nota2 e nota3, atribuindo a elas os respectivos valores, e então calcular a média por meio da soma das três notas, utilizando o botão de operação de somar de forma aninhada (uma dentro da outra). Após obter o resultado, será necessário aplicar o comando “SE” para comparar se $\text{média} \geq 7$. Caso a condição seja verdadeira, o sistema deve imprimir “Aprovado!”, e se não for, “Reprovado!”.

Figura 7 - Fase 3 da plataforma OTTO com o desafio resolvido.



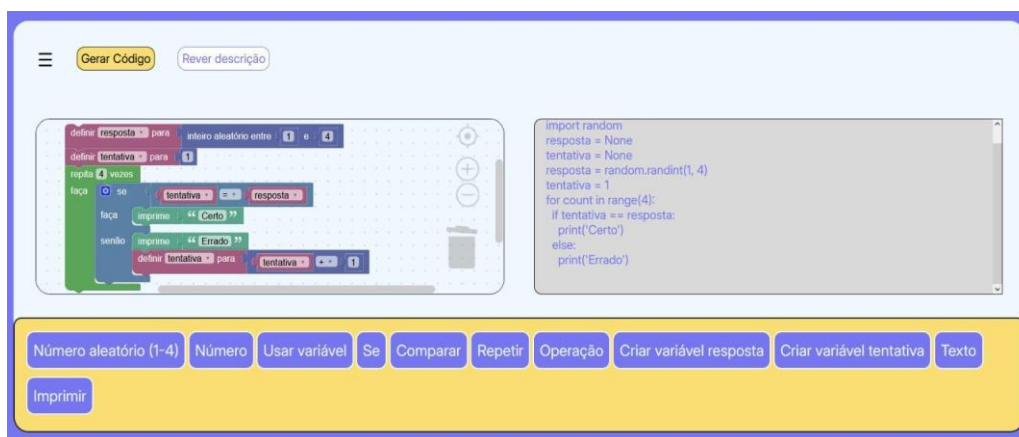
Fonte: Autoria própria, 2025.

4.2.4 Dança do Polvo Feliz

No Nível 4, os usuários desenvolvem o raciocínio lógico e a compreensão de repetições e testes condicionais. Nesta fase, o objetivo é ajudar o OTTO a acertar o número correto utilizando um processo de tentativa e erro. O usuário deve criar as variáveis resposta e tentativa, definindo a resposta como um número aleatório de 1 a 4 e iniciando a tentativa com o valor 1. Em seguida, é necessário usar o comando “repetir” configurado para 4 vezes, permitindo que o programa teste várias possibilidades. Dentro da repetição, o jogador adiciona uma verificação com o comando “SE”, comparando se $\text{tentativa} = \text{resposta}$. Se a condição for verdadeira,

deve-se imprimir “Certo!”; caso contrário, imprimir “Errado!” e atualizar o valor da variável tentativa somando +1, utilizando o botão de operação.

Figura 8 - Fase 4 da plataforma OTTO com o desafio resolvido.

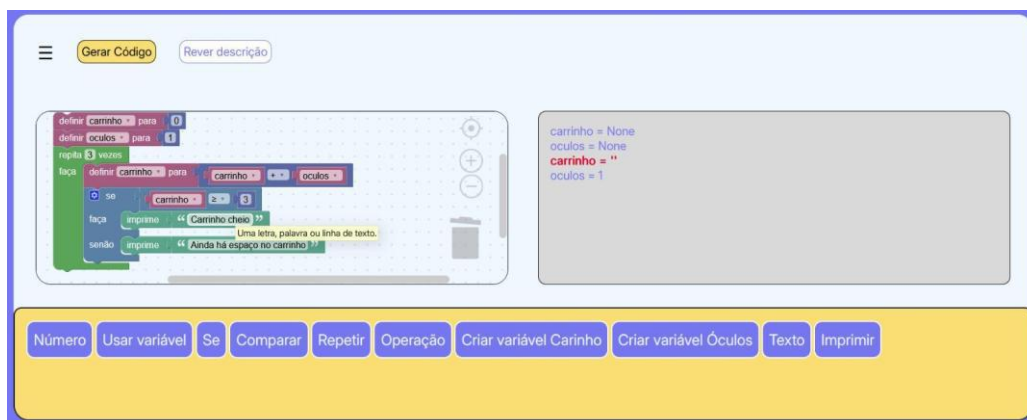


Fonte: Autoria própria, 2025.

4.2.5 Missão Mercado: Polvo Organizado

No Nível 5, os usuários consolidam o aprendizado sobre laços de repetição e acúmulo de valores em variáveis. Nesta fase, o desafio é ajudar o OTTO a encher seu carrinho de compras com três novos óculos. O jogador deve criar a variável carrinho, iniciando-a vazia, e a variável óculos com o valor 1. Em seguida, é preciso montar um laço de repetição configurado para 3 vezes, no qual o valor do carrinho será atualizado a cada ciclo, somando o valor atual com o de óculos. Após o laço, o usuário deve incluir uma verificação com o comando “SE”, comparando se carrinho \geq 3. Se a condição for verdadeira, o sistema deve imprimir “Carrinho cheio!”; caso contrário, exibir “Ainda há espaço no carrinho!”.

Figura 9 - Fase 5 da plataforma OTTO com o desafio resolvido.

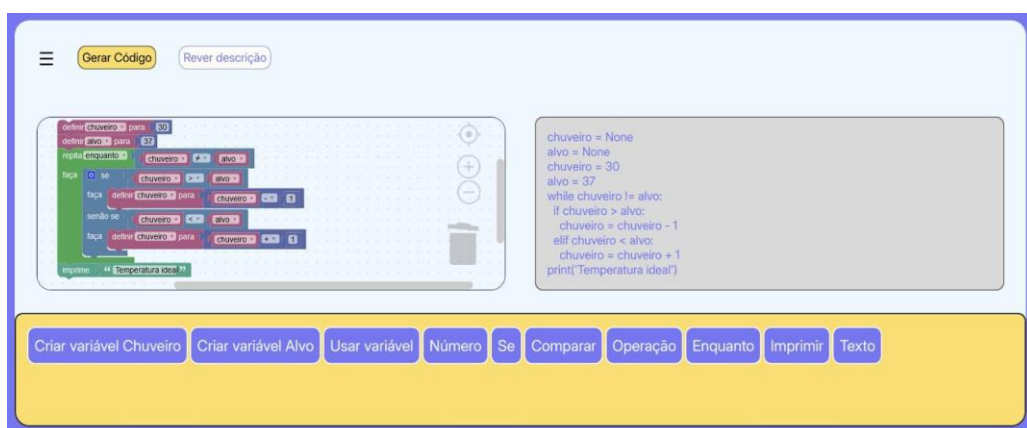


Fonte: Autoria própria, 2025.

4.2.6 Hora do Polvo Limpinho

No Nível 6, os usuários aprimoram seu entendimento sobre laços condicionais e ajustes automáticos de valores. Nesta fase, o objetivo é ajudar o OTTO a regular a temperatura do chuveiro até que ela fique ideal para o banho. O usuário deve iniciar com a variável chuveiro configurada em 30 graus, sabendo que o valor desejado é 37 graus. Enquanto a temperatura do chuveiro for diferente do valor ideal, o programa deverá ajustar automaticamente: Se chuveiro > 37, diminui 1 grau (chuveiro = chuveiro - 1); Se chuveiro < 37, aumenta 1 grau (chuveiro = chuveiro + 1). Quando o chuveiro atingir a temperatura ideal, o laço é encerrado e o sistema imprime a mensagem "Temperatura ideal!", indicando que OTTO pode finalmente tomar seu banho.

Figura 10 - Fase 6 da plataforma OTTO com o desafio resolvido.



Fonte: Autoria própria, 2025.

4.2.7 Restaurante Tentáculos

No Nível 7, os usuários aplicam conceitos de subtração, armazenamento de valores e verificação de resultados. Nesta fase, OTTO vai almoçar em um restaurante e precisa de ajuda para conferir se o troco está correto. O jogador deve criar as variáveis carteira, lanche, resposta e troco. A carteira começa com o valor 50 moedas, e o lanche custa 15 moedas. Em seguida, o troco deve ser calculado como o resultado da operação carteira - lanche. Depois, o usuário deve comparar se o valor do troco é igual à resposta esperada. Se a condição for verdadeira, o sistema imprime “O troco está certo!”; caso contrário, exibe “O troco está errado!”.

Figura 11 - Fase 7 da plataforma OTTO com o desafio resolvido.



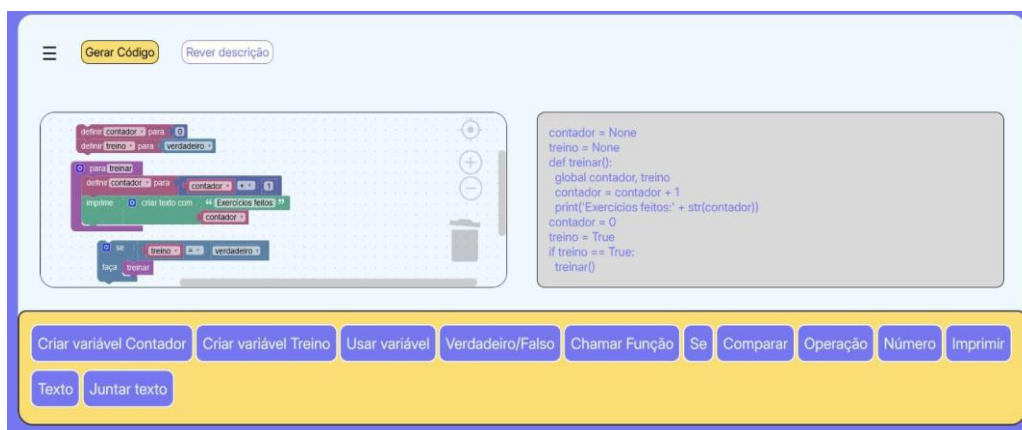
Fonte: Autoria própria, 2025.

4.2.8 Super Treino do Polvo

No Nível 8, os usuários aprendem a trabalhar com funções, variáveis booleanas e incrementos de contador, conceitos fundamentais na programação. Nesta fase, o objetivo é ajudar o OTTO a contar suas repetições de treino enquanto se exercita. O jogador deve iniciar criando a variável contador, com valor 0, e a variável treino, definida com verdadeira. Em seguida, é necessário verificar se treino = verdadeiro; caso seja, o programa deve chamar a função “treinar”. Dentro da função treinar, o valor do contador deve ser atualizado, somando +1 a cada execução (contador = contador + 1). Logo após, o sistema deve imprimir uma mensagem combinando o

texto “Exercícios feitos:” com o valor atual do contador, exibindo o total de repetições realizadas.

Figura 12 - Fase 8 da plataforma OTTO com o desafio resolvido.

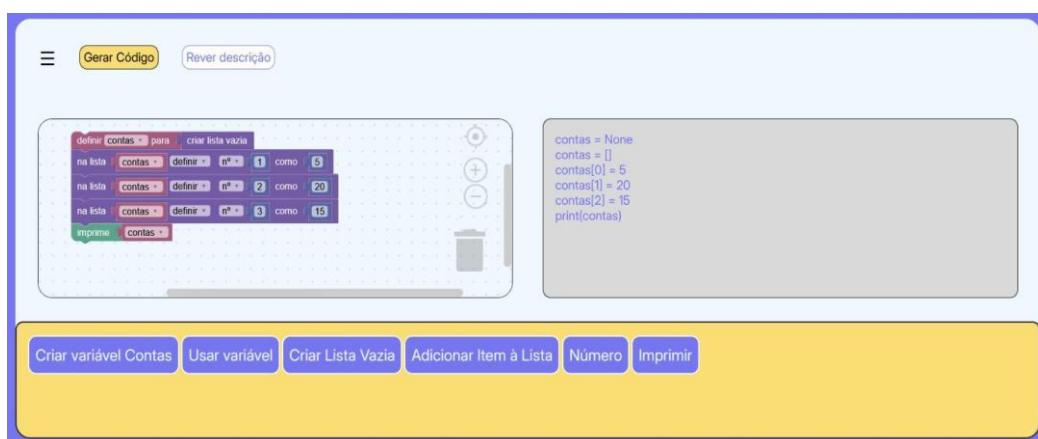


Fonte: Autoria própria, 2025.

4.2.9 Aventuras na Cidade do Otto!

No Módulo 9, os usuários desenvolvem habilidades no uso de listas e armazenamento de múltiplos valores. Nesta fase, o desafio é ajudar OTTO a registrar e analisar seus gastos durante um passeio. O jogador deve criar inicialmente uma lista vazia, armazenada na variável `contas`, e em seguida adicionar os gastos do OTTO: Sorvete: 5 moedas; Bola: 20 moedas; Lanche: 15 moedas. Após inserir todos os valores, o programa deve imprimir a lista da variável `contas`, permitindo visualizar e analisar todos os gastos realizados.

Figura 13 - Fase 9 da plataforma OTTO com o desafio resolvido.

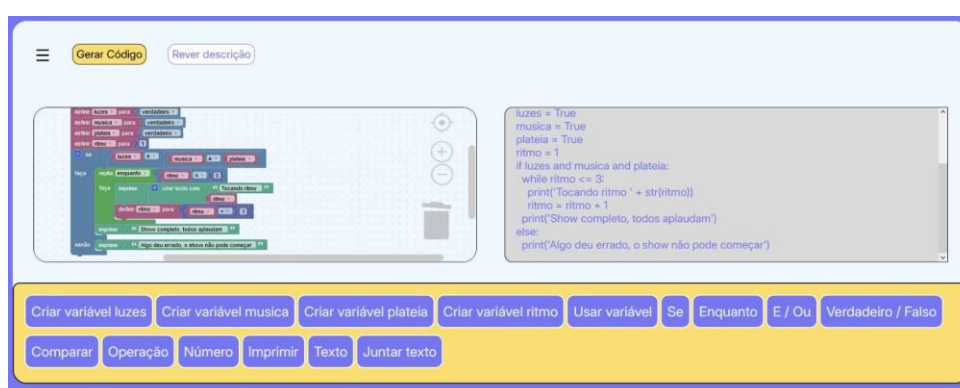


Fonte: Autoria própria, 2025.

4.2.10 Projeto final

Os usuários consolidam conceitos de condições múltiplas, laços de repetição e incremento de variáveis. Nesta fase, o objetivo é ajudar OTTO a realizar sua apresentação de show de forma correta. O jogador deve definir as variáveis luzes, música e plateia como verdadeiras. Quando a apresentação começar, o ritmo inicia em 1. Se luzes, música e plateia estiverem presentes, o programa deve executar um laço de repetição enquanto o ritmo ≤ 3 : Imprimir a mensagem “Tocando ritmo ” seguida do valor da variável ritmo; Incrementar a variável ritmo em +1. Quando o ritmo atingir o valor ideal, o sistema imprime “Show completo! Todos aplaudem!”. Caso alguma das condições iniciais não esteja correta, a mensagem “Algo deu errado! O show não pode começar.” será exibida, indicando que a apresentação não pôde ser realizada.

Figura 14 - Fase 10 da plataforma OTTO com o desafio resolvido.



Fonte: Autoria própria, 2025.

4.3 Funcionalidades principais, interface e usabilidade para crianças.

4.3.1 Funcionalidades Principais

A experiência de aprendizado é gamificada e se assemelha a um jogo. O site é gratuito e pode ser acessado em celulares, tablets ou computadores.

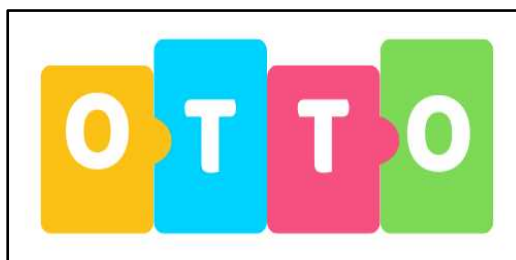
As funcionalidades incluem:

- Acesso à plataforma: Para começar, o usuário pode fazer login ou cadastro;
- Progressão por fases: A plataforma é dividida em fases, permitindo ao usuário progredir no curso e acompanhar seu progresso em porcentagem;
- Recursos de aprendizado: O site oferece dicas e curiosidades que auxiliam no conhecimento da área de tecnologia e programação;
- Dicas e Curiosidades: A plataforma também oferece dicas geradas aleatoriamente, trazendo curiosidades interessantes sobre o âmbito da tecnologia e da programação. São informações rápidas e úteis que ajudam o usuário a aprender de forma leve, descobrir novidades do mundo digital e aprimorar seus conhecimentos na área;
- Níveis: a cada módulo concluído, o nível do usuário sobe, seguindo a escala tecnológica: Bit, Byte, Kilobyte, Megabyte, Gigabyte, Terabyte, Petabyte, Exabyte, Zettabyte e Yottabyte. Sendo assim, o total de 10 níveis para a conclusão.

4.3.2 Identidade Visual

O logotipo foi desenvolvido com base em um formato que remete diretamente aos blocos de programação em blocos, criando uma referência com o intuito do projeto. Essa escolha de design facilita o reconhecimento do conceito de construção e aprendizado por meio da montagem de partes, além de transmitir os objetivos que a plataforma visa trazer: a forma lúdica e a interatividade. As cores amarela, azul, rosa e verde foram selecionadas especialmente para atingir o público infantil, despertando interesse, curiosidade e tornando a identidade mais amigável e divertida.

Figura 15 - Logotipo.

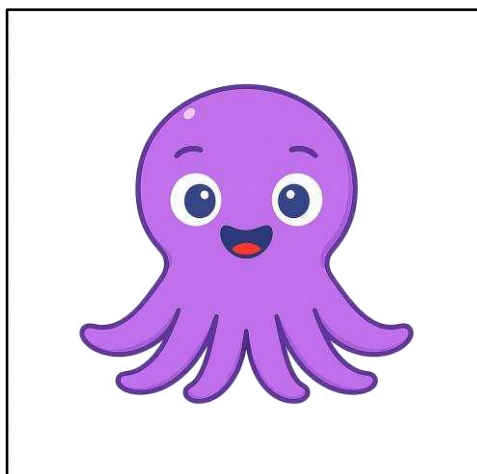


Fonte: Autoria própria, 2025.

O personagem Otto, nome inspirado na palavra do inglês “octopus”, é um polvo de tonalidade roxa que simboliza alegria e sabedoria, serve como base para a identidade visual deste projeto. Seu design é formado por linhas arredondadas, remetendo à figura do polvo. Essa escolha busca criar uma sensação de fluidez e conforto visual, tornando a identidade mais acessível.

A expressão e a sensação do projeto são um reflexo direto do mascote, o objetivo é evocar uma sensação de simpatia, felicidade e curiosidade, alinhada à expressão feliz e amigável do personagem. O caráter de desenho animado se conecta perfeitamente com a paleta de cores e as formas do design, adicionando um toque descontraído que torna o projeto visualmente convidativo e, ao mesmo tempo, acessível.

Figura 16 - Personagem Otto.



Fonte: ChatGPT, 2025.

4.3.3 Acessibilidade

Além disso, a ferramenta tem como objetivo fortalecer a inclusão digital. A acessibilidade é tratada como um elemento central, oferecendo opções como modo escuro, alto contraste, VLibras, leitor de voz e navegação por teclado. O modo escuro ajusta as cores da interface para tons mais escuros, reduzindo o cansaço visual e facilitando o uso em ambientes pouco iluminados. O alto contraste melhora a distinção entre elementos da tela, auxiliando pessoas com baixa visão. O VLibras traduz conteúdos para a Língua Brasileira de Sinais, ampliando o acesso para usuários surdos. Já o leitor de voz converte textos exibidos na tela em áudio, beneficiando pessoas com deficiência visual ou dificuldades de leitura. Por fim, a navegação por teclado permite que o usuário percorra toda a interface sem utilizar o mouse, garantindo autonomia para quem tem limitações motoras.

As preferências selecionadas pelo usuário são armazenadas pelo sistema, assegurando uma experiência personalizada sempre que ele retornar à plataforma. Assim, a ferramenta torna-se mais acessível, inclusiva e democrática, adaptando-se às necessidades de cada pessoa.

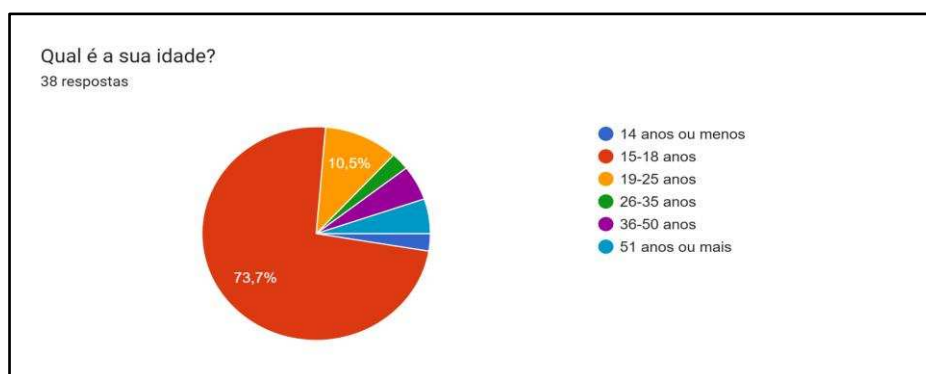
5 AVALIAÇÃO E RESULTADOS

5.1 Análise dos resultados obtidos: Pesquisa de Campo da plataforma

A pesquisa de campo foi realizada em junho de 2025, com 38 participantes, por meio de um questionário estruturado no Google Forms, com o objetivo de investigar percepções de adultos sobre plataformas de programação voltadas para crianças de 8 a 12 anos, incluindo características desejadas, impactos no aprendizado e sugestões de melhoria.

A faixa etária dos respondentes evidencia um público majoritariamente jovem: 73,7% têm entre 15 e 18 anos, 10,5% têm 14 anos ou menos, 14% estão entre 19 e 25 anos, e 5,3% têm 26 anos ou mais. Essa distribuição indica que a maior parte das respostas reflete a percepção de adolescentes e jovens adultos, potencialmente com experiência recente em aprendizado digital.

Figura 17 - Distribuição por faixa etária dos respondentes.

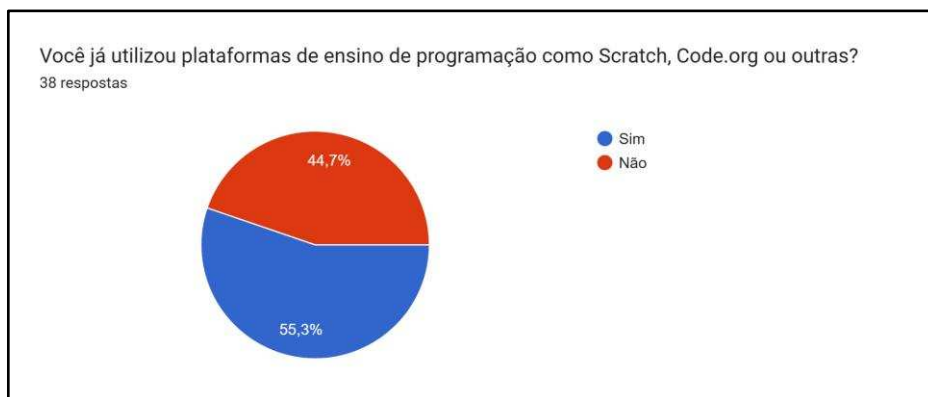


Fonte: Autoria própria, 2025.

Sobre a experiência prévia com plataformas de programação (como Scratch ou Code.org), 44,7% dos participantes afirmaram ter utilizado alguma ferramenta, enquanto 55,3% não possuíam experiência. Esse equilíbrio sugere que a maioria dos respondentes combina perspectivas de iniciantes e usuários mais familiarizados,

contribuindo para opiniões variadas sobre as necessidades de ensino de programação infantil.

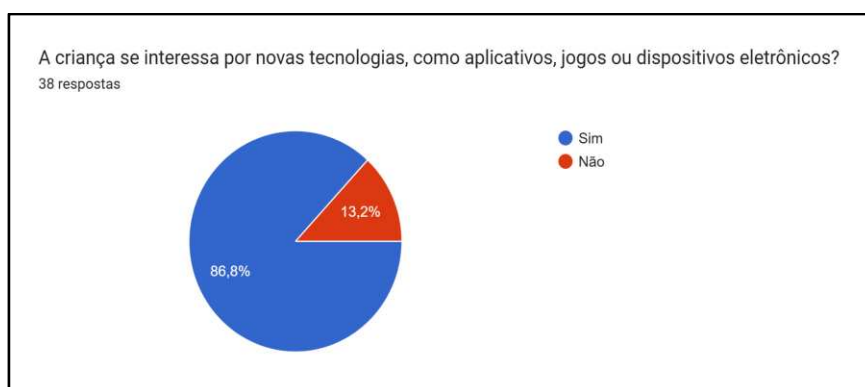
Figura 18 - Uso de plataformas de programação.



Fonte: Autoria própria, 2025.

No que diz respeito ao contato com crianças de 8 a 12 anos, 86,8% afirmaram ter algum contato, e dessa parcela, 86,8% observaram que as crianças demonstram interesse por tecnologias, incluindo aplicativos, jogos e dispositivos eletrônicos. Esses dados indicam que os participantes têm conhecimento direto do comportamento das crianças em relação à tecnologia, tornando suas respostas relevantes para o desenvolvimento da plataforma.

Figura 19 - Interesse da criança por novas tecnologias.



Fonte: Autoria própria, 2025.

A pesquisa revelou que 97,4% dos participantes acreditam que uma plataforma de programação interativa pode ajudar a criança a entender como a tecnologia funciona de maneira prática e divertida. Este consenso demonstra forte percepção sobre o valor de metodologias interativas no ensino de programação, reforçando a importância de interfaces e conteúdos que engajem e motivem o aprendizado.

Figura 20 - Percepção sobre o benefício de plataformas de programação interativa.



Fonte: Autoria própria, 2025.

Os respondentes indicaram de que forma acreditam que a plataforma poderia contribuir para o desenvolvimento das crianças:

- ✓ Ensinar conceitos básicos de programação de forma divertida: 73,7%
- ✓ Desenvolver habilidades de raciocínio lógico e resolução de problemas: 63,2%
- ✓ Estimular a criatividade ao criar e modificar projetos: 57,9%

É interessante notar que o ensino de conceitos básicos teve maior percentual que o desenvolvimento da criatividade ou do raciocínio lógico. Isso indica que os participantes enxergam a plataforma primariamente como ferramenta de introdução à programação, mas também reconhecem seu potencial para fortalecer habilidades cognitivas e criativas.

Quando questionados sobre quais características são mais importantes, os participantes priorizaram:

- ✓ Conteúdo interativo e visual (jogos, animações): 81,6%
- ✓ Interface fácil e amigável: 68,4%
- ✓ Feedback e recompensas rápidas (como medalhas ou pontos): 39,5%
- ✓ Escrita e leitura fácil: 2,6%

Os dados sugerem que interatividade e apelo visual são elementos essenciais para engajamento infantil, enquanto a simplicidade de leitura é considerada pouco relevante, possivelmente porque se presume que crianças dessa faixa etária já possuem habilidades básicas de leitura. O feedback rápido, embora menos citado, é ainda relevante para estimular motivação e reforço positivo.

5.2 Feedback de usuários

A fase de testes da plataforma OTTO permitiu coletar informações valiosas sobre a experiência dos usuários, os pontos fortes identificados e as oportunidades de melhoria. As respostas foram analisadas com base em observação direta, testes práticos e feedback espontâneo dos participantes.

Inicialmente, avaliou-se o nível geral de satisfação dos usuários durante a utilização da plataforma. Os resultados revelaram uma experiência geral altamente positiva. Quando perguntados sobre como avaliariam a experiência geral na plataforma, 50% dos usuários a classificaram como Excelente e os outros 50% como Bom. Este resultado indica um alto grau de satisfação com a interface e a abordagem gamificada.

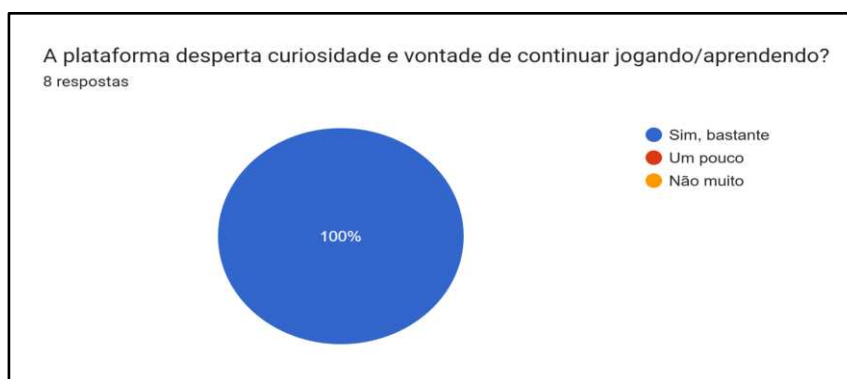
Figura 21 - Avaliação geral da experiência na Plataforma Otto.



Fonte: Autoria própria, 2025.

O potencial da plataforma para manter o interesse e a motivação foi unanimemente confirmado: 100% dos usuários responderam que a plataforma desperta curiosidade e vontade de continuar jogando/aprendendo. Este dado é crucial, pois valida a eficácia das técnicas de gamificação aplicadas para o público-alvo.

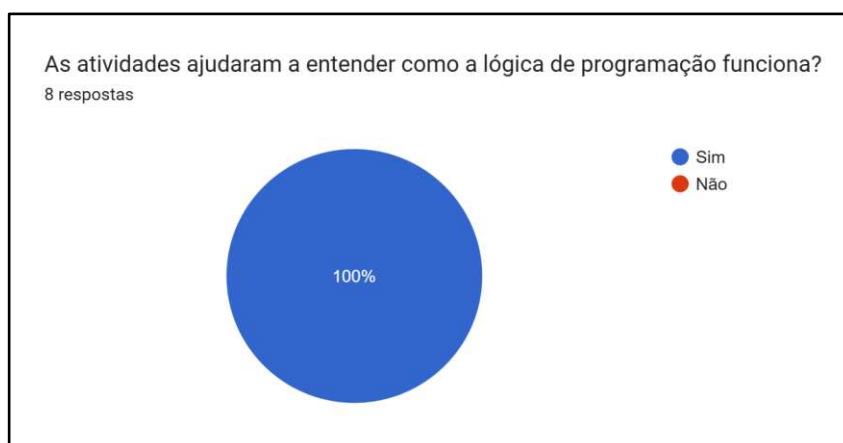
Figura 22 - Curiosidade e vontade de continuar jogando/aprendendo na Plataforma.



Fonte: Autoria própria, 2025.

A eficácia pedagógica da abordagem em blocos foi outro ponto de destaque, com 100% dos usuários afirmando que as atividades ajudaram a entender como a lógica de programação funciona. Isso demonstra que os objetivos de aprendizado da plataforma foram atingidos de forma clara para os usuários.

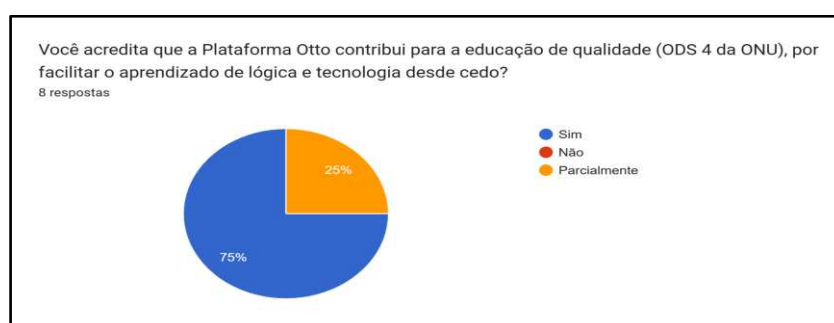
Figura 23 - Contribuição das atividades para o entendimento da lógica de programação.



Fonte: Autoria própria, 2025.

Em relação ao alinhamento com os objetivos sociais do projeto, 75% dos respondentes afirmaram que a Plataforma OTTO contribui para a educação de qualidade (ODS 4 da ONU), enquanto 25% responderam Parcialmente. Nenhum usuário indicou que a plataforma não contribui.

Figura 24 - Contribuição percebida da Plataforma Otto para a ODS 4 (Educação de Qualidade).



Fonte: Autoria própria, 2025.

A acessibilidade e inclusão, pilares do projeto, também obtiveram aprovação: 75% consideram a plataforma acessível para diferentes perfis de usuários e 25% a consideram Parcialmente acessível. Embora positivo, este resultado sugere que melhorias futuras podem ser implementadas para abranger perfis de necessidade mais específicos.

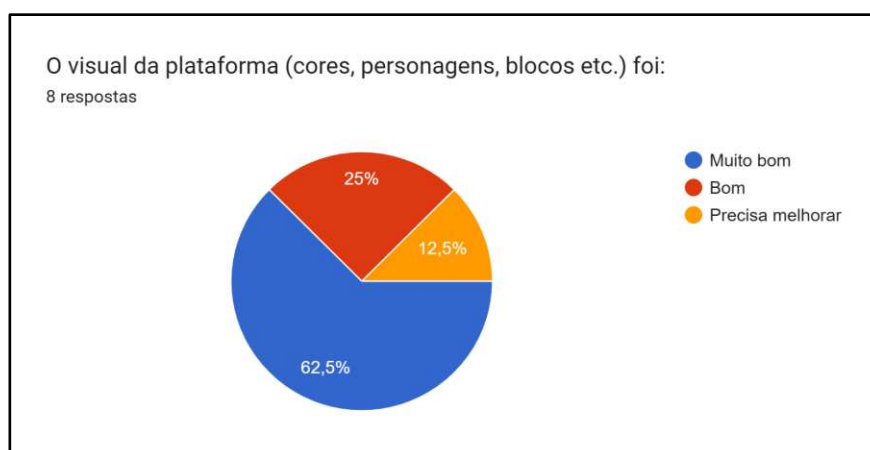
Figura 25 - Percepção sobre a acessibilidade da Plataforma para diferentes perfis de usuários.



Fonte: Autoria própria, 2025.

O aspecto visual da plataforma (cores, personagens, blocos etc.) foi avaliado majoritariamente de forma positiva: 62,5% como Bom e 25% como Muito Bom, somando 87,5% de aprovação. Apenas 12,5% indicaram que o visual Precisa melhorar.

Figura 26 - Avaliação do visual da Plataforma (cores, personagens, blocos, etc.).



Fonte: Autoria própria, 2025.

A escolha da programação em blocos foi validada com 100% de concordância de que a forma de ensino é fácil de entender para o público infantil, reforçando os dados da Figura 13 sobre a eficácia da metodologia.

Figura 27 - Entendimento da lógica de programação ensinada em blocos para crianças.



Fonte: Autoria própria, 2025.

Por fim, a intenção de uso futuro e recomendação atingiu 100% de resposta Sim, demonstrando alto potencial de adoção e satisfação por parte dos usuários testados.

Figura 28 - Intenção de reutilização e recomendação da Plataforma Otto.



Fonte: Autoria própria, 2025.

Em resumo, o feedback obtido na fase de testes confirma o sucesso da Plataforma OTTO. Os resultados são claros: a satisfação geral foi total (100% positiva)

e o alto índice de recomendação futura (100% "Sim") mostra que a plataforma agrada ao público. O ponto mais importante é a validação pedagógica: 100% dos usuários confirmaram que a metodologia de ensino em blocos é fácil de entender para crianças e que as atividades, de fato, ensinam a lógica de programação. Isso prova que o projeto alcançou seus principais objetivos, oferecendo uma ferramenta divertida e eficaz para a educação digital.

6 CONSIDERAÇÕES FINAIS

O presente trabalho apresentou o desenvolvimento da plataforma OTTO, um ambiente digital voltado ao ensino de lógica de programação para crianças de 8 a 12 anos, fundamentado em princípios de inclusão digital, interatividade e aprendizagem lúdica. A pesquisa evidenciou a importância de ampliar o acesso à educação tecnológica de forma equitativa, superando as barreiras sociais e econômicas que ainda limitam o contato de grande parte da população infantojuvenil com ferramentas de ensino inovadoras e acessíveis.

O percurso teórico e metodológico demonstrou que a inserção da programação no contexto educacional infantil vai além do simples ensino de códigos: ela estimula o raciocínio lógico, o pensamento crítico e a criatividade, elementos indispensáveis à formação integral do indivíduo. A integração entre construcionismo, gamificação e tecnologias educacionais mostrou-se um caminho eficaz para tornar o aprendizado mais dinâmico e participativo, aproximando o estudante da prática e da resolução de problemas de maneira contextualizada.

A plataforma foi idealizada com base nesses fundamentos, unindo design intuitivo, acessibilidade e interatividade para promover uma experiência de aprendizado significativa e inclusiva. Essa abordagem está em consonância com as diretrizes da Agenda 2030 da ONU, especialmente o Objetivo de Desenvolvimento Sustentável 4 (ODS 4), que propõe assegurar uma educação de qualidade, inclusiva e equitativa para todos.

A avaliação realizada por meio da pesquisa de campo e dos testes práticos com usuários confirmou que os objetivos propostos foram alcançados. Os participantes reconheceram a relevância de uma plataforma interativa para apoiar o aprendizado de programação infantil e destacaram o potencial da OTTO em despertar o interesse e a curiosidade das crianças, além de favorecer a compreensão da lógica de programação por meio da abordagem em blocos. Os testes indicaram ainda elevado nível de satisfação com a interface, com o caráter lúdico das atividades e com os recursos de acessibilidade incorporados, demonstrando que a experiência oferecida é inclusiva, atrativa e coerente com a proposta pedagógica do projeto.

Dessa forma, os resultados obtidos reforçam que a plataforma OTTO cumpre sua finalidade ao proporcionar um ambiente acessível e engajador para a introdução à programação, contribuindo para o desenvolvimento de habilidades cognitivas essenciais e aproximando as crianças do universo tecnológico de maneira prazerosa e significativa. Conclui-se, portanto, que a iniciativa constitui um recurso educacional relevante e alinhado às demandas da educação contemporânea, com potencial de ampliar o acesso ao conhecimento digital e promover uma aprendizagem mais democrática e transformadora.

O projeto tem como objetivo incentivar o interesse e o conhecimento em tecnologia desde a infância, promovendo uma educação tecnológica inclusiva e equitativa, em conformidade com o Objetivo de Desenvolvimento Sustentável (ODS) 04, que busca assegurar uma educação de qualidade e oportunidades de aprendizagem ao longo da vida para todos.

A plataforma OTTO é um recurso gratuito, acessível e divertido, podendo ser utilizada em celulares, tablets ou computadores. Seu propósito é contribuir para que as crianças adquiram conhecimentos e habilidades essenciais à atualidade, desmistificando a ideia de que a tecnologia é algo complexo e inacessível.

O projeto propõe o ensino de conceitos de programação de forma lúdica e interativa, permitindo que os usuários compreendam tanto a lógica por trás da codificação quanto sua aplicação no mundo real. Por meio de atividades práticas, as crianças podem explorar como funcionam tecnologias presentes em seu cotidiano — como a Internet das Coisas (IoT) —, estimulando a curiosidade e o pensamento investigativo.

A metodologia empregada baseia-se na Programação em Blocos, inspirada na linguagem Scratch, desenvolvida pelo MIT Media Lab. Essa abordagem está alinhada à filosofia STEAM (Ciência, Tecnologia, Engenharia, Artes e Matemática), promovendo o desenvolvimento de raciocínio lógico, pensamento crítico e criatividade, competências fundamentais para a resolução de problemas do dia a dia.

Além disso, a plataforma OTTO oferece momentos de contato com a linguagem Python, possibilitando aos usuários compreenderem a transição da lógica visual para a escrita de código real. Dessa forma, o aprendizado se torna mais completo e

significativo, aproximando as crianças do universo da programação profissional e ampliando suas possibilidades de futuro na área da tecnologia.

REFERÊNCIAS BIBLIOGRÁFICAS

AIRES, Afonso Filipe Garcia. **Desenvolvimento Front-End e Back-End de aplicações que solucionam problemas reais com a integração de Inteligência Artificial**. [S.l.: S.n.]. Disponível em:

<<https://repositorio.ipl.pt/entities/publication/33fc0254-3d57-425c-ad0f-ceceb9938988>>.

Análises situacionais e retrospectivas: inclusão digital. Disponível em:

<<https://repositorio.ipea.gov.br/entities/publication/4023d8c9-21ea-4d06-a35d-b9fc96e57ab8>>. Acesso em: 19 set. 2025.

BACH, Guilherme Berghann; MANGAN, Patrícia Kayser Vargas; RIVA, Aline Duarte.

Lógica de Programação e Pensamento Computacional Aplicados no Ensino Fundamental com o Uso do Scratch. In: ESCOLA REGIONAL DE COMPUTAÇÃO DO RIO GRANDE DO SUL (ERCOMP-RS), 2. , 2022, Canoas. Anais [...]. Porto Alegre: Sociedade Brasileira de Computação, 2022. p. 1-12. DOI: <https://doi.org/10.5753/ercomprs.2022.20401>.

BIFFI, Kelvin Baumhardt. **JavaScript Básico ao Avançado**. [S.l.: S.n.].

BRUNO P. SANTOS, LUCAS A. M. SILVA, CLAYSON S. F. S. CELES, JOÃO B. BORGES NETO, BRUNA S. PERES, MARCOS AUGUSTO M. VIEIRA, LUIZ FILIPE M. VIEIRA, OLGA N. GOUSSEVSKAIA E ANTONIO A. F. LOUREIRO. **Internet das Coisas: da Teoria à Prática**. [S.l.: S.n.].

CATARINA, Daniel Dantas. **STACK DEVELOPMENT EDUCATION - UMA EXTENSÃO PARA O EDITOR DE CÓDIGOS VS CODE QUE ARMAZENA DADOS DA BASE DE CONHECIMENTO STACK OVERFLOW**. [S.l.: S.n.].

CHANDIRAMANI, Ashish; DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING, AMITY SCHOOL OF ENGINEERING AND TECHNOLOGY, AMITY UNIVERSITY UTTAR PRADESH, LUCKNOW CAMPUS, INDIA. Management of Django Web Development in Python. **Journal of Management and Service Science (JMSS)**, v. 1, n. 2, p. 1–17, 2021.

DA SILVA, Aléxis José Rodrigues. **Estudo de Técnicas e de Tecnologias para o Desenvolvimento de Frontend de Aplicações Web.** [S.l.]: Universidade de Trás-os-Montes e Alto Douro, 2017.

DE ALMEIDA LIMA JÚNIOR, Janilson Elias. **IMPLEMENTAÇÃO DE UMA FERRAMENTA DE APOIO AO ENSINO DE SQL UTILIZANDO GOOGLE BLOCKLY.** [S.l.: S.n.].

DE ALMEIDA, Lília Bilati *et al.* O retrato da exclusão digital na sociedade brasileira. **Revista de Gestão da Tecnologia e Sistemas de Informação** , p. 55–67, fev. 2011.

DE CASTRO, Adriane; KOSCIANSKI, André (ORGS.). **O USO DA PROGRAMAÇÃO SCRATCH PARA O DESENVOLVIMENTO DE HABILIDADES EM CRIANÇAS DO ENSINO FUNDAMENTAL.** [S.l.]: Revista Tecnologias na Educação, 07-2017. v. 19

DE PAIVA, Fábio Augusto Procópio *et al.* **Introdução a Python com aplicações de sistemas operacionais.** Rua Dr. Nilo Bezerra Ramalho, 1692, Tirol. CEP: 59015-300, Natal-RN.: Editoraifn, 2021.

DE SÁ, Victor Maia. **EXPLORANDO O GITHUB: UM OBJETO DE APRENDIZAGEM ABRANGENTE SOBRE O SISTEMA DE CONTROLE DE VERSÕES.** [S.l.: S.n.].

DOURADO, Bruna. **ChatGPT: saiba o que é, como funciona e como utilizá-lo.** Disponível em:

<https://www.rdstation.com/blog/marketing/chatgpt/?utm_source=pmax&utm_medium=cpc&utm_campaign=BR-INB-RDSC-GOOGLE-HR_NOVOS_LEADS-BATE_PAPO-PMAX&utm_content=features-geral&utm_term=-&gad_source=1&gad_campaignid=22118179162&gbraid=0AAAAAC8pkCk7MRPecLOmDitWxstM9PWVX&gclid=CjwKCAjwk7DFBhBAEiwAeYbJsXhGnIEKj9tZtSVEs7UdbD5rDiUrjNmviMlftJcu5X_CG3honq21VhoCOR4QAvD_BwE>. Acesso em: 19 set. 2025.

Em 2025 Grande ABC terá mais idosos do que crianças. Disponível em: <<https://www.dgabc.com.br/Noticia/3803795/em-2025-grande-abc-tera-mais-idosos-do-que-criancas?>>. Acesso em: 19 set. 2025.

Ferramenta de prototipagem gratuita para sites e apps. Disponível em: <<https://www.figma.com/pt-br/prototipos/>>. Acesso em: 19 set. 2025.

GOUW, Ana Maria Santos; BIZZO, Nelio Marco Vincenzo. **A percepção dos jovens brasileiros sobre suas aulas de Ciências.** Educar em Revista, abr./jun. 2016.

JÚNIOR, José Walter Paulino; KELSEN, Francisco Oliveira. **Pensamento Computacional: Metodologia alternativa para o ensino de Lógica de Programação.** , 7-8 jun. 2024. Disponível em: <https://periodicos.ifsertao-pe.edu.br/ojs2/index.php/sepog/article/view/2306?utm_source>

JÚNIOR, Rogério Paulo Marcon; BONIATI, Bruno Batista. **LogicBlocks: Uma Ferramenta para o Ensino de Lógica de Programação.** [S.l.: S.n.].

LEPSEN, Edécio Fernando. **Lógica de Programação e Algoritmos com JavaScript.** Rua Luís Antônio dos Santos 110 02460-000 – São Paulo, SP – Brasil Tel.: +55 11 2959-6529: Novatec Editora Ltda, 2018.

LOPES, Paulo Ricardo da Silva. **VERIFICAÇÃO DO NÍVEL DE ACESSIBILIDADE DE FERRAMENTAS PARA DESENVOLVIMENTO DE SOFTWARE POR UM DESENVOLVEDOR BAIXA VISÃO COM A FERRAMENTA DE AMPLIAÇÃO DE TELA.** [S.l.: S.n.].

LÖSCH, S.; RAMBO, C. A.; FERREIRA, J. de L. **A pesquisa exploratória na abordagem qualitativa em educação.** Revista Ibero-Americana de Estudos em Educação, Araraquara, v. 18, n. 00, e023141, 2023. e-ISSN: 1982-5587. DOI: <https://doi.org/10.21723/riaee.v18i00.17958>

MAURI, Teresa. O que faz com que o aluno e a aluna aprendam os conteúdos escolares? In: COLL, Coll et al. **O construtivismo na sala de aula.** 6. ed. São Paulo: Ática, 2009.

MEDEIROS, Thácito Raboni Costa. **Implementação do sistema de controle de versões Git no ambiente de desenvolvimento de Sistemas Multiagente ChonIDE.** [S.l.]: CEFET/RJ - CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA

CELSO SUCKOW DA FONSECA, 2025.

O que é o Azure? Disponível em: <<https://azure.microsoft.com/pt-br/resources/cloud-computing-dictionary/what-is-azure>>. Acesso em: 10 nov. 2025.

Objetivo 4. Assegurar a educação inclusiva e equitativa e de qualidade, e promover oportunidades de aprendizagem ao longo da vida para todas e todos. Disponível em: <<https://gtagenda2030.org.br/ods/ods4/>>. Acesso em: 19 set. 2025.

PENSAMENTO COMPUTACIONAL PARA ENSINO DE ALGORITMO E LÓGICA DE PROGRAMAÇÃO EM CURSOS DE TECNOLOGIA EAD. SciELO Preprints, 2024. DOI: 10.1590/SciELOPreprints.10437. Disponível em: <https://preprints.scielo.org/index.php/scielo/preprint/view/10437>. Acesso em: 19 set. 2025.

PEREIRA, R. S. G.; ANASTÁCIO, P. R.; DAMIN, W.; ALVIM SOUSA, B. N. P. **O uso do Scratch no ensino de programação.** Ensino & Pesquisa, União da Vitória, v. 23, n. 1, p. 117–132, jan./abr. 2025. Disponível em: <https://periodicos.unespar.edu.br/ensinoepesquisa/article/download/9141/6838/38705>.

PINTO, Cibele Lemes; TAVARES, Helenice Maria (ORGS.). **O LÚDICO NA APRENDIZAGEM: APREENDER E APRENDER.** [S.l.]: Revista da Católica, 2010. v. 2

Pisco de Luz - Desenvolvimento Sustentável. Disponível em: <https://www.piscodeluz.org/desenvolvimento-sustentavel?gad_source=1&gad_campaignid=1631379533&gbraid=0AAAAAC5wUG2bTrxOYC0t048A3rWvqT9Gt&gclid=Cj0KCQjwuKnGBhD5ARIsAD19RsZXk9w_J67A7-TjLHzg2OQV542QXisJD6viJKY1X_FVxrKfPCrw_DkaAliYEALw_wcB>. Acesso em: 19 set. 2025.

PODER. 43% dos jovens têm pouco interesse em curso técnico, diz pesquisa. Disponível em: <<https://www.poder360.com.br/educacao/43-dos-jovens-tem-pouco-interesse-em-curso-tecnico-diz-pesquisa/?>>. Acesso em: 19 set. 2025.

Redes de ensino buscam caminhos para uso de tecnologia nas escolas.

Disponível em: <https://agenciabrasil.ebc.com.br/educacao/noticia/2023-08/redes-de-ensino-buscam-caminhos-para-o-uso-de-tecnologia-nas-escolas?utm_source>.

Acesso em: 19 set. 2025.

SANTANA, Maciel Trajano; DE ARAÚJO, Maykon Emanuel Moraes; MESQUITA, Kelânia Freire Martins (ORGS.). **Da intencionalidade à responsabilidade lúdica: o papel da gamificação na educação científica.** [S.l.]: SCIENTIA NATURALIS, 2024. v. 6

Scratch - About. Disponível em: <<https://scratch.mit.edu/about>>. Acesso em: 19 set. 2025.

VANZOLINI. **Gamificação na Educação: o aprendizado por meio dos jogos.**

Disponível em: <<https://vanzolini.org.br/blog/gamificacao-na-educacao/>>. Acesso em: 19 set. 2025.

VARGAS, Ketrin Diovana Alves Rodrigues; DA SILVA, João Pablo Silva; FINGER, Alice Fonseca. **Estratégias para o Ensino de Lógica Matemática com Pensamento Computacional: Uma Revisão Sistemática da Literatura.** *In*: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO (SBIE), 33. , 2022, Manaus. Anais [...]. Porto Alegre: Sociedade Brasileira de Computação, 2022 . p. 1391-1403. DOI: <https://doi.org/10.5753/sbie.2022.225243>.

Vista do Estratégias para o Ensino de Lógica Matemática com Pensamento Computacional: Uma Revisão Sistemática da Literatura. Disponível em:

<<https://sol.sbc.org.br/index.php/sbie/article/view/22510/22334>>. Acesso em: 19 set. 2025. Disponível em: <<https://itforum.com.br/noticias/setor-tic-147-mil-empregos-brasil-2025/>>. Acesso em: 19 set. 2025.

APÊNDICE A - PESQUISA SOBRE A PLATAFORMA OTTO

1) Qual é o seu nome?

2) Qual é a sua idade?

14 anos ou menos

15-18 anos

19-25 anos

26-35 anos

36-50 anos

51 anos ou mais

3) Você mora na região do ABC Paulista (Santo André, São Bernardo do Campo, São Caetano do Sul, etc.)?

Sim

Não

4) Você já utilizou plataformas de ensino de programação como Scratch, Code.org ou outras?

Sim

Não

5) Você tem contato/conhece crianças na faixa etária de 8 a 12 anos?

Sim

Não

6) A criança se interessa por novas tecnologias, como aplicativos, jogos ou dispositivos eletrônicos?

Sim

Não

7) Você acredita que usar uma plataforma de programação interativa pode ajudar a criança a entender como a tecnologia funciona de forma mais prática e divertida?

Sim

Não

8) Como você acredita que a plataforma pode ajudar no desenvolvimento de habilidades da criança, como raciocínio lógico e resolução de problemas?

Desenvolvendo habilidades de resolução de problemas através de atividades práticas

Estimulando a criatividade ao criar e modificar projetos

Ensinando conceitos básicos de programação de forma divertida

Outro:

9) Qual dessas características você considera essencial em uma plataforma de ensino de programação para crianças dessa faixa etária?

Interface fácil e amigável.

Conteúdo interativo e visual (jogos, animações)

Feedback e recompensas rápidas (como medalhas ou pontos)

Outro:

10) O que você sugere para tornar a plataforma mais atraente e eficaz para as crianças no ensino de programação?

APÊNDICE B – ANÁLISE DE RESULTADOS DA PLATAFORMA OTTO

1) Qual seu nome?

2) Como você avaliaria sua experiência geral na Plataforma Otto?

Excelente

Bom

Regular

Ruim

3) A plataforma desperta curiosidade e vontade de continuar jogando/aprendendo?

Sim, bastante

Um pouco

Não muito

4) As atividades ajudaram a entender como a lógica de programação funciona?

Sim

Não

5) Você acredita que a Plataforma Otto contribui para a educação de qualidade (ODS 4 da ONU), por facilitar o aprendizado de lógica e tecnologia desde cedo?

Sim

Não

Parcialmente

6) Você considera que a plataforma é acessível para diferentes perfis de usuários (crianças com diferentes ritmos de aprendizado, necessidades especiais, etc.)?

Sim

Não

Parcialmente

7) O visual da plataforma (cores, personagens, blocos etc.) foi:

Muito bom

Bom

Precisa melhorar

8) Você acha que as crianças de 8 a 12 anos gostariam de usar a Plataforma Otto?

Sim

Não

Talvez

9) A forma como a lógica de programação é ensinada (em blocos) é fácil de entender para o público infantil?

Sim

Não

10) Você usaria novamente a Plataforma Otto ou recomendaria para outras pessoas?

Sim

Não

Talvez