

GABRIELE ROMANATO

Utilizzare le API di Supabase in Laravel per connettersi a un'istanza remota

Supabase è una piattaforma open-source che offre un backend completo simile a Firebase, con autenticazione, database PostgreSQL e storage. Laravel, con la sua potenza lato server, può integrarsi con Supabase per gestire dati remoti in modo efficiente. Questo articolo spiega come connettersi a un'istanza remota di Supabase utilizzando le API REST in un'applicazione Laravel.

Installazione delle dipendenze

Laravel utilizza il client HTTP di Guzzle per interagire con API esterne. Se non è già installato, eseguire:

```
composer require guzzlehttp/guzzle
```

Configurazione delle credenziali di Supabase

Nel file `.env`, aggiungere le credenziali API di Supabase:

```
SUPABASE_URL=https://your-project.supabase.co  
SUPABASE_KEY=your-secret-api-key
```

Creazione di un Service per la connessione a Supabase

Creare una nuova classe di servizio per interagire con l'API di Supabase:

```
php artisan make:service SupabaseService
```

Modificare il file `app/Services/SupabaseService.php`:

```
namespace App\Services;  
  
use GuzzleHttp\Client;  
  
class SupabaseService {  
    protected $client;  
    protected $url;
```

```

protected $key;

public function __construct() {
    $this->url = env('SUPABASE_URL');
    $this->key = env('SUPABASE_KEY');
    $this->client = new Client([
        'base_uri' => $this->url . '/rest/v1/',
        'headers' => [
            'apikey' => $this->key,
            'Authorization' => 'Bearer ' . $this->key,
            'Content-Type' => 'application/json'
        ]
    ]);
}

public function getData($table) {
    $response = $this->client->get($table);
    return json_decode($response->getBody(), true);
}

public function insertData($table, $data) {
    $response = $this->client->post($table, [
        'json' => $data
    ]);
    return json_decode($response->getBody(), true);
}
}

```

Utilizzo del Service in un Controller

Creare un controller per gestire le richieste API:

```
php artisan make:controller SupabaseController
```

Modificare il file `app/Http/Controllers/SupabaseController.php`:

```

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use App\Services\SupabaseService;

class SupabaseController extends Controller {
    protected $supabase;

    public function __construct(SupabaseService $supabase) {
        $this->supabase = $supabase;
    }
}

```

```
public function index() {  
    $data = $this->supabase->getData('your_table');  
    return response()->json($data);  
}  
  
public function store(Request $request) {  
    $data = $this->supabase->insertData('your_table', $request->all());  
    return response()->json($data);  
}  
}
```

Definizione delle Rotte

Nel file `routes/api.php`, aggiungere le rotte:

```
use App\Http\Controllers\SupabaseController;  
  
Route::get('/supabase', [SupabaseController::class, 'index']);  
Route::post('/supabase', [SupabaseController::class, 'store']);
```

Test delle API

Avviare il server Laravel:

```
php artisan serve
```

Testare la connessione con una richiesta GET:

```
curl -X GET http://127.0.0.1:8000/api/supabase
```

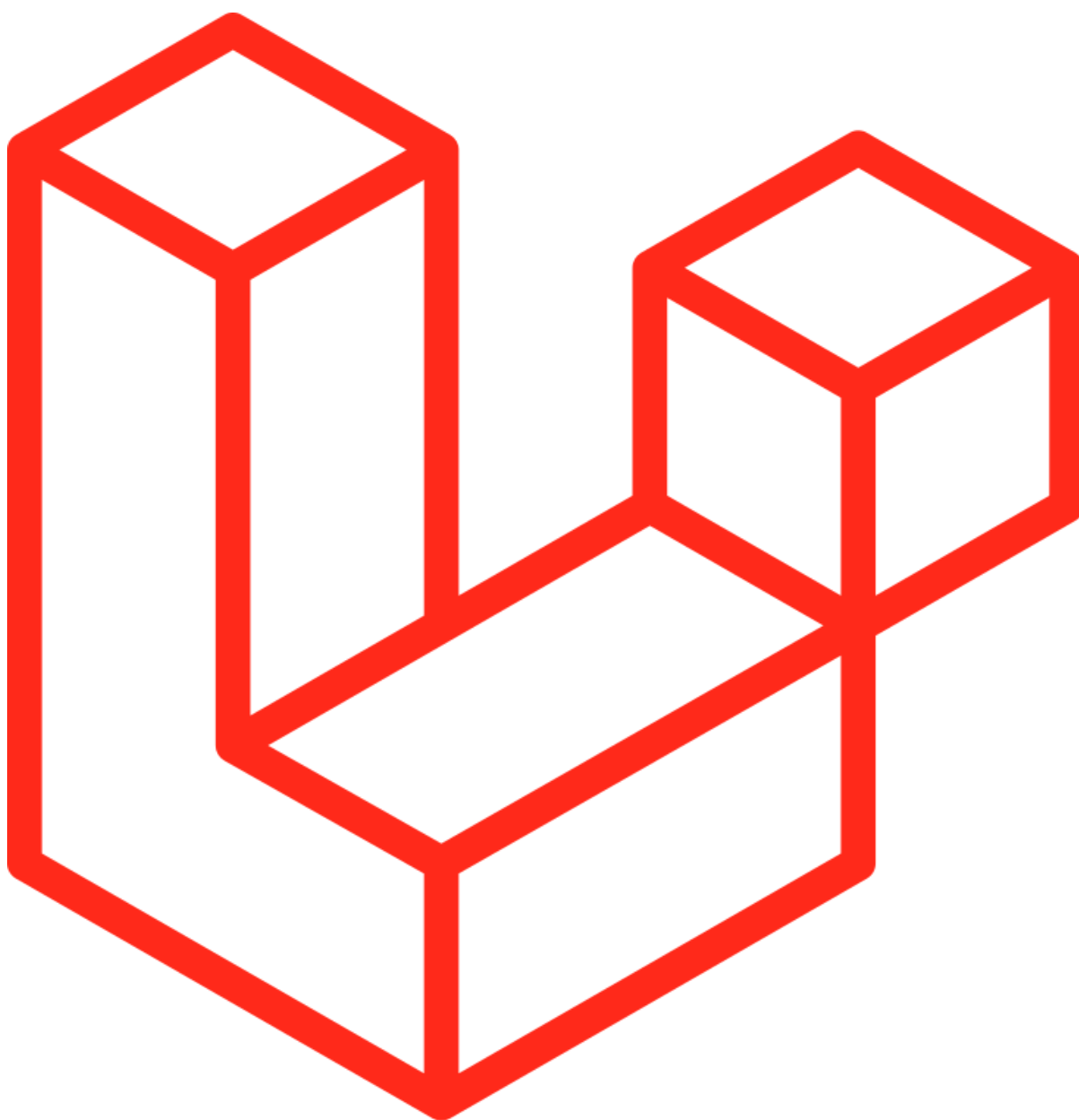
Oppure inviare dati con una richiesta POST:

```
curl -X POST http://127.0.0.1:8000/api/supabase \  
-H "Content-Type: application/json" \  
-d '{"column":"value"}'
```

Conclusione

Laravel può interagire facilmente con le API di Supabase grazie a Guzzle. Questo approccio consente di sfruttare un database remoto mantenendo un'architettura scalabile e sicura.

Applicazioni Correlate

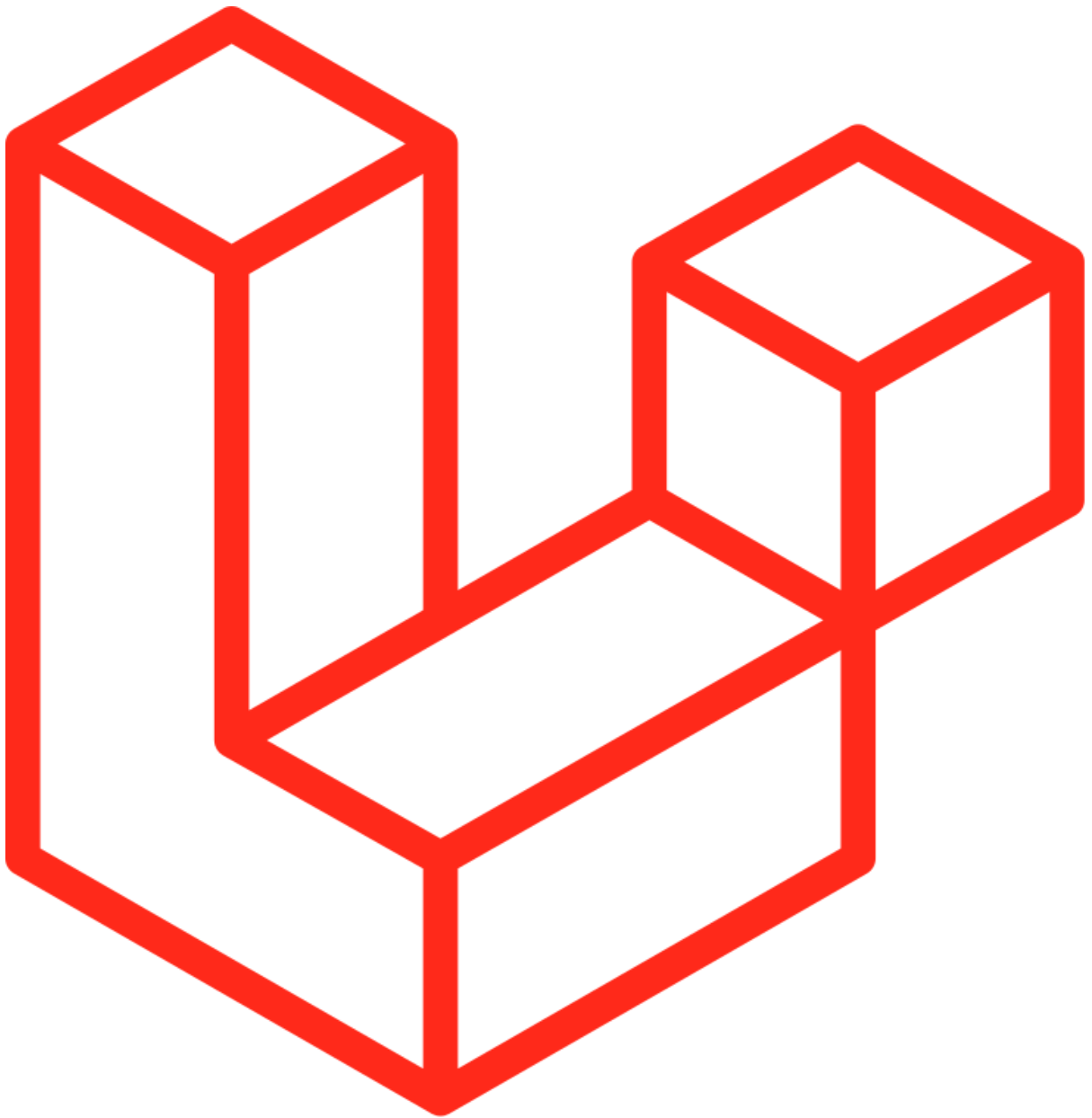


-

Laravel Placeholder Image

Applicazione in Laravel per creare immagini segnaposto.

[Docker](#)[Docker Compose](#)[Composer](#)[PHP](#)[Laravel](#)[JavaScript](#)

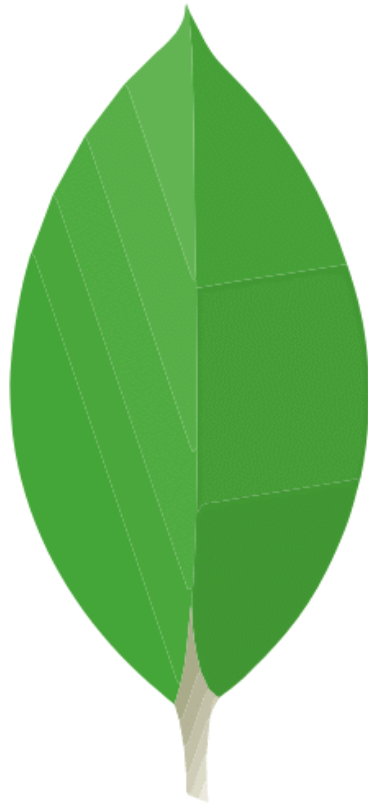


•

Laravel Shopping Cart

Applicazione che fa parte del progetto Laravel E-commerce e implementa la gestione del carrello in Laravel.

Docker Docker Compose Composer PHP Laravel



-

PHP MongoDB App

Applicazione basata su MongoDB con il driver PHP ufficiale.
DockerDocker ComposeComposerPHPMongoDB