E.Cluster

Documentações sobre o Cluster E.tecnolopgia

- GlitchTip + Sentry no Laravel

GlitchTip + Sentry no Laravel

GlitchTip + Sentry no Laravel

Guia genérico de instalação e configuração do monitoramento de erros em **qualquer aplicação Laravel** usando o **GlitchTip** (alternativa open-source compatível com o Sentry) através do pacote oficial `sentry/sentry-laravel`.

A instância utilizada pela E.tecnologia é: **https://glitchtip.etecnologia.com.br/**

> O GlitchTip é totalmente compatível com o protocolo do Sentry, por isso usamos o pacote oficial do Sentry para Laravel. A única diferença é a URL do DSN, que aponta para o servidor do GlitchTip em vez de `sentry.io`.

> **Pré-requisitos:** Laravel 11+ e acesso à instância do GlitchTip para criar o projeto e obter o DSN. Os comandos abaixo assumem **Laravel Sail** (Docker); caso seu projeto não use Sail, basta rodar os comandos `composer` / `artisan` diretamente (sem o prefixo `vendor/bin/sail`).

Índice

---

1. Instalação do pacote

Instale o pacote oficial via Composer:

```
vendor/bin/sail composer require sentry/sentry-laravel
```

> Use a versão estável mais recente do pacote (atualmente a linha `^4.x`). Confira a versão instalada no `composer.json` do seu projeto.

Em seguida, publique o arquivo de configuração (gera `config/sentry.php`):

```
vendor/bin/sail artisan sentry:publish --dsn=<SEU_DSN_DO_GLITCHTIP>
```

> O comando `sentry:publish` publica a config **e** já grava a variável `SENTRY_LARAVEL_DSN` no seu `.env`. Caso prefira configurar manualmente, pule o `--dsn` e edite o `.env` como descrito na seção 4.

---

2. Criação do projeto no GlitchTip

1. Acesse a sua instância do GlitchTip e faça login.
2. Selecione (ou crie) uma **Organização**.
3. No menu lateral, vá em **Projects** e clique em **Create a new project** (botão `+ Create Project`).
4. Em **Platform**, escolha **PHP → Laravel**.
5. Defina o **nome do projeto** (use o nome da sua aplicação) e a equipe responsável.
6. Clique em **Create Project**.
7. Na tela seguinte, o GlitchTip exibe o **DSN** do projeto. Copie-o — ele tem o formato:

   ```
   https://<CHAVE_PUBLICA>@<HOST_DO_GLITCHTIP>/<ID_DO_PROJETO>
   ```

   Exemplo usando a instância da E.tecnologia:

   ```
   https://<CHAVE_PUBLICA>@glitchtip.etecnologia.com.br/<ID_DO_PROJETO>
   ```

   > O DSN também pode ser recuperado depois em **Settings → Projects → (seu projeto) → Client Keys (DSN)**.

3. Configuração no Laravel

O arquivo `config/sentry.php` é o ponto central. Os valores principais são lidos a partir do `.env` (não edite o DSN diretamente no config — use o `.env`). Os campos relevantes:

Chave no config	Origem (.env)	Função
`dsn`	`SENTRY_LARAVEL_DSN`	Endereço do projeto no GlitchTip. **Obrigatório.**
`send_default_pii`	`SENTRY_SEND_DEFAULT_PII`	Envia dados do usuário/requisição (IP, headers, etc.).
`traces_sample_rate`	`SENTRY_TRACES_SAMPLE_RATE`	Taxa de amostragem de performance (`0.0` a `1.0`).
`environment`	`APP_ENV`	Separa eventos por ambiente (local, production...).

4. Variáveis de ambiente

Adicione ao `.env` (exemplo de desenvolvimento):

```
SENTRY_LARAVEL_DSN=https://<CHAVE_PUBLICA>@<HOST_DO_GLITCHTIP>/<ID_DO_PROJETO>
SENTRY_SEND_DEFAULT_PII=true
SENTRY_TRACES_SAMPLE_RATE=1.0
```

E no ambiente de produção (`.env.production` ou as variáveis do servidor), preencha com o DSN **do projeto de produção** (mantenha as chaves presentes no arquivo):

```
SENTRY_LARAVEL_DSN=
SENTRY_SEND_DEFAULT_PII=
SENTRY_TRACES_SAMPLE_RATE=
```

> **Produção:** use `SENTRY_TRACES_SAMPLE_RATE` mais baixo (ex.: `0.2`) para reduzir o volume de eventos de performance, e avalie `SENTRY_SEND_DEFAULT_PII=false` caso não queira enviar dados pessoais (IP, e-mail) ao servidor de erros.

Após alterar o `.env`, limpe o cache de configuração:

```
vendor/bin/sail artisan config:clear
```

---

5. Integração no código

Configure os três pontos de integração a seguir para uma cobertura completa:

5.1. Captura de exceções (`bootstrap/app.php`)

O handler de exceções do Laravel 11 é conectado ao Sentry/GlitchTip:

```
use Sentry\Laravel\Integration;

->withExceptions(function (Exceptions $exceptions) {
    Integration::handles($exceptions);
})->create();
```

5.2. Identificação do usuário (`app/Providers/AppServiceProvider.php`)

Anexa o usuário autenticado a cada evento enviado, facilitando rastrear quem encontrou o erro:

```
use Illuminate\Auth\Events\Authenticated;
use Illuminate\Support\Facades\Event;
use Sentry\Laravel\Integration;
use Sentry\State\Scope;

// Dentro de boot():
Event::listen(function (Authenticated $event) {
    Integration::configureScope(function (Scope $scope) use ($event): void {
        $scope->setUser([
            'id'    => $event->user->getAuthIdentifier(),
            'name'  => $event->user->name ?? null,
            'email' => $event->user->email ?? null,
        ]);
    });
});
```

5.3. Rota de teste (`routes/web.php`)

Uma rota dispara uma exceção proposital para validar a integração:

```
Route::get('/teste01', function () {
    throw new Exception('My first GlitchTip error!');
});
```

> Remova ou proteja a rota `/teste01` antes de ir para produção.

---

6. Testando a integração

1. Suba a aplicação (com Sail):

   ```
   vendor/bin/sail up -d
   ```

2. Acesse a rota de teste no navegador:

   ```
   http://localhost/teste01
   ```

3. A página deve estourar a exceção `My first GlitchTip error!`.
4. Abra sua instância do GlitchTip → seu projeto → **Issues**. O erro deve aparecer em alguns segundos, com a mensagem, o stack trace e (se logado) os dados do usuário autenticado.

Alternativamente, via Tinker:

```
vendor/bin/sail artisan tinker
>>> \Sentry\captureMessage('Teste manual de GlitchTip');
```

---

7. Criação do webhook (alertas)

O webhook permite que o GlitchTip notifique sistemas externos (Slack, Discord, Microsoft Teams, n8n, ou um endpoint próprio) quando um novo erro/issue ocorre.

7.1. Configurar no GlitchTip

1. Na sua instância do GlitchTip, abra o **projeto**.

2. Vá em **Settings → Alerts** (Regras de alerta do projeto).
3. Clique em **Create New Alert** (ou edite uma existente).
4. Configure as condições do disparo:
   - **Quantity / Timespan**: quantas ocorrências em quanto tempo disparam o alerta (ex.: `1` evento em `1` minuto).
5. Em **Alert Recipients**, adicione um recipiente do tipo **Webhook** (ou **General Webhook** / **Discord** / **Slack** conforme o destino).
6. Cole a **URL do webhook** de destino.
   - Para Slack/Discord/Teams: use a *Incoming Webhook URL* gerada na respectiva plataforma.
   - Para um endpoint próprio: informe a URL pública que receberá o `POST`.
7. Salve.

7.2. Formato do payload

O GlitchTip envia uma requisição **`POST`** com `Content-Type: application/json`. O corpo segue o formato compatível com os webhooks do Sentry/MS Teams, contendo os campos do alerta:

```
{
  "alias": "GlitchTip",
  "text": "GlitchTip Alert",
  "attachments": [
    {
      "title": "My first GlitchTip error!",
      "title_link": "https://<HOST_DO_GLITCHTIP>/<org>/issues/<id>",
      "text": "Exception em routes/web.php",
      "color": "#e52b50"
    }
  ],
  "sections": []
}
```

> O formato exato varia conforme o tipo de recipiente escolhido (Slack, Discord, Teams ou webhook genérico). Para integrações customizadas, registre o payload recebido uma vez e ajuste o parser ao formato real.

7.3. Receber o webhook na própria aplicação (opcional)

Caso queira tratar o alerta dentro do Laravel, crie uma rota pública (sem `auth`) para receber o `POST`:

```
// routes/web.php
Route::post('/webhooks/glitchtip', function (\Illuminate\Http\Request $request) {
    \Illuminate\Support\Facades\Log::info('GlitchTip alert', $request->all());

    return response()->json(['ok' => true]);
});
```

> Como o endpoint é público, valide a origem (ex.: um token secreto no path ou header) e garanta que a rota esteja **fora** dos middlewares de autenticação/permissão do seu projeto (ex.: `auth` e quaisquer middlewares de controle de acesso).

Referências

- Pacote: https://github.com/getsentry/sentry-laravel
- Documentação Sentry/Laravel: https://docs.sentry.io/platforms/php/guides/laravel/
- GlitchTip: https://glitchtip.com/documentation
- Instância interna: https://glitchtip.etecnologia.com.br/