

Definir Critérios da Tarefa de IA e Escrever Prompts

Passo 1: Definir Critérios Específicos da Tarefa (Framework CLEAR)

Use o framework CLEAR para delinear os critérios específicos que sua ferramenta de IA deve seguir ao analisar arquivos de log.

1. Relevância do Contexto

- *Em que escopo a IA deve focar? Existem tipos específicos de logs ou palavras-chave para priorizar?*
- Exemplo: "Focar em erros e avisos relacionados à atividade do usuário; excluir logs informativos, a menos que sejam marcados como críticos."

Focar em ensino relacionado a Programação para crianças.

2. Tom e Estilo da Linguagem

- *Qual tom a IA deve usar (ex.: técnico, amigável)?*

Didática simples e divertida de fácil entendimento que uma criança de 6 anos consiga entender

3. Tratamento de Erros e Escalonamento

- *Como a IA deve responder se houver incerteza ou dados faltando?*

- Reconhecer a dúvida com leveza
 - Exemplo: "Hmm, não tenho certeza dessa parte... vamos descobrir juntos?"
 - Isso mostra que errar ou não saber faz parte do aprendizado.
- Transformar em oportunidade de exploração
 - Incentivar a criança a pensar: "O que você acha que pode acontecer se tentarmos?"
 - Estimula curiosidade e raciocínio lógico.
- Dar pistas em vez de respostas diretas
 - Usar perguntas guiadas: "Se o computador não sabe o que imprimir, o que você acha que ele mostra?"
 - Isso ajuda a criança a construir a resposta.
- Usar analogias simples e divertidas
 - Exemplo: "É como montar um quebra-cabeça sem todas as peças. O que podemos fazer?"
 - Facilita a compreensão com imagens familiares.
- Manter o tom positivo e encorajador
- Nunca dizer "isso está errado" de forma dura.
- Preferir: "Quase lá! Vamos tentar de outro jeito?"

4. Precisão e Confiabilidade das Respostas

- *Como você garantirá que as respostas da IA sejam confiáveis e baseadas em dados?*

1. Validação de conteúdo

- Fontes confiáveis: A IA deve buscar informações em fontes verificadas e atualizadas, como sites educacionais, institutos de pesquisa e bases de dados reconhecidas.
- Revisão por especialistas: Conteúdos sensíveis ou complexos devem ser revisados por educadores ou especialistas em pedagogia infantil.

2. Monitoramento de desempenho

- Testes de qualidade: Avaliar regularmente se as respostas da IA são coerentes, corretas e apropriadas para a faixa etária.
- Indicadores de precisão: Usar métricas como taxa de acerto, relevância das respostas e similaridade com respostas esperadas.

3. Transparência e rastreabilidade

- Explicações simples: A IA deve explicar de onde vem a informação, com linguagem acessível para crianças.
- Registro de fontes: Internamente, manter um log das fontes utilizadas para cada resposta, facilitando auditorias.

4. Gestão de incerteza

- Reconhecer limitações: Quando não souber algo, a IA deve dizer isso de forma acolhedora, como: *“Ainda não sei essa parte, mas podemos descobrir juntos!”*
- Evitar respostas inventadas: A IA nunca deve “chutar” ou inventar fatos, especialmente em contextos educacionais.

5. Adaptação ao público infantil

- Linguagem apropriada: Respostas devem ser simples, lúdicas e seguras para crianças **de 6** anos.
- Evitar termos técnicos sem explicação: Sempre que usar uma palavra difícil, a IA deve explicar com analogias ou exemplos visuais.

5. Eficiência da Resposta

- *Que tipos de perguntas a IA deve ser capaz de responder? Como a IA deve adaptar suas respostas com base na complexidade da pergunta?*

Ela deve responder perguntas voltadas à educação de desenvolvimento de programação, ela deve explicar de modo simples e criativo de modo que uma criança consiga entender.

Passo 2: Escrever Prompts (Framework TRACI)

Usando o framework TRACI, crie de 1 a 3 prompts que direcionem a ferramenta de IA para executar cada tarefa.

- Tarefa 1:

Prompt 1

“Explique o que é uma variável como se fosse uma caixinha de brinquedos onde podemos guardar números ou palavras. Use exemplos divertidos e fáceis de imaginar.”

Prompt 2

“Explique o que é um loop como se fosse uma dança repetida. Crie uma historinha com um robô que dança sempre os mesmos passos até alguém dizer ‘pare’.”

Prompt 3

“Dê um desafio divertido: peça para a criança escrever um código que faça o computador mostrar o nome dela na tela. Explique passo a passo e comemore quando ela conseguir.”

Prompt 4

“Dê um desafio divertido: peça para a criança escrever um código que faça o computador mostrar o nome dela na tela. Explique passo a passo e comemore quando ela conseguir.”

Direções

1. Introdução

- **Objetivo da Guia**

- Estabelecer padrões de qualidade: garantir que a IA não forneça respostas inventadas ou imprecisas.
- Orientar o uso de fontes confiáveis: direcionar a IA a se basear em dados verificados e atualizados.
- Promover transparência: indicar como a IA deve explicar suas respostas de forma clara e rastreável.
- Gerenciar incertezas: ensinar a IA a reconhecer quando não possui informações suficientes e responder de forma adequada.
- Adaptar ao público-alvo: assegurar que as respostas sejam compreensíveis, seguras e apropriadas para crianças de 6 anos.

- **Escopo**

1. Escopo de Conteúdo

- **Inclui:**
 - Conceitos básicos de programação (variáveis, loops, comandos simples).
 - Exercícios lúdicos e criativos (histórias, jogos, desafios simples).
 - Explicações com analogias acessíveis (caixinhas, brinquedos, danças).
- **Exclui:**
 - Programação avançada (estruturas complexas, algoritmos sofisticados).
 - Conteúdos técnicos sem adaptação infantil.
 - Temas fora da programação (matemática avançada, ciências complexas).

2. Escopo de Linguagem

- **Inclui:**
 - Linguagem simples, clara e divertida.
 - Uso de metáforas e histórias para explicar conceitos.
 - Tom encorajador e positivo.
- **Exclui:**
 - Jargão técnico
 - Linguagem adulta ou formal demais.
 - Respostas negativas ou desmotivadoras.

3. Escopo de Interação

- **Inclui:**
 - Responder dúvidas com paciência e leveza.
 - Transformar erros em oportunidades de aprendizado.
 - Estimular curiosidade com perguntas abertas.
- **Exclui:**
 - Respostas ríspidas ou críticas severas.
 - Incentivo a práticas inseguras ou inadequadas.
 - Dependência emocional (a IA não substitui professores ou familiares).

4. Escopo de Confiabilidade

- **Inclui:**
 - Basear respostas em dados corretos e pedagógicos.
 - Admitir quando não se sabe algo e propor exploração conjunta.
 - Garantir segurança e adequação ao público infantil.
- **Exclui:**
 - “Chutes” ou informações inventadas.
 - Conteúdo não verificado ou sem fonte confiável.
 - Qualquer material impróprio para crianças.

2. Visão Geral da Ferramenta

- **Descrição**

- ***Estabelecer padrões de qualidade:*** garantir que a IA não forneça respostas inventadas ou imprecisas.
- ***Orientar o uso de fontes confiáveis:*** direcionar a IA a se basear em dados verificados e atualizados.
- ***Promover transparência:*** indicar como a IA deve explicar suas respostas de forma clara e rastreável.
- ***Gerenciar incertezas:*** ensinar a IA a reconhecer quando não possui informações suficientes e responder de forma adequada.
- ***Adaptar ao público-alvo:*** assegurar que as respostas sejam compreensíveis, seguras e apropriadas para crianças de 6 anos.

- **Principais Recursos**

1. Recursos pedagógicos

- Explicações simples e lúdicas de conceitos de programação.
- Uso de analogias (caixinhas, brinquedos, danças) para facilitar a compreensão.
- Exercícios práticos e desafios divertidos.

2. Recursos de linguagem

- Comunicação clara e acessível para crianças de 6 anos.
- Tom positivo, encorajador e paciente.
- Frases curtas e exemplos visuais ou narrativos.

3. Recursos de interação

- Capacidade de transformar erros em oportunidades de aprendizado.
- Respostas acolhedoras diante de incertezas ou dúvidas.
- Estímulo à curiosidade com perguntas abertas e criativas.

4. Recursos de confiabilidade

- Basear respostas em informações corretas e verificadas.
- Reconhecer limitações e admitir quando não sabe algo.
- Garantir segurança e adequação ao público infantil.

3. Processo de Desenvolvimento

Guia Passo a Passo

Definição do Objetivo

O objetivo da ferramenta é ensinar programação para crianças de forma divertida e fácil de entender.

Queremos que cada explicação seja como uma historinha ou brincadeira, para que a criança aprenda sem perceber que está estudando.



Coleta de Dados

Exemplos simples de programação (como variáveis e loops).

Analogias lúdicas (caixa de brinquedos, dança repetida, robô obediente).

Atividades práticas (mostrar o nome na tela, repetir frases).

Feedback de professores e crianças, para saber se as explicações estão claras e divertidas.

Pré-processamento dos Dados

Antes de usar os dados, organizamos tudo para ficar fácil:

- Tiramos exemplos muito complicados.
- Simplificamos a linguagem para frases curtas.
- Adaptamos os desafios para o nível infantil.
- Garantimos que cada explicação tenha uma analogia divertida.

Treinamento e Refinamento da Ferramenta

A ferramenta aprende com os exemplos e vai melhorando:

- Testamos explicações com crianças e professores.
- Ajustamos quando algo não ficou claro.
- Refinamos as respostas para que sejam sempre objetivas, seguras e divertidas.
- Mantemos o foco em transformar erros em oportunidades de aprendizado.

Testes e Avaliação

Depois de pronta, a ferramenta é testada:

- Com crianças: para ver se entendem e se se divertem.
- Com professores: para garantir que o conteúdo esteja correto.
- Com exemplos práticos: como criar uma variável ou repetir uma frase.
- Avaliamos se a criança consegue explicar o que aprendeu com suas próprias palavras — esse é o sinal de sucesso!

4. Principais Desafios e Soluções

- **Desafio 1**
 - Manter a atenção das crianças
 - As crianças se distraem rápido.
 - Solução: usar histórias, cores e exemplos do dia a dia (brinquedos, jogos, músicas).
 - Transformar cada conceito em uma brincadeira.
 - **Desafio 2**
 - Explicar sem complicar
 - A programação pode parecer difícil.
 - Solução: usar analogias simples, como “variável é uma caixa de brinquedos” ou “loop é uma dança repetida”.
 - **Dicas para o Sucesso**
 - Sempre começar com exemplos práticos. Reforçar que errar faz parte do aprendizado.
 - Incentivar a criança a explicar com suas próprias palavras o que entendeu.
 - Usar desafios curtos e divertidos
-

5. Possíveis Casos de Uso

- **Além da Análise de Logs**
 - A ferramenta pode ser usada para:
 - Ensinar conceitos básicos de programação em sala de aula.
 - Criar atividades interativas para crianças em casa.
 - Apoiar professores com exemplos prontos e analogias lúdicas.
 - **Oportunidades Futuras**
 - **Jogos educativos que ensinam lógica de programação.** Aplicativos que transformam erros em desafios divertidos.
 - **Expansão para outras áreas, como matemática ou ciências, usando a mesma linguagem simples.**
-

6. Instruções para o Usuário

- **Como Usar a Ferramenta**
- **Escolha um conceito (ex.: variável, loop).** Leia a explicação simples e divertida.
- **Faça o desafio proposto com a criança.**

- Incentive que ela crie sua própria versão.
 - Entradas e Saídas
 - Entrada: o usuário escolhe o conceito ou desafio.
 - Saída: explicação clara, analogia divertida e atividade práticas
 - Dicas para o Usuário
 - Use sempre exemplos próximos da realidade da criança.
 - Repita os exercícios em forma de jogo.
 - Valorize cada tentativa, mesmo quando não dá certo.
-

7. Cenários de Exemplo

- Cenário 1
 - Variável
 - A professora explica: “Uma variável é como uma caixa de brinquedos. Você pode guardar um número ou uma palavra dentro dela.”
 - Desafio: peça para a criança criar uma variável chamada **meu_nome** e colocar o próprio nome dentro.
- Cenário 2
 - Loop
 - A professora diz: “Um loop é como uma dança repetida. O robô dança até alguém dizer para parar.”
 - Desafio: peça para a criança escrever um loop que repete a frase “Eu gosto de programar!” cinco vezes.

SALA 1:

Allyson Gonçalves
Ana Beatriz Monteiro
Beatriz Alexandrino
Bruna Melo
Ewerton Bezerra