

O que é, o que faz e o que entrega

1. O que é

O CEBI Tester é a ferramenta de **validação funcional de liberação de versão** dos sistemas.

Ele executa, sem intervenção, os casos de teste já catalogados contra o ambiente de um cliente, recolhe evidência de cada passo e emite um **veredito sobre a versão**: apta, apta com ressalvas ou não recomendada.

A pergunta que ele responde não é "*a tela abriu?*", e sim:

“ O processo de negócio continuou funcionando do início ao fim, e os dados que ficaram gravados estão consistentes?

É a diferença entre a tela dizer que gravou e **ter gravado**. Quando o caso permite, a ferramenta confere no banco o que a tela afirmou.

2. O problema que ele resolve

A validação de uma versão era feita a partir de checklists em texto, lidos e executados à mão a cada liberação. Isso tem três custos:

Custo	O que acontecia
Tempo	Cada liberação recomeçava do zero, tela a tela
Cobertura	O que não dava tempo de conferir simplesmente não era conferido — e ninguém sabia o que ficou de fora
Rastro	"Testei e passou" não é evidência; sem print e sem dado, a conversa com o desenvolvimento vira opinião contra opinião

O CEBI Tester transforma cada item de checklist em **caso permanente**: escrito uma vez, executado quantas vezes for preciso, sempre do mesmo jeito, sempre deixando prova.

3. O que ele faz, numa execução

Cada rodada percorre esta sequência, sozinha:

1. **Identifica o ambiente** do cliente escolhido e valida que ele responde.
2. **Detecta a versão publicada** lendo o próprio sistema — a versão nunca é digitada, porque muda várias vezes ao dia e sobe por módulo.
3. **Prepara a base**, quando o cliente exige preparação (perfil de permissão, massa, parâmetros).
4. **Seleciona os casos** no catálogo, conforme o escopo pedido.
5. **Executa** pela interface e pela API, como um usuário faria.
6. **Captura evidência** de cada passo: prints, dados da tela, requisições e respostas, console e rede.
7. **Confere na base** o que a tela afirmou, nos casos em que isso é possível.
8. **Classifica a causa** de cada falha — produto, ambiente, base ou a própria ferramenta.
9. **Grava o resultado** e **gera o relatório** da versão.

O passo 8 é o que diferencia a ferramenta de um robô de clique: falha não vira defeito automaticamente. Falta de massa, ambiente fora do ar e permissão ausente saem como **bloqueio**, com o motivo medido — não como acusação ao produto.

4. O que ele entrega

Veredito da versão

Apta, apta com ressalvas ou não recomendada, com o que **impede liberar** explicitado: reprovados de severidade Crítica/Alta, casos sem causa identificada e módulos sem nenhuma medição.

Cobertura

Quanto do catálogo foi efetivamente exercido naquela rodada. Cobertura baixa não é defeito do produto — é aviso de que a versão foi pouco medida, e isso fica registrado em vez de passar despercebido.

Defeitos com evidência

Cada defeito sai com severidade, passo a passo para reproduzir, prints, dados e o contexto (versão, ambiente, caso). Dá para copiar e abrir o chamado sem reescrever nada.

Bloqueios por responsável

O que não pôde ser medido, agrupado por **de quem depende destravar**: ambiente, base ou a própria ferramenta. Isso vira lista de trabalho, não discussão.

Relatório da versão

Documento completo, com quatro escopos de impressão — do resumo executivo ao dossiê com evidências.

Histórico

Defeito que reincide é marcado como tal, inclusive com a contagem de versões seguidas. Registra também defeito **retirado**, para que uma retratação não suma sem rastro.

Catálogo permanente

A biblioteca de casos é o ativo que fica: hoje são **388 casos, 338 automatizados**, em **15 ambientes de cliente** configurados.

Produto	Casos	Automatizados
Balcão Comercial	275	275
Baixa	63	63
Hidrometria	49	—
Usuários	—	—

5. Como a IA entra (e onde ela não entra)

	Claude (IA)	CEBI Tester (programa)
Quando	Ao chegar uma versão nova; ao analisar falha sem causa determinada	Em toda execução
Faz	Interpreta as demandas, faz análise de impacto, seleciona a regressão, escreve os casos que faltam, redige o resumo executivo	Prepara a base, autentica, executa, compara, confere no banco, classifica a causa, gera relatório e evidências
Custa token	Sim	Não

Depois que um caso é escrito e aprovado, ele roda quantas vezes for preciso **sem IA**. O trabalho caro acontece uma vez; a execução é barata e repetível.

6. O que ele não faz

- **Não decide sozinho liberar.** Ele mede e sustenta a decisão com evidência; a decisão é do time.
- **Não testa desempenho nem carga.** O escopo é funcional.
- **Não envia nada para cliente.** E-mail e SMS só saem para destinos autorizados do time — nas bases de homologação o contato cadastrado costuma ser de pessoa real.
- **Não inventa resultado.** Quando não dá para medir, o caso sai bloqueado, com a causa. Caso bloqueado nunca é contado como aprovado.
- **Não cobre o que não está no catálogo.** O que não tem caso escrito aparece como módulo sem medição — a lacuna é exibida, não escondida.

7. Resultados até aqui

- A ferramenta encontrou defeitos que a conferência manual não pegava — entre eles, um modal de pagamento que mostrava só parte do valor devido, uma operação de desagrupamento que falha na primeira chamada e deixa o agrupamento pela metade, e uma segunda via de parcela que não é emitida em nenhum ambiente.
- E derrubou acusações falsas ao produto: casos que pareciam defeito e eram limitação da própria automação — grid paginado lido pela metade, aviso que some da tela em quatro segundos, botão julgado antes de preencher os campos obrigatórios. Cada um desses virou regra permanente da ferramenta.

A segunda parte vale tanto quanto a primeira: uma ferramenta que acusa o que não existe perde a confiança do desenvolvimento na primeira reunião.

