

Manual de atualização

Como atualizar uma instalação que já existe, sem perder configuração nem resultados.

Para instalar do zero, veja o [Manual de instalação](#).

“ **Dois coisas diferentes se chamam "versão"**. Este manual trata de atualizar **a ferramenta** (o CEBI Tester). Atualizar **a versão do sistema que será validado** é outra coisa — está no fim, na seção 7.

1. Antes de começar

A atualização leva poucos minutos, mas **não faça com execução em andamento**: abra a aba *Execuções* e confirme que nada está rodando.

Depois, pare o servidor: volte ao terminal onde ele está no ar e pressione `Ctrl + C`.

2. Conferir se há algo local a preservar

Dentro da pasta do projeto:

```
git status
```

- **"nothing to commit, working tree clean"** → pode seguir para o passo 3.
- Se aparecer uma lista de arquivos, há alterações feitas nesta máquina. Normalmente são ambientes novos em `config/ambientes/`. Guarde-as antes de atualizar:

```
git stash
```

Depois da atualização, para trazê-las de volta:

```
git stash pop
```

O arquivo `.env` **nunca** aparece nessa lista e nunca é alterado pela atualização — ele não é versionado de propósito.

3. Baixar a versão nova

```
git pull
```

O resumo no final mostra quantos arquivos mudaram. Se aparecer `Already up to date.`, não havia nada novo e o trabalho acaba aqui.

4. Atualizar as dependências

Sempre que o `package-lock.json` aparecer entre os arquivos alterados:

```
npm install
```

E, se a versão do Playwright tiver mudado:

```
npx playwright install chromium
```

Na dúvida, rodar os dois não causa problema: quando não há nada a fazer, eles terminam em segundos.

5. Subir o servidor de novo

```
npm run servidor -- --http --porta 4480 --host 127.0.0.1
```

No navegador, recarregue a página com `Ctrl + F5` — isso descarta a versão antiga da interface que o navegador guardou.

6. Conferir se a atualização funcionou

Três checagens rápidas, em ordem de custo:

```
npm run clientes
```

Todos os ambientes que você usa continuam na lista?

```
npm run validar-versao -- --ambiente guaratingueta-qa --sistema balcao --somente-versao
```

A ferramenta ainda alcança o ambiente e lê a versão?

```
npm run test:harness
```

Esta é a bateria de testes **da própria ferramenta** — ela não toca em ambiente de cliente. Se terminar tudo verde, a atualização está sadia.

7. Atualizar a versão do sistema a ser validado

Isso **não** se faz pela ferramenta: a publicação é feita pela infraestrutura, a pedido do time.

O fluxo é:

1. Vocês pedem a publicação da versão no ambiente.
2. Na aba *Executar*, passo 4, clique em **Detectar no ambiente**.
3. Confirme que a versão detectada é a que você pediu — e só então execute.

A versão muda várias vezes ao dia e sobe por módulo. Em execução longa, vale repetir a detecção ao final: se a versão mudou no meio, o resultado é de duas versões diferentes e precisa ser refeito.

8. Voltar para a versão anterior da ferramenta

Se algo passar a falhar depois de atualizar, dá para voltar. Primeiro veja o histórico:

```
git log --oneline -10
```

Depois volte para o ponto anterior, usando o código que aparece à esquerda:

```
git checkout <codigo-do-commit>
```

Para retornar à versão mais nova depois:

```
git checkout master
```

Nenhum desses comandos apaga resultados gravados nem o `.env`.

9. Problemas comuns

Sintoma	Causa provável	Solução
<code>git pull</code> reclama de alterações locais	Há arquivos modificados nesta máquina	Passo 2 (<code>git stash</code>)
A interface parece a antiga	O navegador guardou a página	<code>Ctrl + F5</code>
Erro de módulo não encontrado ao subir	Dependência nova não instalada	<code>npm install</code>
Erro do navegador ao executar casos	Playwright desatualizado	<code>npx playwright install chromium</code>
Ambiente sumiu da lista	A alteração local se perdeu	<code>git stash pop</code> , ou recriar o arquivo em <code>config/ambientes/</code>
Falha só depois da atualização	Regressão da ferramenta	Volte pelo passo 8 e registre o que quebrou