OTHALA - Omega (Ω)

**VISÃO DO PRODUTO E DO PROJETO**

Versão 1.1

Tabela - Integrantes do Grupo:

| Mat. | Nome | Função(responsabilidade) |
| --- | --- | --- |
| 231011892 | ARTHUR SISMENE CARVALHO | DESENVOLVEDOR FRONTEND |
| 221022480 | CARLOS HENRIQUE DE PAIVA MUNIS | DESENVOLVEDOR FRONTEND |
| 231030691 | GUILHERME FERREIRA BRANDAO | DESENVOLVEDOR BACKEND |
| 232002996 | HEYTTOR AUGUSTO DE ASSIS SILVA | DESENVOLVEDOR BACKEND |
| 202045348 | INGRID ALVES ROCHA | DONO DO PRODUTO |
| 232003661 | JOAO PEDRO ARAUJO DE FREITAS LYRA | LÍDER E DONO DO PRODUTO |
| 222022135 | LETICIA DE CARVALHO DOS SANTOS | ANALISTA DE QUALIDADE |
| 211045178 | LUIS GUILHERME BORGES MONTEIRO | ANALISTA DE QUALIDADE |
| 221007608 | NAYRA SILVA NERY | ANALISTA DE TESTES |
| 211031440 | PEDRO GOMES OLIVEIRA | ANALISTA DE TESTES |
| 232024026 | RIVADALVIO JOAQUIM DA SILVA FILHO | CLIENTE |
| 232014271 | YASMIN SOUSA ABDON | CLIENTE |

**Histórico de Revisões**

| **Data** | **Versão** | **Descrição** | **Autor** |
| --- | --- | --- | --- |
| **14/04/2025** | 0.1 | Informações iniciais a respeito do projeto e da equipe | Luis Guilherme Borges |
| **14/04/2025** | 0.2 | Complementando os parâmetros iniciais(1.1) | João Pedro Araújo de Freitas Lyra |
| **15/04/2025** | 0.3 | Declaração de posição do Produto (1.2) | Heyttor Augusto de Assis Silva, João Pedro Araújo de Freitas Lyra |
| **15/04/2025** | 0.3 | Edição da Visão Geral do Produto (1.1) e Bibliografia, diagrama da visão geral do produto | Yasmin Sousa Abdon, Pedro Gomes Oliveira |
| **15/04/2025** | 0.3 | Montagem do diagrama de ciclo de vida (2.1) | Guilherme Ferreira Brandão, João Pedro Araújo de Freitas Lyra e Rivadalvio Joaquim da Silva Filho. |
| **15/04/2025** | 0.3 | Objetivo do Produto (1.3) | João Pedro Araújo de Freitas Lyra, Yasmin Sousa Abdon |
| **15/04/2025** | 0.3 | Nomeação do projeto | Votação com todos integrantes. |
| **15/04/2025** | 0.3 | Tecnologias a serem utilizadas (1.4) | Vitória Aquere Matos,  Leticia Carvalho dos Santos, João Pedro Araújo de Freitas Lyra, Guilherme Ferreira Brandão |
| 14/05/2025 | 1.0 | [Organização do Projeto](#_p8v5oe3iowlu)(2.2) | Leticia Carvalho dos Santos |
| 14/05/2025 | 1.0 | Planejamento das fases(2.3) | João Pedro Araújo de Freitas Lyra |
| 14/05/2025 | 1.0 | Matriz de comunicação(2.4) | Heyttor Augusto de Assis Silva |
| 14/05/2025 | 1.0 | Gerenciamento de riscos(2.5) | Pedro Gomes Oliveira, Nayra Silva Nery |
| 14/05/2025 | 1.0 | Critérios de planejamento(2.6) | Nayra Silva Nery, Heyttor Augusto de Assis Silva |
| 14/05/2025 | 1.0 | Processo de desenvolvimento de software corpo(3) | Carlos Henrique de Paiva Munis |
| 14/05/2025 | 1.0 | Backlog do Produto(4.1) | Guilherme Ferreira Brandão, João Pedro Araújo de Freitas Lyra |
| 14/05/2025 | 1.0 | Perfis (4.2) | Guilherme Ferreira Brandão, Carlos Henrique de Paiva Munis |
| 14/05/2025 | 1.0 | Cenários (4.3) | Guilherme Ferreira Brandão |
| 14/05/2025 | 1.0 | Tabela backlog(4.4) | Guilherme Ferreira Brandão, João Pedro Araújo de Freitas Lyra |
| 16/05/2025 | 1.1 | Diagrama de processos(3) | Ingrid Alves Rocha |
| 16/05/2025 | 1.1 | GQM de medições (5.1) | Arthur Sismene Carvalho |
| 16/05/2025 | 1.1 | Estratégia de Testes(6.1) | Nayara Silva Nery |
| 16/05/2025 | 1.1 | Roteiro de Testes (6.2) | Luis Guilherme Borges e Yasmin Sousa Abdon |

**Sumário**

[***1***](#_wk8rfw8xlj02) ***VISÃO GERAL DO PRODUTO 3***

[**1.1**](#_pxir0kn6662a) **Problema 3**

[**1.2**](#_oupuxk9y22kl) **Declaração de Posição do Produto 3**

[**1.3**](#_veltoy21xja0) **Objetivos do Produto 4**

[**1.4**](#_59vap6n67wb2) **Tecnologias a Serem Utilizadas 4**

[***2***](#_7g3om19l8x4j) ***VISÃO GERAL DO PROJETO 4***

[**2.1**](#_ml6m6bw7cbro) **Ciclo de vida do projeto de desenvolvimento de software 4**

[**2.2**](#_p8v5oe3iowlu) **Organização do Projeto 4**

[**2.3**](#_ibhwwbyx5a8r) **Planejamento das Fases e/ou Iterações do Projeto 4**

[**2.4**](#_euwpze3y9f4d) **Matriz de Comunicação 5**

[**2.5**](#_kq1su2ubvm99) **Gerenciamento de Riscos 5**

[**2.6**](#_fyx9ov1ohmor) **Critérios de Replanejamento 5**

[***3***](#_x2e43tettim6) ***PROCESSO DE DESENVOLVIMENTO DE SOFTWARE 6***

[***4***](#_m61ebvvroiq2) ***DECLARAÇÃO DE ESCOPO DO PROJETO 6***

[**4.1**](#_iox6abcwx5oa) **Backlog do produto 6**

[**4.2**](#_sz9h0m4aqzqw) **Perfis 6**

[**4.3**](#_iqyvwjuq610l) **Cenários 6**

[**4.4**](#_h41qp42jyehu) **Tabela de Backlog do produto 7**

[***5***](#_f6tchedm5jka) ***MÉTRICAS E MEDIÇÕES 8***

[**5.1**](#_y82se01k5iou) **GQM de medições 8**

[***6***](#_lewv02h2pl22) ***TESTES DE SOFTWARE 8***

[**6.1**](#_887n1x1eqzt8) **Estratégia de testes contendo: 8**

[**6.2**](#_qun0r5vqqo6h) **Roteiro de teste: 8**

[***7***](#_p624a0j9bohj) ***REFERÊNCIAS BIBLIOGRÁFICAS 8***

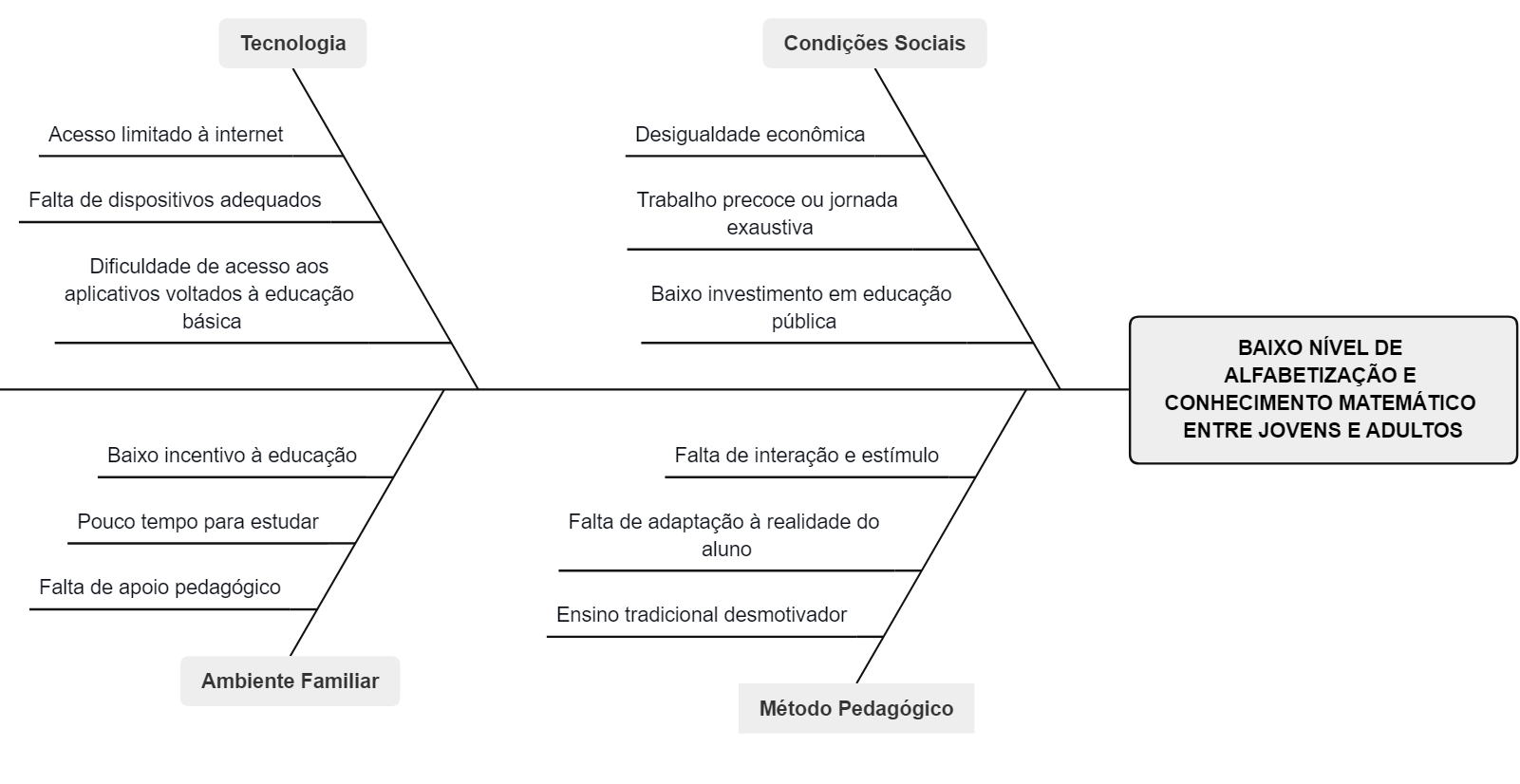
**VISÃO DO PRODUTO E PROJETO**

# VISÃO GERAL DO PRODUTO

## Problema

O Grupo **OTHALA** ficou responsável por desenvolver uma aplicação que estivesse relacionada com o ODS (Objetivo de Desenvolvimento Sustentável) 4, garantir educação inclusiva, equitativa e de qualidade, promovendo oportunidades de aprendizado para jovens e adultos da faixa entre 15 e 21 anos.

* **Contexto**: Ainda existe um índice preocupante de jovens e adultos que não possuem educação básica, carecendo de alfabetização e domínio das operações de matemática básica, de acordo com o censo de 2022 do IBGE, cerca de 7% da população, equivalente a 11,4 milhões de brasileiros, não são alfabetizados (IBGE, 2022), além de também no âmbito das ciências exatas, Pisa (Programa Internacional de Avaliação de Alunos) apontou que a matemática no Brasil é considerada “o maior gargalo brasileiro” no campo do ensino (Agência Brasil, 2023). Dentro deste contexto, além de mostrar um sério problema no sistema da educação brasileira, é perceptível a exclusão de milhões de cidadãos de direitos básicos, como o acesso ao conhecimento, à cidadania e ao mercado de trabalho.
* **Problema**: Dado essa existência da falta de educação básica que acarreta em uma série de problemas sociais em conjunto com a não participação de uma parcela do mercado consumidor brasileiro. Propõe-se como parte de uma solução mais ampla, pois apenas um produto de software não seria o suficiente para dar uma solução total a este problema complexo, uma aplicação em celular, pois este é o tipo de tecnologia mais comum de se ter, que funcionará dentro de um sistema de divisão em níveis com atividades que interagem com o usuário, progresso gradual e educação positiva buscando seguir a visão do filósofo Paulo Freire sobre o aprender humano, para criar uma aplicação interativa com o usuário que busca ensinar como ler desde de a identificação de fonemas, reconhecimento de símbolos e quantidades, até as quatro operações básicas da matemática até às centenas e ler frases simples.

****

## Declaração de Posição do Produto

| Para: | Pessoas de faixa etária de 15 a 21 anos que sejam semi-analfabetas nível básico, que tenham dificuldade com operações matemáticas, dificuldade em escrita correta, interpretação com um nível de leitura básica. |
| --- | --- |
| Necessidade: | Por mais que tenha sido reduzida com o passar dos anos, de acordo com o IBGE, ainda existem cerca de 9,3 milhões de brasileiros com algum nível de analfabetismo, então vê-se a necessidade de auxiliar nessa luta com o projeto desenvolvido. |
| O (nome do produto): | O produto é um app mobile com o nome omega (Ω) |
| Que: | Acredita-se que o projeto, se bem aplicado, contribuiria significativamente no combate ao analfabetismo e ao anumerismo. |
| Ao contrário: | Acredita-se que, caso nosso produto não exista, a taxa de analfabetismo continuará a mesma, e nenhuma mudança na situação acontecerá. |
| Nosso produto: | Nosso produto se propõe a ser uma maneira simplificada, rápida, acessível a todos e divertida no aprendizado. |

## Objetivos do Produto

Objetivo Principal: Auxiliar no combate ao analfabetismo e anumerismo.

Objetivo secundário **:** Motivar jovens e adultos a aprenderem a ler, escrever e dominar matemática básica de forma acessível, prática e lúdica, com a finalidade de incentivar o retorno aos estudos.

## Tecnologias a Serem Utilizadas

*Frontend*

* *React Native*
* *Typescript*

*Backend*

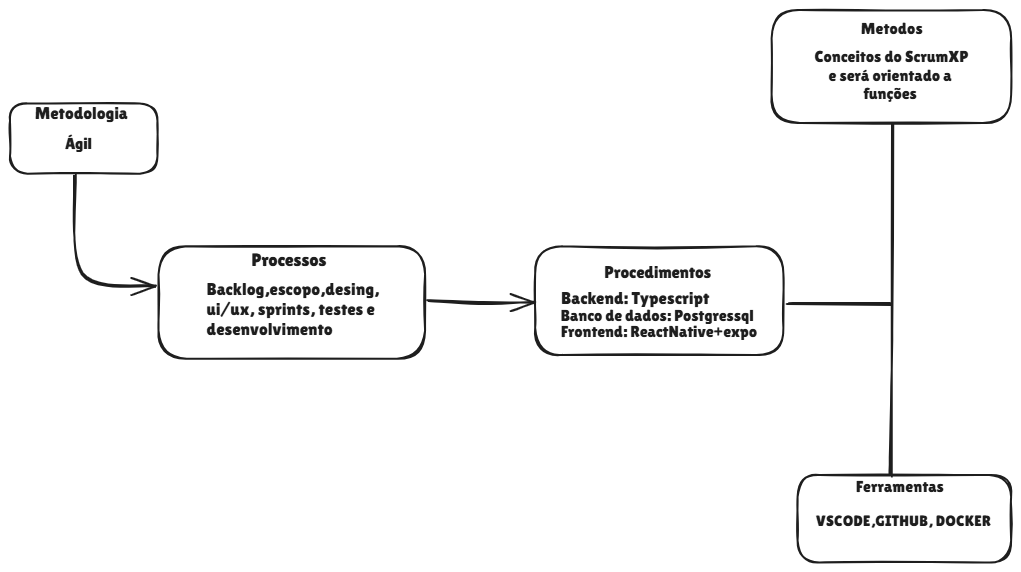
* *Typescript*
* *SQL*

*Ambiente de desenvolvimento*

* *Github*
* *VsCode*
* *Docker*

# VISÃO GERAL DO PROJETO

## Ciclo de vida do projeto de desenvolvimento de software



## Organização do Projeto

| Papel | Atribuições | Responsável | Participantes |
| --- | --- | --- | --- |
| Desenvolvedor Frontend | Implementação da Interface do Usuário (UI)  Criação de Componentes Interativos  Responsividade e Acessibilidade  Integração com APIs  Otimização de Performance  Segurança no Frontend | Arthur Sismene Carvalho | Arthur Sismene Carvalho, Carlos Henrique de Paiva Munis |
| Desenvolvedor Backend | Design e Implementação de APIs  Gerenciamento de Banco de Dados  Autenticação e Autorização  Integração com Sistemas Externos  Performance e Escalabilidade do Servidor  Tratamento de Erros e Logs  Segurança no Backend  Manutenção e Deploy da Aplicação | Guilherme Ferreira Brandão | Guilherme Ferreira Brandão, Heyttor Augusto de Assis Silva |
| Dono do Produto | Atualizar o escopo do produto, organizar o escopo das sprints, validar as entregas | João Pedro Araújo de Freitas Lyra | Ingrid Alves Costa, João Pedro Araújo de Freitas Lira |
| Analista de Qualidade | Garantir a qualidade do produto, garantir o cumprimento do conceito de pronto, realizar inspeções de código | Letícia de Carvalho dos Santos. | Letícia de Carvalho dos Santos, Luis Guilherme Borges Monteiro |
| Analista de Testes | Análise de Requisitos  Planejamento de Testes  Criação de Casos de Teste  Execução de Testes Manuais  Automação de Testes  Identificação e Reporte de Bugs  Reexecução de Testes Após Correções  Validação de Funcionalidades | Nayra Silva Nery | Nayra Silva Nery, Pedro Gomes Oliveira |
| Cliente | Definição de Requisitos  Priorização de Funcionalidades  Validação dos Resultados  Disponibilidade para Feedback  Participação em Reuniões (como refinamento ou revisão)  Teste e Homologação do Sistema  Aceite Final do Projeto | Rivaldo Joaquim da Silva Filho | Rivaldo Joaquim da Silva Filho, Yasmin Sousa Abdon |

## Planejamento das Fases e/ou Iterações do Projeto

| ***Sprint*** | ***Produto (Entrega)*** | ***Data Início*** | ***Data Fim*** | ***Entregável(eis)*** | ***Responsáveis*** | ***% conclusão*** |
| --- | --- | --- | --- | --- | --- | --- |
| *Sprint 1* | Definiu-se que se trata de um app mobile, de combate ao analfabetismo e anumerismo. | 14/04/2025 | 14/05/2025 | **Documentação** | João Pedro Araújo de Freitas Lyra | Divisão de tarefas início efetivo do projeto |
| Sprint 2 | MVP e Planejamento do Projeto | 14/05/2025 |  |  |  |  |
| *Sprint 3* | Reunião com uma parte dos integrantes para definir algumas metodologias para agilizar o processo de desenvolvimento. Também foi feito o primeiro commit de pastas que serão utilizadas para organização do trabalho. | 21/05 |  |  |  |  |
| *Sprint 4* |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  | *...* | *...* |  |  |  |

## Matriz de Comunicação

| **Descrição** | **Área/ Envolvidos** | **Periodicidade** | **Produtos Gerados** |
| --- | --- | --- | --- |
| * Acompanhamento das Atividades em Andamento * Acompanhamento dos Riscos, Compromissos, Ações Pendentes, Indicadores | * Equipe do Projeto | * Semanal | * Ata de reunião * Relatório de situação do projeto * feedback do resto equipe sobre as atividades exercidas pela equipe de desenvolvimento |
| * Equipe do projeto | * Semanal |
| * Comunicar situação do projeto | * Equipe * Prof/Monitor | * Semanal | * Ata de reunião * Relatório de situação do projeto |

## Gerenciamento de Riscos

| **Risco** | **Grau de Exposição** | **Mitigação** | **Plano de Contingência** |
| --- | --- | --- | --- |
| Atrasos na entrega | Alto | Estabelecer cronograma com prazos realistas e reuniões de acompanhamento | Priorizar tarefas, focar nas funcionalidades essenciais. |
| Bugs e falhas no aplicativo | Alto | Testar cada funcionalidade antes de avançar para a próxima etapa | Corrigir imediatamente, usar ferramentas de depuração e controle de versões |
| Falta de engajamento da equipe | Médio | Divisão clara de tarefas, comunicação constante e metas curtas | Redistribuir tarefas |
| Má gestão do tempo | Alto | Uso de cronogramas e métodos ágeis | Eliminar ou adiar itens de menor prioridade para garantir entrega mínima viável |
| Problemas com ferramentas | Médio | Usar ferramentas estáveis, backups frequentes e controle de versão (Git) | Mudar a ferramenta, restaurar backup |

2.5.2 Critérios para atrasos na entrega (Risco alto)

* Atraso superior a 20% no cronograma planejado
* Duas sprints consecutivas com menos de 70% das entregas concluídas
* Impossibilidade de cumprimento de um macro crítico na data prevista

2.5.2 Critérios para atrasos na entrega (*Risco Alto)*



## Critérios de Replanejamento

**2.6.1 Critérios relacionados ao cronograma de entregas**

| ***Critério*** | ***Descrição*** | ***Risco Associado*** | **plano de Mitigação/Contingência** |
| --- | --- | --- | --- |
| *Atraso de Sprint* | *Sprint com menos de 70% das histórias concluídas* | *Risco de atraso na entrega final* | *Mitigação: Revisão diária do progresso das tarefas*  *Contingência: Realocação de recursos ou ajuste de escopo* |
| *Acúmulo de débito técnico* | *Débito técnico ultrapassando 20% do backlog total* | *Risco de comprometimento de qualidade* | *Mitigação: Reservar tempo na Sprint para tratamento de débitos*  *Contingência: Sprint dedicada à redução de débito técnico* |
| *Desvio no tempo de entrega* | *Desvio superior a 15% do planejado por duas sprints consecutivas* | *Risco de não cumprimento do prazo final* | *Mitigação: Monitoramento diário das estimativas*  *Contingência: Revisão do Escopo ou aumento da equipe* |

***2.6.2 Critérios relacionados aos requisitos e usabilidade***

| *Critério* | *Descrição* | *Risco associado* | *Plano de mitigação/Contingência* |
| --- | --- | --- | --- |
| *Alterações significativas em requisitos* | *Mudanças que impactam mais de 25% das funcionalidades* | *Risco de escopo inadequado* | *Mitigação: Validação contínua com stakeholders<br>Contingência: Congelamento de escopo com priorização* |
| *Baixa usabilidade para o público-alvo* | *Menos dos 70% dos usuários analfabetos conseguirem completar tarefas básicas* | *Risco de baixa adoção* | *Mitigação: Testes frequentes com usuários reais<br>Contingência: Redesenho de interface com foco em acessibilidade* |
| *Feedback negativo recorrente* | *Funcionalidade com feedback negativo de mais de 50% dos usuários* | *Risco de insatisfação do cliente* | *Mitigação: Protótipos validados antes da implementação<br>Contingência: Redesenho da funcionalidade problemática* |

***2.6.3 Critérios relacionados à qualidade e testes***

| ***Critérios*** | ***Descrição*** | ***Risco associado*** | ***Plano de mitigação/Contingência*** |
| --- | --- | --- | --- |
| *Taxa de bugs elevadas* | *Mais de 5 bugs críticos/bloqueadores por Sprint* | *Risco de instabilidade do produto* | *Mitigação:Práticas TDD e revisão de código*  *Contingência: Sprint focada em correção de bugs* |
| *Reprovação em homologação* | *Funcionalidade reprovada mais de 2 vezes* | *Risco de atraso na entrega* | *Mitigação: Testes de aceitação<br>Contingência: Revisão dos critérios de aceitação* |
| *Falhas recorrentes* | *Mesmo componente com falhas em mais de 3 ciclos* | *Risco de degradação da arquitetura* | *Mitigação: Revisão arquitetural proativa*  *Contingência: Refatoração do componente problemático* |

**2.6.4 Critérios relacionados a recursos e infraestrutura**

| **Critérios** | **Descrição** | **Risco associado** | **Plano de mitigação/Contingência** |
| --- | --- | --- | --- |
| Indisponibilidade da equipe | Redução de mais de 20% na capacidade da equipe | Risco de atraso em entregas | Mitigação: Documentação e compartilhamento de conhecimento  Contingência: Redistribuição de tarefas ou contratação |
| Limitações técnicas | Identificação de limitações que comprometam a arquitetura | Risco de inviabilidade técnica | Mitigação: Provas de conceito antecipadas  Contingência: Redesenho da solução técnica |
| Problemas de integração | Falhas de integração persistentes por mais de uma sprint | Risco de incompatibilidade de Sistemas | Mitigação: Testes de integração contínuos<br>Continência: Desenvolvimento de adaptadores ou APIs alternativas |

**2.6.5 Processo de monitoramento e atualização:**

Os critérios de replanejamento serão revisados e atualizados:

* Ao final de cada ciclo/Sprint
* Após a identificação de novos riscos
* Quando um replanejamento for realizado

Cada atualização nos critérios de planejamento ou qualquer alteração no projeto que demande replanejamento resultará em uma nova versão deste documento, seguindo controle de versões estabelecido.

# PROCESSO DE DESENVOLVIMENTO DE SOFTWARE

#### Metodologia Adotada

A equipe optou por utilizar uma abordagem baseada em metodologias ágeis, especialmente os conceitos do Scrum e do XP. A aplicação desses métodos visa garantir uma organização eficiente do trabalho, entregas contínuas, priorização de valor para o usuário e capacidade de adaptação ao longo do projeto.

#### Visão Geral do Processo

O processo de desenvolvimento é cíclico e iterativo, seguindo os seguintes estágios principais:

* Backlog e Escopo: levantamento inicial das funcionalidades desejadas, criando uma lista priorizada de tarefas.
* Design de UI/UX: criação de protótipos de interface e fluxos de navegação.
* Sprints: o desenvolvimento é dividido em ciclos curtos (sprints), com entregas incrementais e reuniões de planejamento, acompanhamento e retrospectiva.
* Testes: validação constante do sistema com testes manuais e automatizados.
* Desenvolvimento Continuo: integração frequente de novas funcionalidades e correções.

#### Procedimentos Técnicos

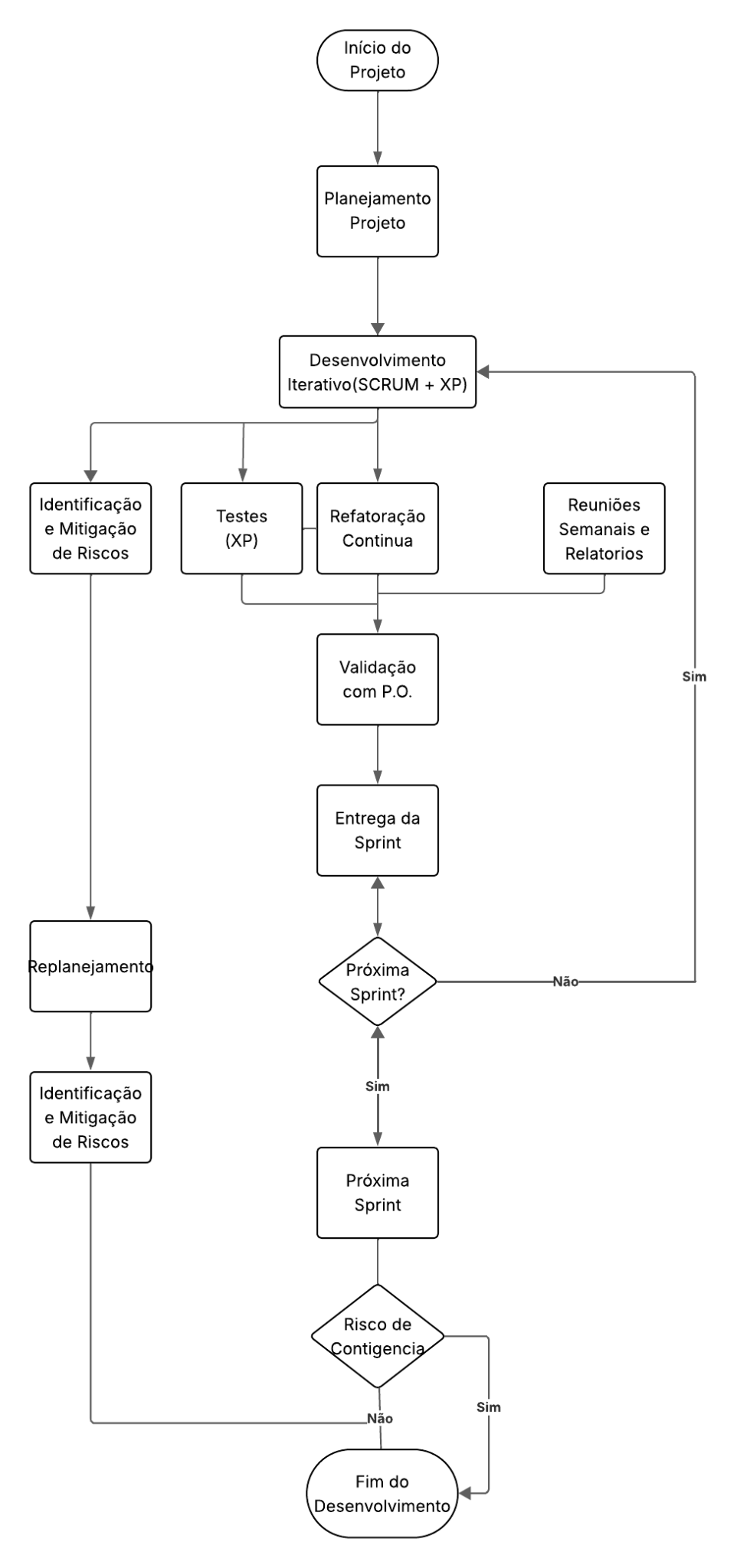
* Backend: desenvolvido em TypeScript.
* Banco de Dados: utiliza o PostgreSQL para armazenamento e gerenciamento das informações.
* Frontend: construído com React Native utilizando Expo, permitindo maior agilidade no desenvolvimento de aplicativos mobile.

#### Métodos Utilizados

* Aplicação de práticas do Scrum e XP, com foco em:  
  + Desenvolvimento incremental.
  + Comunicação constante entre os membros da equipe.
  + Entregas frequentes e com feedback.
  + Código limpo e práticas como refatoração (quando aplicável).
* O projeto é orientado a funções, priorizando funcionalidades com valor claro ao usuário.

#### Ferramentas

* VSCode: ambiente de desenvolvimento utilizado por toda a equipe.
* GitHub: plataforma de versionamento e colaboração no código.
* Docker: utilizado para criação de ambientes consistentes e padronizados.



# DECLARAÇÃO DE ESCOPO DO PROJETO

## Backlog do produto

O backlog do produto foi construído com base em pesquisas sobre a temática da ODS 4 (analfabetismo e anumerismo em território nacional), reuniões de brainstorm semanais, levando em consideração o prazo até entrega final, decidiu-se focar na simplicidade e interatividade da aplicação mobile, definiu-se graus de prioridade(must, should, could), conforme a importância de cada parte.

## Perfis

*Aluno: usuário que acessa os conteúdos e faz os exercícios.*

*Administrador: gerencia os conteúdos, usuários, questões e progresso do usuário.*

Tabela : Perfis de acesso

| # | Nome do perfil | Características do perfil | Permissões de acesso |
| --- | --- | --- | --- |
| *<1>* | Aluno | Usuário que acessa os conteúdos educacionais. | Acesso às aulas, e as listas de exercícios. |
| *<2>* | Administrador | Responsável por manter os perfis de acesso da aplicação, criar novos usuários, alterar usuários já existentes, ou excluir usuários (Manter usuários) | Permissões para criar, editar e remover conteúdos educacionais, gerenciar usuários, visualizar estatísticas e relatórios de desempenho dos alunos, configurar categorias, níveis e parâmetros do sistema. |

## Cenários

Tabela : Cenários funcionais

| Sistema: Omega | | |
| --- | --- | --- |
| Numeração do cenário | Nome do cenário | Sprints |
| 01 | Cadastro e Login | 1 |
| 02 | Realização de Atividades | 2 |
| 03 | Sistema de Progresso | 2 |
| 04 | Gerenciamento de Conteúdo | 3 |
| 05 | Testes e geração do apk | 4 |

*.*

## Tabela de Backlog do produto

Tabela : Backlog do produto

| **Sistema:Omega(**Ω**) – Backlog do produto** | | | | | |  |
| --- | --- | --- | --- | --- | --- | --- |
| Numeração  (Cenário / requisito) | Sprint | Nome do requisito | Tipo de requisito  (Funcional / não funcional) | Priorização do requisito  Must, Should, Could | Descrição sucinta do requisito | **User stories (U.S.) associadas**  Identifique as U.S. associadas ao requisito |
| 1.1 | 1 | Cadastro de Usuário | Funcional | Must | O sistema deve permitir que o aluno crie uma conta com email e senha | Como aluno, quero poder me cadastrar para acessar o conteúdo do aplicativo. |
| 1.2 | 1 | Login de Usuário | Funcional | Must | O sistema deve permitir autenticação por email e senha. | Como aluno, quero poder fazer login para continuar meus estudos. |
| 2.1 | 2 | Questões de português | Funcional | Must | Deve ter 3 questões em cada um dos 3 níveis em português. 1- estrutura gramatical. 2- pontuação e 3 - interpretação básica de texto | Como aluno, quero poder realizar questões de portugues. |
| 2.2 | 2 | Questões de matemática | Funcional | Must | Deve ter 3 questões em cada um dos 3 níveis em matemática.1 - multiplicação (tabuada). 2- divisão/fração. 3- expressão numérica (soma, subtração, divisão, multiplicação) | Como aluno, quero poder realizar questões e aprender matemática básica. |
| 3.1 | 3 | Sistema de progresso | Funcional | Must | Deve ser possível recomeçar da atividade onde o usuário parou na última interação com o aplicativo. | Como aluno, quero poder ter continuidade nos estudos com um sistema de progressão. |
| 3.2 | 3 | Sistema de correção | Funcional | Must | Ao término de cada nível, será dado ao usuário um feedback sobre as atividades, erros e acertos, corretos grifados em verde, errados em grifados vermelha, e se tiver uma taxa de acerto menor que 2, o usuário volta ao início do nível, após o feedback. | Como aluno, quero saber quais questões acertei e quais errei. |
| 4.1 | 4 | Painel do administrador | Funcional | Should | O administrador deve poder adicionar, editar ou remover atividades dentro dos níveis. Não há acesso direto do administrador aos dados do usuário, apenas ao sistema de questões. | Como administrador, quero atualizar os conteúdos do aplicativo. |
| 4.2 | 4 | Aba de créditos | Não funcional | Should | O aplicativo deve ter uma aba que indica os créditos de cada questão. | Como aluno, quero saber o material bibliográfico das questões que realizo para aprofundar os estudos |
| 5.1 | 5 | Testes finais, geração do apk | Funcional | Must | O aplicativo deve funcionar perfeitamente sem bugs ou erros, mesmo havendo testes após cada sprint será feito um teste geral intenso no final. Logo após esse teste intenso será feito o deploy que no caso é a geração do APK. | Como aluno, quero baixar o aplicativo em meu celular (android) , ter uma boa experiência com o app, realizando as atividades de forma linear e com as funcionalidades sem bugs. |
| 5.2 | 5 | Modo Offline | Não funcional | Could | O aplicativo deve funcionar parcialmente sem conexão, salvando a progressão do usuário. | Como aluno, quero poder estudar mesmo sem internet. |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

# MÉTRICAS E MEDIÇÕES

## GQM de medições

Para garantir a qualidade e eficácia do aplicativo mobile voltado à alfabetização de pessoas analfabetas e semianalfabetas, adotamos a abordagem **GQM (Goal / Question / Metric)** como método de definição e acompanhamento de métricas.

* **Objetivo (Goal)**

Analisar o aplicativo de alfabetização com o propósito de avaliar sua usabilidade, eficácia pedagógica e impacto no aprendizado, do ponto de vista dos usuários e educadores, no contexto da inclusão digital e social.

* **Questões (Questions)**
* O aplicativo é fácil de usar, mesmo por pessoas com baixa familiaridade com tecnologia?
* Os usuários estão conseguindo se desenvolver com o uso do aplicativo?
* Quanto tempo os usuários dedicam ao uso do app diariamente?
* Quanto tempo os usuários dedicam ao uso do app diariamente?
* O desempenho dos usuários melhora com o uso contínuo do aplicativo?
* O conteúdo é acessível e adequado aos diferentes níveis de alfabetização?
* **Métricas (Metrics)**
* Taxa de conclusão de atividades (% de atividades concluídas com sucesso por usuário)
* Tempo médio por sessão de uso
* Número de palavras aprendidas por semana (por usuário)
* Número de acessos diários/mensais
* Nível de erro nas atividades de escrita e leitura (antes e depois de certo tempo de uso)
* Pontuação média nas avaliações integradas ao app
* Tempo até o abandono da aplicação (churn rate)
* Feedback qualitativo dos usuários e educadores (via formulários ou entrevistas)
* Quantidade de recursos multimodais utilizados (áudio, imagens, vídeos) para facilitar a compreensão
* **Formas alternativas e/ou informais**
* Modelo ISO/IEC 25010 ( Funcionalidade, Usabilidade, Confiabilidade, Eficiência, Portabilidade, Manutenibilidade, Segurança, Compatibilidade).
* Indicadores de Acessibilidade Digital (WCAG)
* Análise de Logs e Eventos
* Observação direta dos usuários
* Testes com Usuários
* Relatos de educadores e mediadores
* Observações de uso em campo

# TESTES DE SOFTWARE 6.1 Estratégia de testes contendo: 6.1.1 Níveis de Testes Abordados

| **Nível de Teste** | **Descrição** | **Prioridade** | **Objetivo** |
| --- | --- | --- | --- |
| **Testes Unitários** | Testes isolados de componentes individuais | Alta | Validar o funcionamento correto das funções e métodos específicos que implementam a lógica educacional |
| **Testes de Integração** | Verificação da comunicação entre módulos | Média | Garantir que os diferentes componentes da aplicação funcione corretamente quando integrados |
| **Testes de Sistema** | Avaliação do sistema como um todo | Baixa | Verificar o funcionamento completo da aplicação em um ambiente próximo ao de produção |
| **Testes de Aceitação** | Validação dos requisitos de negócio | Média | Confirmar que a aplicação atende às expectativas |

**6.1.2 Tipos de Testes Abordados**

| **Tipo de Teste** | **Descrição** | **Técnicas Aplicadas** | **Ferramentas** |
| --- | --- | --- | --- |
| **Testes Funcionais** | Verificação das funcionalidades específicas | Particionamento de equivalência, análise de valor limite | JUnit, TestNG |
| **Testes de Usabilidade** | Avaliação da experiência do usuário | Testes com usuários representativos, heurísticas de Nielsen | Ferramentas de gravação de sessão |
| **Testes de Desempenho** | Análise do tempo de resposta e uso de recursos | Testes de carga, testes de estresse | JMeter, Gatling |
| **Testes de Segurança** | Verificação de vulnerabilidades | OWASP Top 10, análise estática de código | SonarQube, OWASP ZAP |
| **Testes de Acessibilidade** | Validação da acessibilidade para diferentes perfis | WCAG 2.1, testes com tecnologias assistivas | Lighthouse, Axe |

**6.1.3 Ambientes de Testes**

| **Ambiente** | **Propósito** | **Configuração** | **Responsável** |
| --- | --- | --- | --- |
| **Desenvolvimento** | Execução de testes unitários durante o desenvolvimento | Ambiente local com frameworks de teste | Desenvolvedores |
| **Integração Contínua** | Execução automática de testes após cada commit | Pipeline CI/CD com Jenkins/GitHub Actions + SonarQube | DevOps e QA |
| **Pré-produção** | Testes integrados em ambiente similar ao de produção | Configuração espelhando o ambiente de produção | Equipe de QA |
| **Dispositivos Físicos** | Testes em diferentes hardwares e sistemas operacionais | Conjunto de smartphones com diferentes especificações e versões de SO | Equipe de QA |

**6.1.4 Métricas de Análise do SonarQube**

| **Métrica** | **Descrição** | **Meta** | **Impacto** |
| --- | --- | --- | --- |
| **Cobertura de Código** | Percentual de código coberto por testes unitários | ≥ 80% | Garantir que a maioria do código foi testada |
| **Bugs** | Problemas que representam falhas no código | 0 (críticos e altos) | Evitar erros durante a execução |
| **Vulnerabilidades** | Potenciais brechas de segurança | 0 (críticas e altas) | Proteger dados dos usuários |
| **Code Smells** | Problemas de manutenibilidade | < 20 por 1000 linhas | Facilitar manutenção futura |
| **Duplicação de Código** | Código repetido | < 5% | Melhorar manutenibilidade |
| **Complexidade Ciclomática** | Complexidade de caminhos de execução | < 10 por método | Facilitar manutenção e testes |
| **Débito Técnico** | Tempo estimado para resolver issues | < 5% do tempo de desenvolvimento | Manter qualidade ao longo do tempo |

**6.1.5 Plano de Testes Unitários**

| **Componente** | **Casos de Teste** | **Mocks Necessários** | **Critérios de Aceitação** |
| --- | --- | --- | --- |
| **Módulo de Reconhecimento de Fonemas** | Identificação de fonemas consonantais, vocálicos, validação de pronúncia | API de áudio, serviço de feedback | 100% de assertividade no reconhecimento de fonemas básicos |
| **Módulo de Alfabetização** | Formação de sílabas, construção de palavras, leitura de frases simples | Banco de dados de palavras, sistema de pontuação | Progressão correta entre níveis de dificuldade |
| **Módulo de Matemática Básica** | Reconhecimento numérico, operações básicas (adição, subtração, multiplicação, divisão) | Gerador de problemas, calculadora | Cálculos corretos e verificação de resultados |
| **Sistema de Progressão** | Avanço entre níveis, desbloqueio de conteúdo, relatórios de progresso | Perfil de usuário, histórico de atividades | Correto controle de acesso aos níveis baseado no desempenho |
| **Interface do Usuário** | Renderização de elementos, resposta a interações, feedback visual e sonoro | Simulador de eventos de UI, renderizador | Conformidade com diretrizes de design e acessibilidade |

## 6.2 Roteiro de teste:

| **ID do Teste** | **Nome do Teste** | **Objetivo do Teste** | **Nível** | **Tipo** | **Pré-condições** | **Critério de Aceitação** | **Resultado/Evidencia** | **Reparos Executados** | **Ciclos Executados** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| T01 | Cadastro de novo usuário | Verificar se o usuário pode se cadastrar com dados válidos | Sistema | Funcional | App instalado, acesso a tela de cadastro | Usuário é redirecionado à tela inicial e recebe mensagem de boas-vindas |  |  |  |
| T02 | Login com credenciais válidas | Verificar se o login funciona com e-mail e senha corretos | Sistema | Funcional | Conta já cadastrada | Redirecionamento para o dashboard com os dados do usuário |  |  |  |
| T03 | Login com credenciais inválidas | Validar o tratamento de erro em tentativa de senha incorreta | Sistema | Funcional | Conta existente com senha conhecida | Mensagem de erro é exibida e usuário permanece na tela de login |  |  |  |
| T04 | Execução de Lição completa | Garantir que a lição salva progresso após concluída | Sistema | Funcional | Usuário logado, lição disponível | Ao finalizar a lição, progresso é salvo e registrado no perfil |  |  |  |
| T05 | Feedback interativo | Verificar se o app oferece retorno visual e sonoro após resposta correta | Sistema | Funcional | Lição ativa com perguntas | Resposta correta aciona som positivo e animação; resposta errada gera alerta |  |  |  |
| T06 | Teste de Responsividade | Avaliar se o layout adapta-se a diferentes tamanhos de tela | Sistema | Não Funcional (portabilidade) | Simulação em celular, tablet e web | Elementos do layout se ajustam sem sobreposição ou controle de conteúdo |  |  |  |
| T07 | Teste de Acessibilidade | Garantir contraste adequado e legibilidade para usuários com baixa visão | Sistema | Não Funcional(usabilidade) | App em modo escuro/claro | Textos visíveis, com contraste e espaçamento adequadosTextos visíveis, com contraste e espaçamento adequados |  |  |  |
| T08 | Teste de desempenho do login | Medir tempo de resposta após clicar em "Entrar” | Sistema | Não funcional (desempenho) | Backend operacional e servidor ativo | Tempo de resposta inferior a 2 segundos para login com rede estável |  |  |  |
| T09 | Registro de progresso | Validar se o progresso em matemática e leitura é registrado no perfil | Integração | Funcional | Conta logada, atividades feitas | Percentuais atualizados no painel do usuário após concluir atividades |  |  |  |
| T10 | Erro de campo obrigatório | Verificar se o sistema detecta cadastro com campos vazios | Sistema | Funcional | Tela de cadastro aberta | Exibição de mensagem de erro clara para cada campo vazio |  |  |  |

# REFERÊNCIAS BIBLIOGRÁFICAS

| 1. **IBGE.** Censo 2022: taxa de analfabetismo cai de 9,6% para 7,0% em 12 anos, mas desigualdades persistem. Agência de Notícias IBGE, Rio de Janeiro, 17 maio 2024. Disponível em:<https://agenciadenoticias.ibge.gov.br/agencia-noticias/2012-agencia-de-noticias/noticias/40098-censo-2022-taxa-de-analfabetismo-cai-de-9-6-para-7-0-em-12-anos-mas-desigualdades-persistem>. Acesso em: 15 abr. 2025 2. **TOKARNIA, Mariana.** *Resultados do Pisa reforçam gargalo no ensino de matemática no Brasil.* Agência Brasil, Rio de Janeiro, 5 dez. 2023. Disponível em:<https://agenciabrasil.ebc.com.br/educacao/noticia/2023-12/resultados-do-pisa-reforcam-gargalo-no-ensino-de-matematica-no-brasil>. Acesso em: 15 abr. 2025. |
| --- |