

Alicerce CMS — Guia do desenvolvedor

Versão 1.0 · 23/10/2026 · TECINFOSP

Visão geral

O Alicerce CMS é um CMS seguro da TECINFOSP, construído sobre o WebsiteBaker 2.14, que exige PHP 8.4 ou superior e não executa PHP vindo de conteúdo ou de temas. Ele se estende de duas formas: **plugins** (módulos em PHP, instalados pelo administrador e verificados pelo CodeGuard e pelo Guardião de extensões) e **templates** (temas em Twig, renderizados num sandbox que não permite executar código).

Tipo de extensão	Pasta	Linguagem	Para quê
Módulo de página (page)	modules/<nome>	PHP	Um tipo de seção nas páginas: galeria, catálogo, formulário
Ferramenta de admin (tool)	modules/<nome>	PHP	Uma tela em Admin > Ferramentas, como a de Fontes do Google
Snippet (snippet)	modules/<nome>	PHP	Funções carregadas no site para templates e outros módulos
Template (tema)	templates/<nome>	Twig + CSS	O visual do site
Seção Code e Droplets	dentro das páginas	Twig	Trechos dinâmicos escritos pelo editor do site

O núcleo técnico fica em `framework/`: `src\Template\SafeTwig` (motor Twig em sandbox), `src\Template\GoogleFonts`, `src\Security\PwHash` (senhas), `src\Security>LoginThrottle` (bloqueio de login), `src\Security\ExtensionGuard` (Guardião de extensões) e `src\Security\CodeGuard` (verificação de código enviado). As bibliotecas externas ficam em `vendor/`, gerenciadas pelo Composer.

Regras de segurança

Toda extensão precisa seguir estas regras; o que viola as cinco primeiras é bloqueado na instalação ou no salvamento.

- Sem execução dinâmica de código.** O CodeGuard recusa `eval()`, `call_user_func()`, funções variáveis (`{f() }`), `system()` e afins, inclusive nas formas com namespace (`\system()`).
- Templates não têm PHP.** O layout fica em `template.twig`; o `index.php` do tema só chama o motor. No sandbox não há `include`, `source`, `constant()` nem acesso a objetos.
- Formulários do admin usam FTAN.** Imprima `$admin->getFTAN()` dentro do `<form>` e confira `$admin->checkFTAN()` antes de gravar qualquer coisa.
- Permissão antes de agir.** Ferramentas e módulos conferem `$admin->get_permission('<nome>', 'module')`.
- Nenhum arquivo acessível direto.** Todo `.php` começa com a trava `if (!\defined('SYSTEM_RUN')) { ... 404 ... }`.
- Escape na saída, SQL protegido.** Use `htmlspecialchars()` em tudo que vem do usuário; em SQL, `(int)` para números e `$database->escapeString()` para textos.
- Senhas só por PwHash.** Use `\src\Security\PwHash::make()` e `::verify()`; nunca `md5()` ou comparação com `==`.
- Uploads pelo núcleo.** Arquivos enviados passam pelas rotinas de mídia, que verificam extensões e SVG ativos.

Criando um plugin (módulo)

Um plugin é uma pasta em `modules/` com um `info.php` que o descreve; o tipo (`$module_function`) define quais outros arquivos o núcleo procura.

Arquivo	Tipos	Quando é chamado
<code>info.php</code>	todos	Ao listar e instalar o módulo
<code>install.php</code>	todos	Uma vez, na instalação: cria tabelas e configurações
<code>upgrade.php</code>	todos	Em cada atualização; precisa poder rodar várias vezes
<code>uninstall.php</code>	todos	Na remoção: apaga tabelas e configurações
<code>add.php</code> / <code>delete.php</code>	page	Quando uma seção do módulo é criada ou removida numa página
<code>modify.php</code> / <code>save.php</code>	page	Editor da seção no admin e gravação
<code>view.php</code>	page	Exibição da seção no site
<code>tool.php</code>	tool	A tela em Admin > Ferramentas
<code>include.php</code>	snippet	Carregado no site a cada visita
<code>twig.php</code>	todos	Funções novas para os templates (ver abaixo)
<code>languages/EN.php</code> , <code>PT.php</code> , <code>ES.php</code>	todos	Textos da interface

O `info.php`

```
<?php
if (!defined('SYSTEM_RUN')) {\header($_SERVER['SERVER_PROTOCOL'],' 404 Not Found'); echo '404 Not Found';
exit;}

$module_directory      = 'depoimentos';
$module_name           = 'Depoimentos';
$module_function       = 'page';          // page | tool | snippet
$module_version        = '1.0.0';
$module_platform       = '2.14.0';       // versão mínima do Alicerce
$module_requires_php   = '8.4';          // opcional
$module_requires_extensions = 'gd';      // opcional, separadas por vírgula
$module_author         = 'TECINFOSP';
$module_license         = 'GNU General Public License';
$module_description     = 'Lista de depoimentos de clientes.'
```

Exemplo: módulo de página “Depoimentos”

1. `install.php` cria a tabela, usando sempre o prefixo do site:

```
$database->query('CREATE TABLE IF NOT EXISTS `'.TABLE_PREFIX.'mod_depoimentos` (
  `section_id` INT NOT NULL, `page_id` INT NOT NULL,
  `itens` MEDIUMTEXT NULL, PRIMARY KEY (`section_id`))');
```

2. `add.php` cria a linha da seção nova:

```
$database->query('INSERT INTO `'.TABLE_PREFIX.'mod_depoimentos` (`section_id`,`page_id`,`itens`) VALUES
('.(int)$section_id.','.(int)$page_id','\[\]\')');
```

3. `modify.php` mostra o formulário com `<?= $admin->getFTAN() ?>` e envia para `save.php`.
4. `save.php` confere `$admin->checkFTAN()`, valida os campos e grava com `escapeString()`.
5. `view.php` lê os itens e imprime com `htmlspecialchars()`; para HTML mais elaborado, use `\src\Template\SafeTwig::renderSafe($twig, ['itens' => $aItens])`.
6. `delete.php` e `uninstall.php` removem a linha e a tabela.

Ferramenta de admin (`tool`)

A tela recebe `$admin` e as constantes do sistema. O formulário deve enviar `tool=<nome>` num campo oculto, porque o núcleo identifica a ferramenta por esse parâmetro também no POST. O módulo `modules/google_fonts` é um exemplo completo: FTAN, permissão, três idiomas e gravação em `settings`.

Instalação e distribuição

Compacte a pasta em ZIP e instale em Admin > Complementos > Módulos. O CodeGuard e o Guardião de extensões verificam o ZIP antes de extrair. Para que um módulo venha instalado de fábrica, inclua o nome dele em `install/ModuleWhiteList`.

Plugins que oferecem funções aos templates

Um plugin pode criar funções para os templates, Droplets e seções Code com um arquivo `twig.php`. O núcleo carrega esse arquivo de todo módulo instalado, no site e no painel, antes de montar o Twig.

```
<?php
if (!defined('SYSTEM_RUN')) {\header($_SERVER['SERVER_PROTOCOL'].' 404 Not Found'); echo '404 Not Found';
exit;}

use src\template\SafeTwig;

// nome: minúsculas, números e _ ; o terceiro argumento true marca o retorno como HTML pronto
SafeTwig::registerFunction('saudacao', static fn(string $sNome = ''): string
=> '<p class="saudacao">Olá, '.$htmlspecialchars($sNome).'!</p>', true);
```

No template: `{{ saudacao(user.display_name) }}`. Regras:

- O nome não pode repetir uma função do núcleo; se repetir, o registro é recusado.
- Funções que devolvem HTML (`true`) precisam escapar tudo o que vem de fora com `htmlspecialchars()`.
- Receba só textos, números e listas (`array`); o sandbox não entrega objetos.
- O que precisa rodar a cada visita (registrar acesso, por exemplo) vai no `include.php` de um módulo `snippet`. O `twig.php` só cria funções.
- O CodeGuard recusa funções guardadas em variável e chamadas assim, como `$h = fn(...)` seguido de `$h(...)`. Chame a função pelo nome.

Os três plugins de exemplo

Plugin	Tipo	Uso no Twig	Dados guardados
Contador de visitantes (<code>visitor_counter</code>)	snippet	<pre>{{ visitor_counter({show: ['today', 'total']}) }}, {{ visitor_count('month') }}</pre>	Só totais por dia; nenhum IP, cookie novo ou dado pessoal
Quem está online (<code>who_online</code>)	snippet	<pre>{{ who_online({minutes: 5, names: false}) }}, {{ online_count() }}</pre>	Um código anônimo da sessão e o horário, apagados após 15 minutos
Contagem regressiva (<code>countdown</code>)	page	<pre>{{ countdown('2026-12-31 23:59', {title: 'Falta pouco', done: 'Chegou!'}) }}</pre> ou como seção de página	Data, título e mensagem final de cada seção

Visitas de robôs de busca não são contadas. O `countdown` mostra como juntar seção de página e função de template: as duas usam a mesma `countdown_html()` em `functions.php`.

Links com acentos

O endereço de cada página é gerado do título sem perder letras: “Sobre nós” vira `sobre-nos` e “Ação de Verão” vira `acao-de-verao`. Em plugins que criam links, use `\bin\helpers\PreCheck::sanitizeFilename($titulo)`.

Plugins de terceiros e o Guardião de extensões

Todo plugin ou tema passa por três barreiras: o CodeGuard e o Guardião verificam o pacote antes de instalar, e o Guardião pausa durante o uso o plugin que falhar.

Barreira	Quando	O que faz	Resultado para o administrador
CodeGuard	Antes de extrair o ZIP	Recusa <code>eval</code> , comandos de sistema, chamadas dinâmicas, extensões perigosas e <code>.htaccess</code> perigoso	Instalação bloqueada, com o arquivo e o motivo
Guardião: requisitos	Antes de extrair	Confere versão do Alicerce, versão do PHP e extensões do PHP do <code>info.php</code>	Bloqueio com o que falta
Guardião: temas	Antes de extrair	Aceita só Twig; valida cada <code>.twig</code> no sandbox	Bloqueio com o arquivo e a linha
Guardião: risco	Antes de extrair	Aponta rede, gravação de arquivos, e-mail, cookies, scripts externos; bloqueia código codificado e alteração de tabelas do núcleo	Avisos junto da mensagem de instalado; bloqueio para risco alto
Guardião: assinatura	Antes de extrair	Confere <code>signature.json</code> com as chaves confiáveis	“assinado por TECINFOSP”, “sem assinatura” ou bloqueio se o pacote foi alterado
Guardião: execução	Durante o uso	Captura erros de <code>include.php</code> , <code>twig.php</code> e <code>view.php</code> ; após 3 erros, ou um erro fatal, pausa o plugin	Aviso vermelho no topo do painel com o plugin e o motivo; o site continua no ar

O painel fica em Admin > Ferramentas > Guardião de extensões: plugins pausados e o motivo, reativação, verificação das extensões instaladas e regras.

Assinatura de pacotes

1. Uma vez só: `php tools/alicerce-sign.php keygen tecinfosp.key`. Guarde o `.key` em local seguro. A chave pública impressa vai em Guardiã de extensões > Chaves de assinatura, no formato `TECINFOSP: <chave>`.
2. Para cada versão aprovada: `php tools/alicerce-sign.php sign <pasta do plugin> tecinfosp.key` e depois compacte a pasta.
3. Qualquer alteração depois de assinar faz o pacote ser bloqueado. A regra “aceitar só pacotes assinados” fecha a porta para o resto.

Como avaliar um plugin enviado por terceiros

1. **Recebimento:** código-fonte completo, `info.php` com versão e requisitos, descrição do que faz e de quais dados guarda, licença compatível com a GPL.
2. **Triagem automática:** instale num site de teste limpo; o CodeGuard e o Guardiã recusam o proibido e listam os riscos.
3. **Revisão de código:** cada aviso do Guardiã precisa de justificativa; confira FTAN, permissões, escape e SQL.
4. **LGPD:** quais dados pessoais coleta, por quanto tempo guarda e se apaga tudo ao desinstalar.
5. **Teste funcional:** instalar, usar, atualizar e desinstalar, no PHP 8.4 e 8.5, nos três idiomas, com log de erros limpo.
6. **Aprovação:** assinatura pela TECINFOSP, registro da versão e do desenvolvedor, publicação.
7. **Após publicar:** cada nova versão passa de novo pelos passos 2 a 6.

Biblioteca de mídias e galerias

A Biblioteca de mídias organiza imagens já enviadas pelo gerenciador de mídia em álbuns, e qualquer tema mostra esses álbuns como galeria sem código próprio.

1. Envie as imagens em Mídia, numa pasta (por exemplo `/media/ferias`).
2. Em Admin > Ferramentas > Biblioteca de mídias, crie um álbum a partir da pasta. As miniaturas são geradas em `/media/biblioteca-cache` (WebP ou JPEG).
3. Edite título, texto alternativo, legenda e ordem de cada imagem. “Adicionar imagens novas da pasta” traz o que foi enviado depois.
4. Mostre o álbum:

Onde	Como
Numa página	Seção Galeria : álbum, formato (grade, mosaico ou carrossel), 2 a 6 colunas, legendas, ampliar ao clicar
No tema ou numa seção Code	<pre>{{ gallery('ferias-de-verao', {layout: 'mosaic', columns: 4}) }}</pre>
No texto de qualquer página	<pre>[[Galeria?album=ferias-de-verao&layout=carousel&colunas=3]]</pre>
Layout próprio do tema	<pre>{% for foto in media_album('ferias-de-verao') %}{% endfor %}</pre>

Cada imagem de `media_album()` traz `url`, `thumb` (640 px), `large` (1600 px), `title`, `alt`, `caption`, `width` e `height`; `media_albums()` lista `slug`, `title` e `items`.

Por que funciona em qualquer tema: classes próprias (`.alc-gallery`) sem Bootstrap ou jQuery; cor e fonte herdadas do tema; visualizador com o elemento `<dialog>` do navegador (setas, Esc, foco e rótulos para leitores de tela); ajustes por variáveis CSS (`--alc-gap`, `--alc-radius`, `--alc-caption`); 2 colunas no celular. Testado com os 5 temas do Alicerce e com o DefaultTemplate antigo.

Criando um template (tema)

Um template é uma pasta em `templates/` cujo layout fica inteiro em `template.twig`. O jeito mais rápido de

começar é copiar um dos cinco temas inclusos (`alicerce_fundacao`, `alicerce_vitrine`, `alicerce_clinica`, `alicerce_sabor`, `alicerce_atlas`) e renomear.

```
templates/meu_tema/
  info.php      dados do tema, menus e blocos
  index.php     carregador (sem layout)
  template.twig todo o HTML
  css/style.css estilos
  editor.css    estilos dentro do editor de texto
  preview.png   miniatura em Admin > Complementos
```

`index.php` (o mesmo em todo tema)

```
<?php
if (!\defined('SYSTEM_RUN')) {\header($_SERVER['SERVER_PROTOCOL'],' 404 Not Found'); echo '404 Not Found';
exit;}

echo \src\Template\SafeTwig::renderFile(__DIR__.'/template.twig', [
    'tpl' => ['name' => 'Meu Tema', 'year' => (int) \date('Y')],
]);
```

`template.twig` **mínimo**

```
<!DOCTYPE html>
<html lang="{{ site.language }}">
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>{{ page.title }} | {{ site.title }}</title>
  <link rel="stylesheet" href="{{ template_url('css/style.css') }}">
  {{ frontend_head() }}
</head>
<body>
  <header>
    <a href="{{ site.url }}">{{ site.title }}</a>
    {{ menu(menu: 1, levels: 2, expand: 'all', class: 'menu') }}
  </header>
  <main>
    <h1>{{ page.title }}</h1>
    {{ page_content(1) }}
    {% set lateral = page_content(2)|trim %}
    {% if lateral %}<aside>{{ lateral|raw }}</aside>{% endif %}
  </main>
  <footer>{{ site_footer() }}</footer>
  {{ frontend_body() }}
</body>
</html>
```

Chame `page_content(n)` uma vez por bloco; `frontend_head()` e `frontend_body()` carregam o CSS e o JS dos módulos. Um erro de template aparece numa caixa vermelha só para administradores.

Referência do Twig

Função	Retorna	Observação
<code>menu({...})</code>	HTML do menu	<code>menu</code> , <code>start</code> ('root', 'current' ou nível), <code>levels</code> , <code>expand</code> ('trim', 'all', 'current', 'crumb'), <code>class</code> , <code>item_open</code> , <code>item_close</code> , <code>menu_open</code> , <code>menu_close</code>
<code>page_content(n)</code>	HTML do bloco n	Só em templates
<code>page_title()</code> , <code>page_description()</code> , <code>page_keywords()</code>	texto	Página atual
<code>page_root('page_title')</code>	texto	Página principal do ramo atual
<code>page_url(id)</code>	URL	Sem <code>id</code> , a página atual
<code>media_url('arquivo')</code> , <code>template_url('arquivo')</code>	URL	<code>..</code> é removido
<code>gallery('album', {...})</code> , <code>media_album('album')</code> , <code>media_albums()</code>	HTML / listas	Biblioteca de mídias (plugin)
<code>frontend_head()</code> , <code>frontend_body()</code>	tags	CSS e JS dos módulos
<code>site_footer()</code> , <code>search_url()</code>	HTML / URL	Rodapé e busca (<code>site.search</code>)
<code>google_fonts()</code>	tags	Fontes do Google configuradas
<code>trans('CHAVE')</code>	texto	Textos do sistema no idioma atual
<code>range</code> , <code>cycle</code> , <code>date</code> , <code>max</code> , <code>min</code> , <code>random</code>	—	Padrão do Twig

Variáveis: `site.title`, `site.url`, `site.language`, `site.search`, `page.id`, `page.title`, `page.menu_title`, `page.description`, `user.logged_in`, `user.display_name`, `now`, `section.id` (seções Code), `args` e parâmetros (Droplets), `tpl.*` (temas).

Tags: `if`, `for`, `set`, `apply`, `with`, `macro`, `import`, `verbatim`. **Filtros:** `abs`, `batch`, `capitalize`, `date`, `default`, `e/escape`, `first`, `format`, `join`, `json_encode`, `keys`, `last`, `length`, `lower`, `merge`, `nl2br`, `number_format`, `raw`, `replace`, `reverse`, `round`, `slice`, `sort`, `split`, `striptags`, `title`, `trim`, `upper`, `url_encode`.

Recursos da plataforma

Droplets em Twig

Droplets são trechos `[[Nome?param=valor]]` no conteúdo. O código de um Droplet é Twig; os parâmetros chegam como variáveis e em `args`. Exemplos inclusos: `[[year]]`, `[[SearchBox?msg=Procure]]`, `[[ShowRootParent]]`, `[[Lorem?blocks=3]]`, `[[Visitantes]]`, `[[Online]]`, `[[Contagem?data=2026-12-31 23:59&titulo=Falta pouco]]`.

Seções Code em Twig

O módulo Code não executa mais PHP: o conteúdo é Twig, validado ao salvar. Na atualização de sites antigos, seções e Droplets simples são convertidos; os demais são desativados e o código original fica guardado para consulta.

Fontes do Google

Em Admin > Ferramentas > Fontes do Google, cole até 5 links de `fonts.google.com`. **Cópia local** (padrão, recomendada pela LGPD) ou **direto do Google**. As tags entram sozinhas antes de `</head>`.

Segurança de login

Senhas com `bcrypt` (contas MD5 antigas são convertidas no primeiro login), sessão renovada no login e bloqueio de 15 minutos após 5 tentativas erradas por usuário ou 20 por endereço.

Idiomas

Instalador em português, inglês e espanhol, com detecção do navegador; painel em 22 idiomas. Em plugins, crie `languages/EN.php` completo e os demais só com o que traduzir.

Empacotamento e publicação

- **Versões:** `MAIOR.MENOR.CORREÇÃO` ; mantenha `upgrade.php` capaz de atualizar qualquer versão anterior.
- **Plataforma:** `$module_platform / $template_platform` = versão mínima testada (hoje 2.14.0).
- **Licença:** extensões que usam o núcleo entram no regime GPL; você pode vender, e o comprador recebe o código-fonte. Imagens e fontes podem ter licença própria em arquivos separados.
- **Nomes:** pasta e `*_directory` em minúsculas, sem espaços nem acentos.

Checklist antes de publicar

- ☐ Lint sem erros no PHP 8.4 e 8.5
- ☐ Todo `.php` com a trava `SYSTEM_RUN`
- ☐ Nenhum `eval`, `call_user_func`, função variável ou comando de sistema
- ☐ Formulários do admin com FTAN e verificação de permissão
- ☐ Saída escapada e SQL com `(int)` ou `escapeString()`
- ☐ Textos em `languages/` (EN, PT, ES)
- ☐ Instalar, atualizar e desinstalar testados num site limpo
- ☐ Template validado nos temas claro e escuro, no celular e no computador
- ☐ `preview.png` e descrição no `info.php`
- ☐ Pacote assinado pela TECINFOSP