

# Introdução ao Hyperf

Um guia básico para iniciantes

**Aluno: Reinan Gabriel Dos Santos Souza**  
Bacharelado em Sistemas de Informação  
Instituto Federal de Sergipe  
Campus Lagarto

## Reinan Gabriel Dos Santos Souza



Sou um entusiasta de tecnologia apaixonado por aprender e explorar novas áreas.

Desenvolvedor desde 2018  
Engenheiro DevOps Pleno – **MOVA**

<https://linktr.ee/reinanhs>



## Arquivos para a apresentação

Todos os materiais relacionados à apresentação do minicurso estão disponíveis digitalmente no meu repositório do **Gitlab**. Para acessar esses recursos, basta escanear o **QR Code** na imagem ao lado.



---

Gitlab: <https://gitlab.com/snct-lagarto-2023>

Github: <https://github.com/ReinanHS>

Nesta apresentação, exploraremos o PHP Hyperf, um framework de alto desempenho para aplicações web.

Vamos descobrir como essa tecnologia pode impulsionar:

- Performance dos nossos projetos.
- Otimizando o tempo de resposta.
- Otimizando o escalabilidade.

- Durante muito tempo, a linguagem de programação PHP foi uma escolha popular para o desenvolvimento de aplicações web.
- Sua popularidade se deve a vários fatores, incluindo:
  - Fácil aprendizagem.
  - Ampla documentação.
  - Vasta comunidade de desenvolvedores.
  - Ampla variedade de frameworks disponíveis

o PHP tinha um modelo de execução síncrona, que facilitava a escrita de código para aplicações web mais simples e com requisitos de processamento menos intensivos. Além disso, existiam ferramentas que facilitavam a sua utilização:

- WordPress
- Joomla
- Drupal
- Magento

## Exemplo:

publicacoes.ifs.edu.br/index.php/eventos

SisPubli  
Sistema de Publicação do IFS

INICIO IFS PROP

**Acesso ao Sistema**

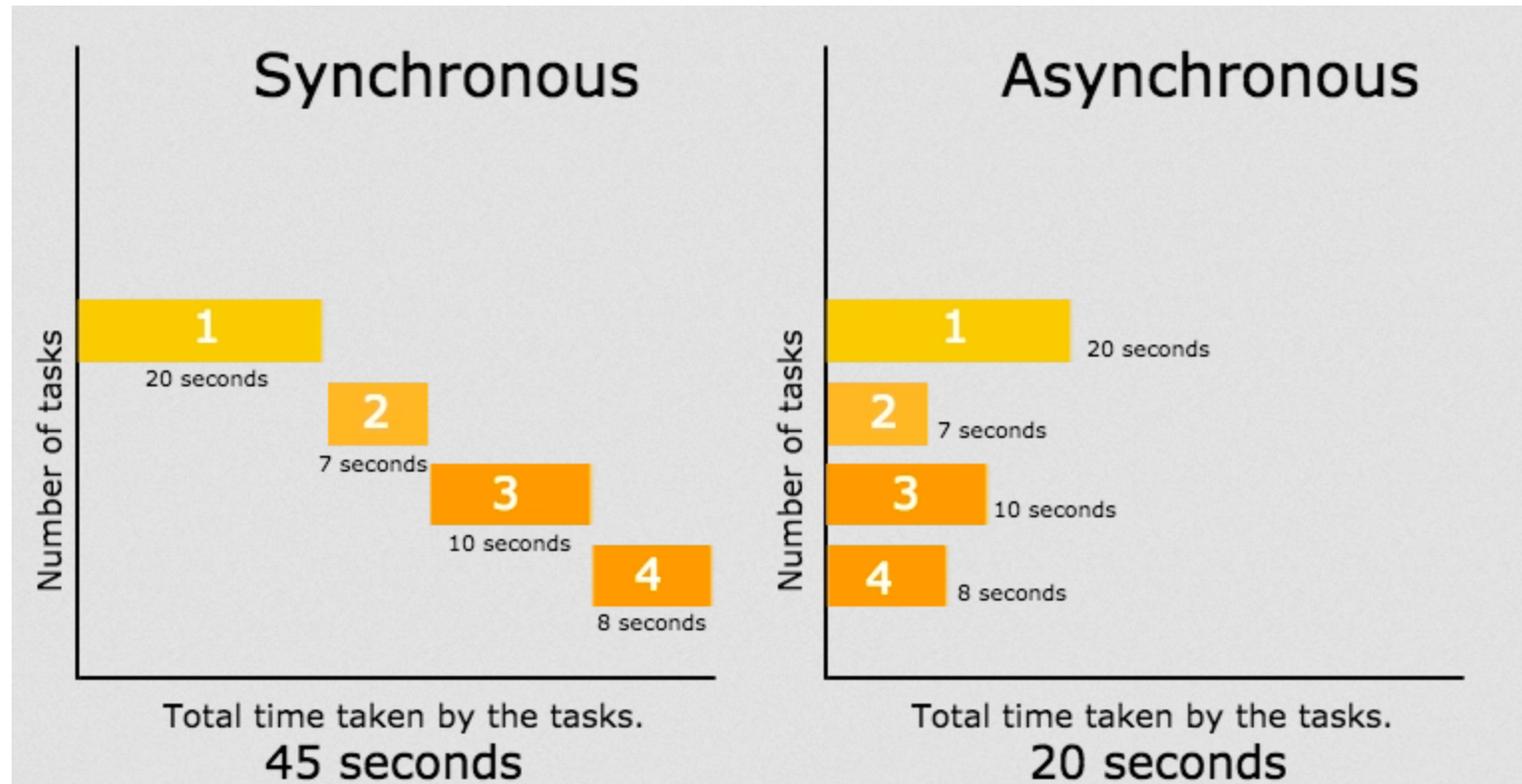
CADASTRO

Não seguro | view-source:publicacoes.ifs.edu.br/index.php/eventos

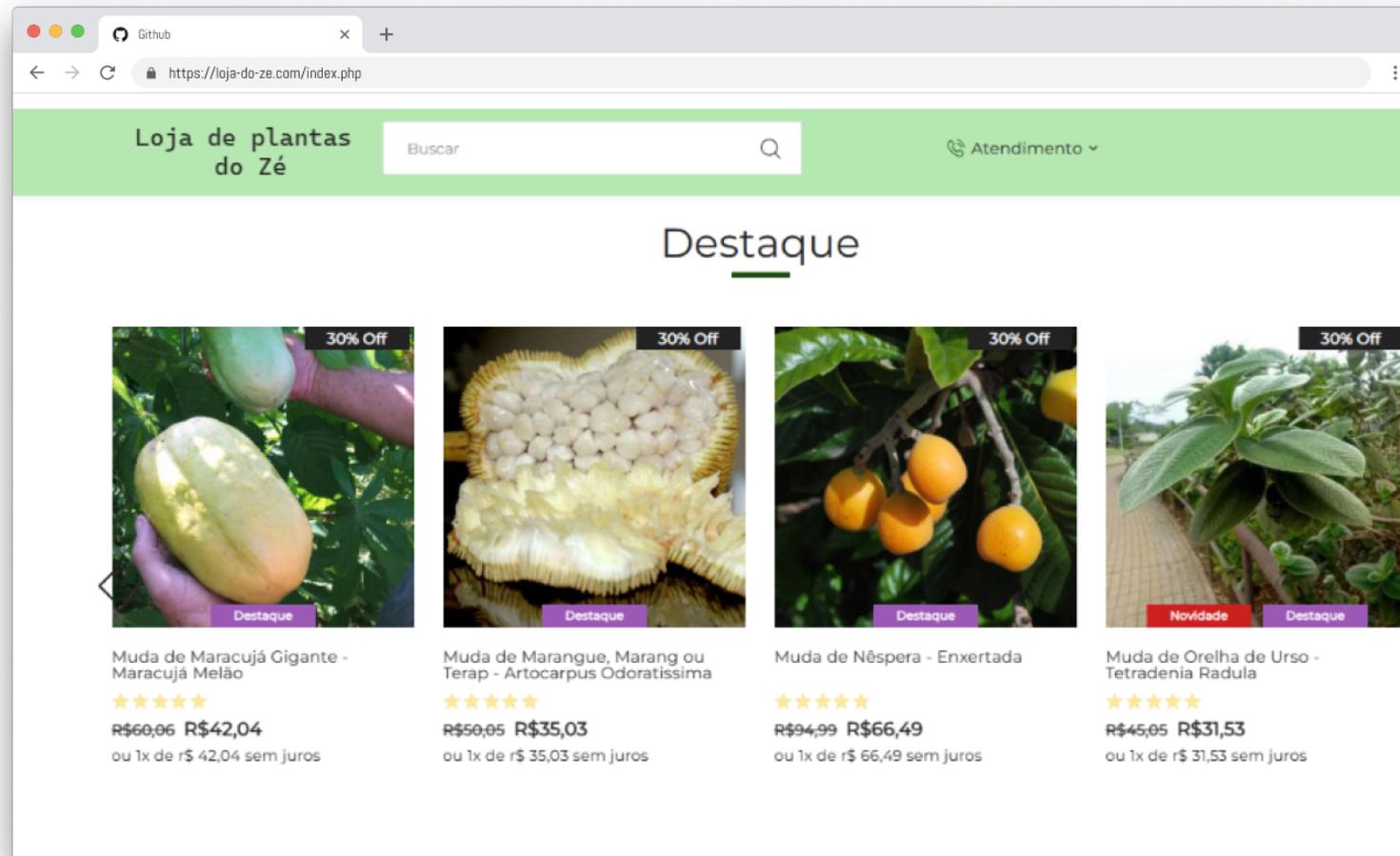
```
1 <!DOCTYPE html>
2 <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="pt-br" lang="pt-br" dir="ltr">
3 <head>
4   <base href="http://publicacoes.ifs.edu.br/index.php/eventos"/>
5   <meta http-equiv="content-type" content="text/html; charset=utf-8"/>
6   <meta name="description" content="Site de Publicações do IFS"/>
7   <meta name="generator" content="Joomla! - Open Source Content Management"/>
8   <title>Eventos - Publicações IFS</title>
9   <link href="http://publicacoes.ifs.edu.br/index.php/component/search/?Itemid=163&amp;
10  <link rel="stylesheet" href="/components/com_jevents/views/A.flat,,_assets,,_css,,_mo
11
12 <style type="text/css">div.mod_search110 input[type="search"]{width:auto}</style>
13 <script src="/media/com_wrapper_js/iframe-height.min.js+modules_mod_jevents_cal,
14 <script>eval(mod_pagespeed_y60uyzV8br);</script>
15 <script src="/media/jui/js/jquery.min.js.pagespeed.jm.iDyG3vc4gw.js" type="text/javas
16 <script src="/media_jui_js/jquery-noconflict.js+media_jui_js/jquery-migrate.min
17 <script>eval(mod_pagespeed_pi_txB_CqZ);</script>
```

No entanto, com a **evolução da tecnologia** e o aumento das demandas de desempenho e escalabilidade, o mercado está mudando para aplicações que trabalham de forma assíncrona.

A **programação assíncrona** permite que várias operações de entrada e saída sejam executadas simultaneamente, sem bloquear a execução do programa.



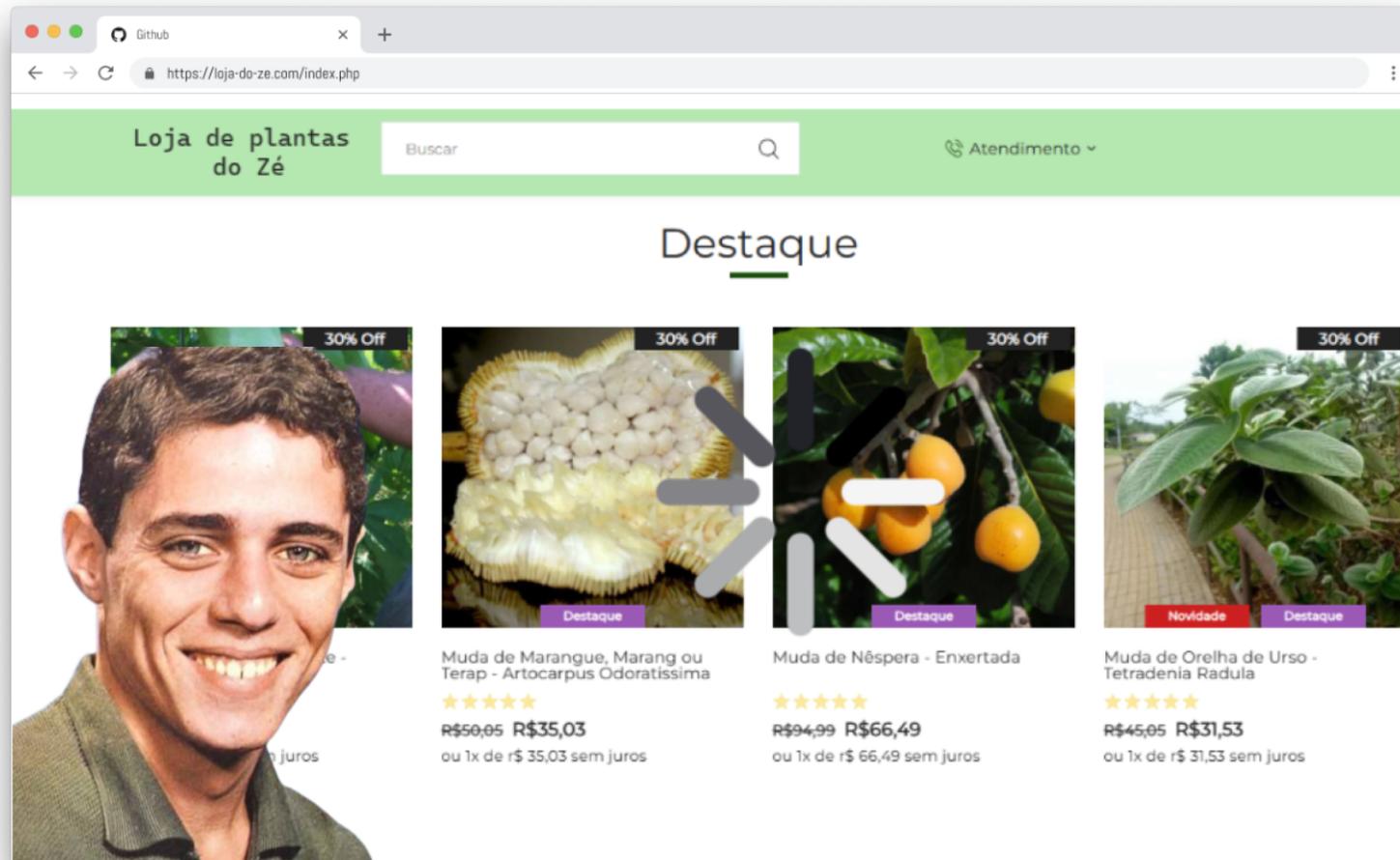
# Caso de estudo fictício: Loja que atende os moradores da cidade de Boquim



The screenshot shows a web browser window with the URL <https://loja-do-ze.com/index.php>. The website header is green and contains the logo 'Loja de plantas do Zé', a search bar with the text 'Buscar', and a 'Atendimento' dropdown menu. The main content area is titled 'Destaque' and features four plant listings, each with a '30% Off' badge. The first listing is for 'Muda de Maracujá Gigante - Maracujá Melão' with a 'Destaque' badge. The second is for 'Muda de Marangue, Marang ou Terap - Artocarpus Odoratissima' with a 'Destaque' badge. The third is for 'Muda de Nêspera - Enxertada' with a 'Destaque' badge. The fourth is for 'Muda de Orelha de Urso - Tetradenia Radula' with a 'Novidade' badge and a 'Destaque' badge. Each listing includes a star rating and pricing information.

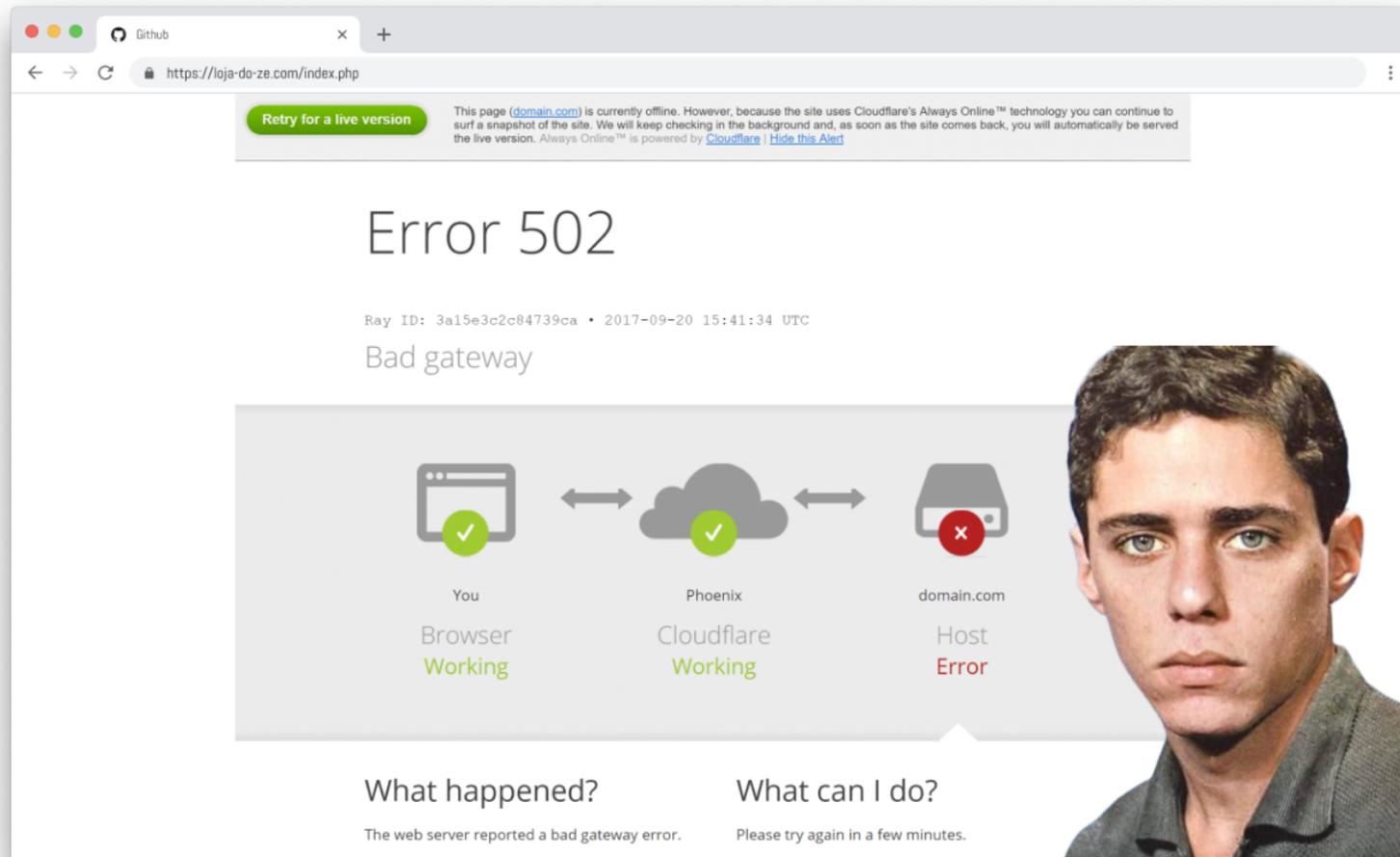
Produto	Desconto	Destaque	Classificação	Preço Original	Preço Atual	Condição de Pagamento
Muda de Maracujá Gigante - Maracujá Melão	30% Off	Destaque	★★★★★	R\$60,06	R\$42,04	ou 1x de R\$ 42,04 sem juros
Muda de Marangue, Marang ou Terap - Artocarpus Odoratissima	30% Off	Destaque	★★★★★	R\$50,05	R\$35,03	ou 1x de R\$ 35,03 sem juros
Muda de Nêspera - Enxertada	30% Off	Destaque	★★★★★	R\$94,99	R\$66,49	ou 1x de R\$ 66,49 sem juros
Muda de Orelha de Urso - Tetradenia Radula	30% Off	Novidade Destaque	★★★★★	R\$45,05	R\$31,53	ou 1x de R\$ 31,53 sem juros

# Caso de estudo fictício: A loja vende para todos do estado de Sergipe



# Caso de estudo fictício: A loja vende para todo o Brasil

O site dele não aguentou o tráfego.



A transição para aplicações assíncronas é impulsionada por várias razões:

- O aumento do tráfego na web.
- O crescimento do número de usuários.

A programação assíncrona permite que as aplicações gerenciem eficientemente essas requisições, melhorando o desempenho e a escalabilidade.

As aplicações modernas frequentemente dependem de operações intensivas de entrada e saída:

- Chamadas a APIs externas.
- Acesso a bancos de dados.
- Integração com serviços de terceiros.

A programação assíncrona permite que essas operações sejam executadas de forma não bloqueante.

A linguagem de programação PHP foi amplamente utilizada no passado devido:

- Sua facilidade de uso e ampla variedade de frameworks disponíveis.
- Ao modelo de execução síncrona que se adequava bem a muitos casos de uso.

No entanto, com o aumento das demandas de desempenho, escalabilidade e interatividade em tempo real, o mercado está migrando para aplicações assíncronas.

## Como surgiu o Hyperf?

O framework **Hyperf** em 2018, destinado a trazer soluções para a programação assíncrona utilizando a linguagem de programação PHP.

Ao contrário de muitos outros frameworks disponíveis no mercado para PHP, o **Hyperf** foi concebido desde o início para se adaptar aos novos padrões exigidos pelo mercado para aplicações modernas.

Agora que temos uma compreensão do contexto que levou à criação do Hyperf, vamos explorar como o framework incorporou programação assíncrona em seu código.

- Como ele superou as limitações do PHP?

## O que é o Swoole?

O Swoole é uma biblioteca para rede assíncrona em PHP que proporciona um ambiente de execução de alto desempenho e suporte para programação assíncrona.

Diferentemente do modelo de execução síncrona tradicional do PHP, o Swoole permite que os desenvolvedores escrevam código assíncrono e aproveitem as vantagens da programação não bloqueante.

## Exemplo:

```
<?php

use function Swoole\Coroutine\run;
use function Swoole\Coroutine\go;

run(function() {
    go(function(){
        sleep(1);
        echo "Execução do meu primeiro processo";
    });

    go(function(){
        echo "Execução do meu segundo processo.";
    });
});
```

# O que é o Hyperf?

O Hyperf é um framework PHP de alto desempenho para o desenvolvimento de aplicações web, baseado no ecossistema Swoole.

Inspirado pelo **Laravel** e outros frameworks populares, o Hyperf foi desenvolvido com o objetivo de fornecer uma solução eficiente e escalável para a criação de aplicações modernas.

o Hyperf foi projetado desde o início para se destacar em:

- Performance
- Microsserviços
- Componentes desacoplados

Diferentemente de muitos outros frameworks no mercado, como:

- Laravel
- Symfony
- Slim
- CakePHP

o Hyperf não foi adaptado para utilizar o Swoole, mas sim concebido para trabalhar em conjunto com ele.

Todos os seus componentes foram desenvolvidos considerando o uso do Swoole, proporcionando uma integração harmoniosa entre as tecnologias.

Temos discutido a questão da performance e as distinções entre o Swoole e o PHP convencional. Agora, é hora de analisarmos os dados concretos!

Eu compilei alguns resultados de testes realizados em alguns dos principais frameworks em PHP. É importante salientar que esses testes não foram conduzidos por mim, mas foram extraídos de uma publicação específica:

- [Hyperf - PHP Coroutine Framework baseado em Swoole](#)

## Laravel 8 com PHP 8.0 (93 rq/s)

```
Running 10s test @ http://0.0.0.0:8000
12 threads and 400 connections
Thread Stats      Avg          Stdev         Max      +/- Stdev
  Latency    550.01ms    335.24ms     1.12s    63.01%
  Req/Sec    12.38        8.80       70.00    84.72%
938 requests in 10.10s, 16.64MB read
Socket errors: connect 155, read 1214, write 12, timeout 0
Requests/sec:    92.83
Transfer/sec:    1.65MB
```

## Lumen 8 com PHP 8.0 (279 rq/s)

```
Running 10s test @ http://0.0.0.0:8000
12 threads and 400 connections
Thread Stats      Avg      Stdev     Max    +/- Stdev
  Latency    108.93ms  41.13ms 193.70ms   76.31%
  Req/Sec    73.96    40.36   184.00    69.49%
2820 requests in 10.10s, 751.82KB read
Socket errors: connect 157, read 3987, write 4, timeout 0
Requests/sec:    279.10
Transfer/sec:    74.41KB
```

## Laravel Octane com PHP 8.0 e Swoole (1.273 rq/s)

```
Running 10s test @ http://0.0.0.0:8000
12 threads and 400 connections
Thread Stats      Avg          Stdev         Max    +/- Stdev
  Latency    110.06ms    190.13ms    1.65s    84.78%
  Req/Sec    118.57      96.64     550.00    69.97%
12796 requests in 10.05s, 13.09MB read
Socket errors: connect 157, read 101, write 4, timeout 0
Requests/sec:    1273.75
Transfer/sec:      1.30MB
```

## Hyperf 2.2 com PHP 8.0 (95.515 rq/s)

```
Running 10s test @ http://0.0.0.0:9502
12 threads and 400 connections
Thread Stats      Avg      Stdev     Max    +/-  Stdev
  Latency        2.55ms   1.37ms   21.38ms  90.08%
  Req/Sec       12.02k   1.77k    15.27k   58.88%
956407 requests in 10.01s, 173.30MB read
Socket errors: connect 779, read 95, write 0, timeout 0
Requests/sec:  95515.68
Transfer/sec:   17.31MB
```

# Comparações

Framework	rq/s
Laravel	92
Lumen	279
Laravel Octane	1.273
Hyperf	95.515

o Hyperf é mais de 986 vezes mais rápido do que o Laravel

Veja uma lista com as principais funcionalidades que esse framework oferece:

1. Alta performance
2. Arquitetura baseada em componentes
3. Suporte nativo para programação assíncrona
4. Ecossistema extensível
5. Suporte para contêineres de injeção de dependência

# Principais diferenciais do Hyperf em relação a outros frameworks

O Hyperf se diferencia de outros frameworks PHP devido:

- Suporte nativo à programação assíncrona.
- Foco em desempenho e escalabilidade.
- Arquitetura orientada a microsserviços.
- Ecossistema específico e flexibilidade personalizável.

Essas características tornam o Hyperf uma escolha atraente para o desenvolvimento de aplicações web que exigem alto desempenho e lidam com um grande volume de requisições assíncronas.

## Desafios e considerações

Embora o PHP Hyperf seja uma excelente escolha para aplicações de alta performance, é importante estar ciente dos desafios e considerações envolvidos. Alguns pontos a serem considerados incluem:

- Curva de aprendizado inicial
- Compatibilidade com bibliotecas existentes
- Manutenção e suporte da comunidade

# Quais empresas utilizam esse framework?

Atualmente, várias empresas adotam o uso do Hyperf em suas aplicações:

- PicPay
- MOVA SEP
- Linksoft
- MangaToon

## Requisitos e qualificações

- PHP 8+;
- Frameworks Laravel, Lumen, Swoole + HyperF;
- Armazenamento de dados em MySQL (ou outros bancos relacionais) ou Mongo/E
- Ambiente de desenvolvimento com Docker;
- Atenção para qualidade de código e design patterns;
- Desenvolvimento orientado a testes;
- Versionamento de códigos utilizando Git.

# A comunidade do Hyperf

A ativa participação da comunidade é evidenciada pela constante evolução do projeto, que se beneficia de:

- Contribuições regulares
- Correções de bugs
- Melhorias contínuas

A comunidade no GitHub do Hyperf oferece suporte e recursos valiosos para os desenvolvedores que desejam aproveitar ao máximo o potencial do framework.

# [QUESTION] Shared Table issue between Process and Controller in Hyperf

#6144

Edit New issue

 Closed ReinanHS opened this issue on Sep 12 · 3 comments



ReinanHS commented on Sep 12

Contributor ...

## Description

I am currently facing a challenge with sharing a table created during the initialization of the Hyperf server. My current approach involves creating this table when the server is started, and my process is responsible for managing the data in this table. When my application receives a request, it needs to query the information stored in this table.

我目前正面临一个关于在 Hyperf 服务器初始化期间共享表的挑战。我的当前方法涉及在服务器启动时创建这个表，而我的进程负责管理该表中的数据。当我的应用程序接收到请求时，需要查询存储在该表中的信息。

Here is an example of the approach I am currently using:

以下是我当前使用的方法示例：

```
<?php
namespace App\Process;

use Hyperf\Process\AbstractProcess;
use Hyperf\Process\Annotation\Process;
use Swoole\Table;

#[Process(name: "demo_process")]
class DemoProcess extends AbstractProcess
{
    public function handle(): void
    {
        $table = new Table(100);
        $table->column('token', Table::TYPE_STRING, 1024);
    }
}
```

Assignees

 limingxinleo

Labels

question

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

Notifications

Customize

 Unsubscribe

You're receiving notifications because you modified the open/close state.

3 participants



# [QUESTION] How to use trace together with coroutine in the Hyperf ecosystem #5871

Edit New issue

Closed ReinanHS opened this issue on Jun 23 · 2 comments



ReinanHS commented on Jun 23

Contributor

## Describe your question

My question is whether there is any recommendation on how to use trace together with coroutines. I have searched the official Hyperf documentation and various issues in GitHub, but I have not found any recommendation or in-depth discussion on this matter.

Before explaining my question, let me describe the scenario I have created to illustrate my concern. I have a sample project that makes some HTTP requests, and some of these requests are executed using coroutines. Here is an example of the code:

```
/**
 * Demonstration of a trace execution with coroutines
 * @return array
 * @throws GuzzleException
 */
public function index(): array
{
    $span = $this->startSpan('span #1');
    $this->client->get('https://gorest.co.in/public/v2/posts');

    /**
     * When executing the code inside the coroutine, the application loses the tracer.root context
     */
    co(function () {
        $spanTwo = $this->startSpan('span #2');

        $this->client->get('https://gitlab.com/api/v4/projects');

        $spanTwo->finish();
    });
}
```

Assignees

No one assigned

Labels

question

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

Notifications

Customize

Unsubscribe

You're receiving notifications because you modified the open/close state.

2 participants





# Comunidades no GitHub relacionados ao Hyperf



## OpenCodeCo

Colaboração gratuita de código aberto. O OpenCodeCo é um espaço livre, sem bloqueios, barreiras ou impedimentos para a colaboração de código aberto.

49 seguidores • Brasil • <https://opencodeco.dev> • [@opencodeco](https://twitter.com/opencodeco) • Empresa/OpenCodeco • [leo@opencodeco.dev](mailto:leo@opencodeco.dev)

[Deixar de seguir](#)

README.md

Oi 🙌

Nós somos o OpenCodeCo (mpany|llaborative|mmunity)

Sem complicações, sem barreiras, sem políticas absurdas, para colaboração de código aberto para todos!

*Sinta-se à vontade para se juntar a nós.*

Fixado

 **hiperf-traçador** Público

Bifurcado do [hiperf/tracer](#)

🔧 Substituição do componente Hyperf Tracer.

• .PHP ☆ 2

 **hiperf-métrica** Público

Bifurcado de [hiperf/metric](#)

🔧 Substituição do componente Hyperf Metric.

• .PHP

 **GoTask** Público

Bifurcado de [hiperf/gotask](#)

⚡ Um substituto de velocidade relâmpago para o Swoole TaskWorker em

 **Fastmongo** Público

Bifurcado de [Reasno/fastmongo](#)

🔧 Driver Plug & Play MongoDB para Hyperf (com suporte a Coroutines).

Principais discussões no mês passado

As discussões servem para compartilhar anúncios, criar conversas em sua comunidade, responder perguntas e muito mais.

[Iniciar uma nova discussão](#)

Povo



Principais idiomas

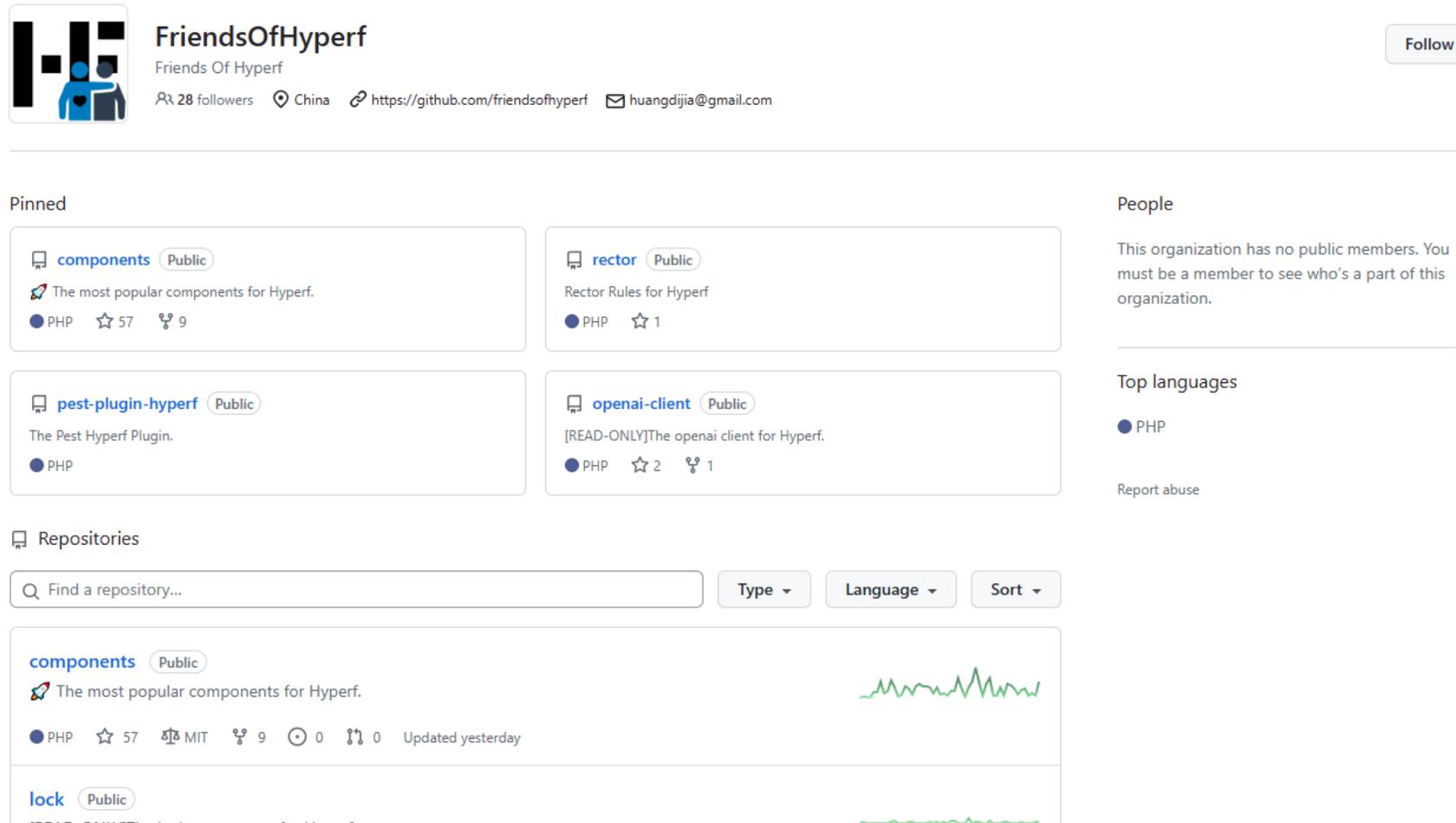
• .PHP • Concha • Ir • Makefile

• Dockerfile

Tópicos mais usados

[Hacktoberfest](#) [hiperf](#) [.php](#) [lã](#) [Cli](#)

# Comunidades no GitHub relacionados ao Hyperf



The screenshot shows the GitHub profile for the organization "FriendsOfHyperf". The profile includes a logo, the organization name, a bio "Friends Of Hyperf", 28 followers, location in China, a website link to the GitHub repository, and an email address. A "Follow" button is visible in the top right.

**Pinned**

- components** (Public): The most popular components for Hyperf. PHP, 57 stars, 9 forks.
- rector** (Public): Rector Rules for Hyperf. PHP, 1 star.
- pest-plugin-hyperf** (Public): The Pest Hyperf Plugin. PHP.
- openai-client** (Public): [READ-ONLY]The openai client for Hyperf. PHP, 2 stars, 1 fork.

**People**

This organization has no public members. You must be a member to see who's a part of this organization.

**Top languages**

- PHP

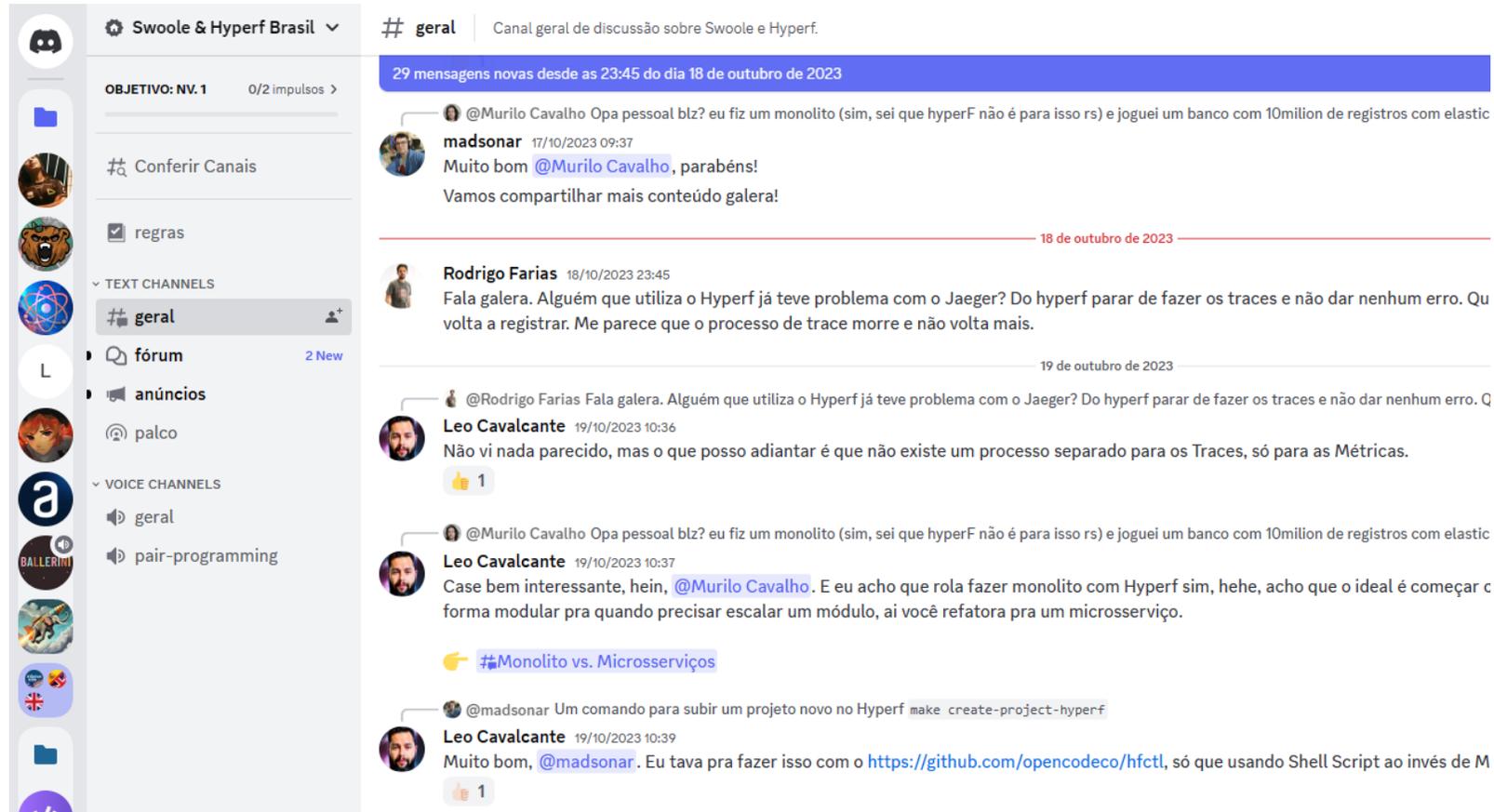
**Report abuse**

**Repositories**

Find a repository... Type Language Sort

- components** (Public): The most popular components for Hyperf. PHP, 57 stars, MIT license, 9 forks, 0 issues, 0 pull requests. Updated yesterday.
- lock** (Public): [READ-ONLY]The lock client for Hyperf.

# Comunidade Brasileira no Discord



The screenshot shows a Discord server interface for "Swoole & Hyperf Brasil". The server has a goal of "OBJETIVO: NV. 1" with "0/2 impulsos". The left sidebar lists channels: "Conferir Canais", "regras", "TEXT CHANNELS" (including "# geral", "fórum", "anúncios", "palco"), and "VOICE CHANNELS" (including "geral", "pair-programming"). The main chat area shows a history of messages in the "# geral" channel. A blue banner at the top of the chat indicates "29 mensagens novas desde as 23:45 do dia 18 de outubro de 2023".

Canal geral de discussão sobre Swoole e Hyperf.

29 mensagens novas desde as 23:45 do dia 18 de outubro de 2023

@Murilo Cavalho Opa pessoal blz? eu fiz um monolito (sim, sei que hyperF não é para isso rs) e joguei um banco com 10milion de registros com elastic

**madsonar** 17/10/2023 09:37  
Muito bom @Murilo Cavalho, parabéns!  
Vamos compartilhar mais conteúdo galera!

18 de outubro de 2023

**Rodrigo Farias** 18/10/2023 23:45  
Fala galera. Alguém que utiliza o Hyperf já teve problema com o Jaeger? Do hyperf parar de fazer os traces e não dar nenhum erro. Qu volta a registrar. Me parece que o processo de trace morre e não volta mais.

19 de outubro de 2023

@Rodrigo Farias Fala galera. Alguém que utiliza o Hyperf já teve problema com o Jaeger? Do hyperf parar de fazer os traces e não dar nenhum erro. Q

**Leo Cavalcante** 19/10/2023 10:36  
Não vi nada parecido, mas o que posso adiantar é que não existe um processo separado para os Traces, só para as Métricas.

1

@Murilo Cavalho Opa pessoal blz? eu fiz um monolito (sim, sei que hyperF não é para isso rs) e joguei um banco com 10milion de registros com elastic

**Leo Cavalcante** 19/10/2023 10:37  
Case bem interessante, hein, @Murilo Cavalho. E eu acho que rola fazer monolito com Hyperf sim, hehe, acho que o ideal é começar c forma modular pra quando precisar escalar um módulo, ai você refatora pra um microserviço.

← #Monolito vs. Microserviços

@madsonar Um comando para subir um projeto novo no Hyperf `make create-project-hyperf`

**Leo Cavalcante** 19/10/2023 10:39  
Muito bom, @madsonar. Eu tava pra fazer isso com o <https://github.com/opencodeco/hfctl>, só que usando Shell Script ao invés de M

1

# Apresentação prática sobre o Hyperf

Para começar, vamos acessar o repositório no **Gitlab** listado abaixo.



---

Fonte: <https://gitlab.com/snct-lagarto-2023/development-department/backend/poc-project-cloud-run-hyperf>

Chegamos ao fim desta apresentação do mini curso sobre Hyperf.

- Espero que vocês tenham compreendido a importância desse framework para impulsionar projetos web.
- Com sua *arquitetura sólida, recursos avançados e casos de sucesso comprovados*, o Hyperf se destaca como uma escolha poderosa para quem busca máxima performance.