

# GABRIELE ROMANATO

Menu

## Creare un server SOAP in PHP

Un server SOAP (Simple Object Access Protocol) consente di esporre servizi web che possono essere invocati da applicazioni client utilizzando un protocollo standard basato su XML. In questo articolo, esploreremo come creare un server SOAP in PHP, passo dopo passo.

Per prima cosa, dobbiamo assicurarci che l'estensione SOAP di PHP sia abilitata. Questo può essere fatto modificando il file `php.ini` o installando l'estensione tramite il gestore di pacchetti del sistema operativo.

Per abilitare SOAP in `php.ini`, cerca e decommenta (o aggiungi) la linea seguente:

```
extension=soap
```

Se stai usando un sistema basato su Debian/Ubuntu, puoi installare l'estensione SOAP usando il seguente comando:

```
sudo apt-get install php-soap
```

Il WSDL (Web Services Description Language) è un file XML che descrive i servizi offerti dal server SOAP. Definisce le operazioni disponibili, i parametri di input e output, e come accedere ai servizi.

Esempio di un semplice file `example.wsdl`:

```
<definitions name="ExampleService"
    targetNamespace="http://www.example.org/exampleservice"
    xmlns="http://schemas.xmlsoap.org/wsdl/"
    xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
    xmlns:tns="http://www.example.org/exampleservice"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">

    <message name="getHelloRequest">
        <part name="name" type="xsd:string"/>
    </message>
```

```

<message name="getHelloResponse">
  <part name="greeting" type="xsd:string"/>
</message>

<portType name="ExamplePortType">
  <operation name="getHello">
    <input message="tns:getHelloRequest"/>
    <output message="tns:getHelloResponse"/>
  </operation>
</portType>

<binding name="ExampleBinding" type="tns:ExamplePortType">
  <soap:binding style="rpc"
transport="http://schemas.xmlsoap.org/soap/http"/>
  <operation name="getHello">
    <soap:operation soapAction="getHello"/>
    <input>
      <soap:body use="literal"/>
    </input>
    <output>
      <soap:body use="literal"/>
    </output>
  </operation>
</binding>

<service name="ExampleService">
  <documentation>Example Service</documentation>
  <port name="ExamplePort" binding="tns:ExampleBinding">
    <soap:address location="http://localhost/soap-server.php"/>
  </port>
</service>
</definitions>

```

Creiamo un file PHP chiamato `soap-server.php` che implementa il server SOAP.

```

// Include il file WSDL
$wsdl = 'path/to/example.wsdl';

// Implementa la classe dei servizi
class ExampleService {
  public function getHello($name) {
    return "Hello, " . $name;
  }
}

// Crea un'istanza del server SOAP
$options = ['uri' => 'http://www.example.org/exampleservice'];
$server = new SoapServer($wsdl, $options);
$server->setClass('ExampleService');

```

```
// Gestisce le richieste SOAP
$server->handle();
```

Ora che il server SOAP è configurato, possiamo testarlo utilizzando un client SOAP. Creiamo un semplice client PHP per testare il nostro server.

```
// URL del server SOAP
$wsdl = 'http://localhost/path/to/example.wsdl';

// Crea un'istanza del client SOAP
$client = new SoapClient($wsdl);

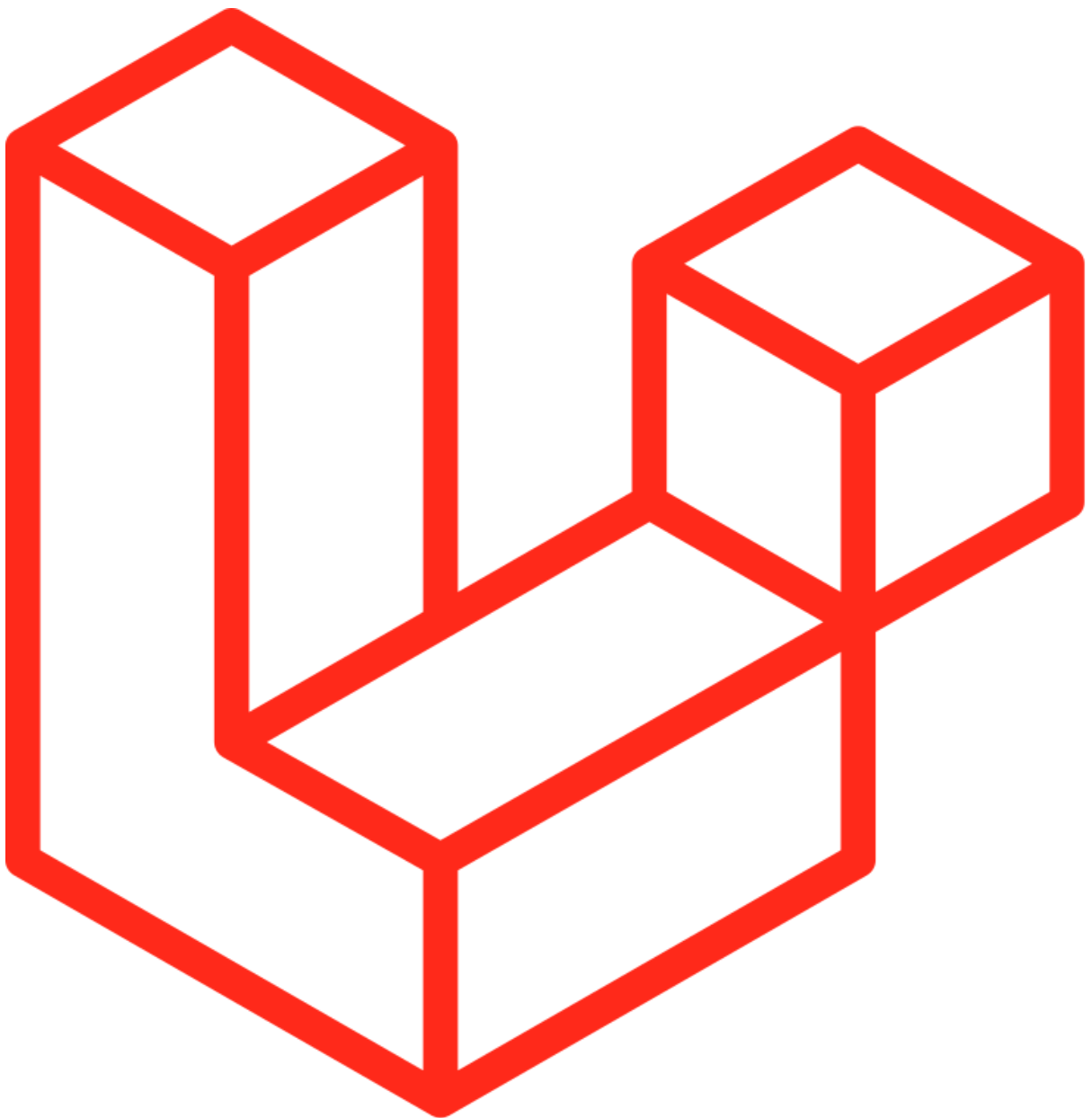
// Chiama il metodo getHello
$response = $client->getHello('World');

// Visualizza la risposta
echo $response;
```

## Conclusione

Abbiamo creato con successo un server SOAP in PHP. Questo server può essere esteso per includere più servizi e operazioni. Il WSDL funge da contratto tra il server e il client, garantendo che entrambi abbiano una comprensione comune dei servizi offerti.

## Applicazioni Correlate

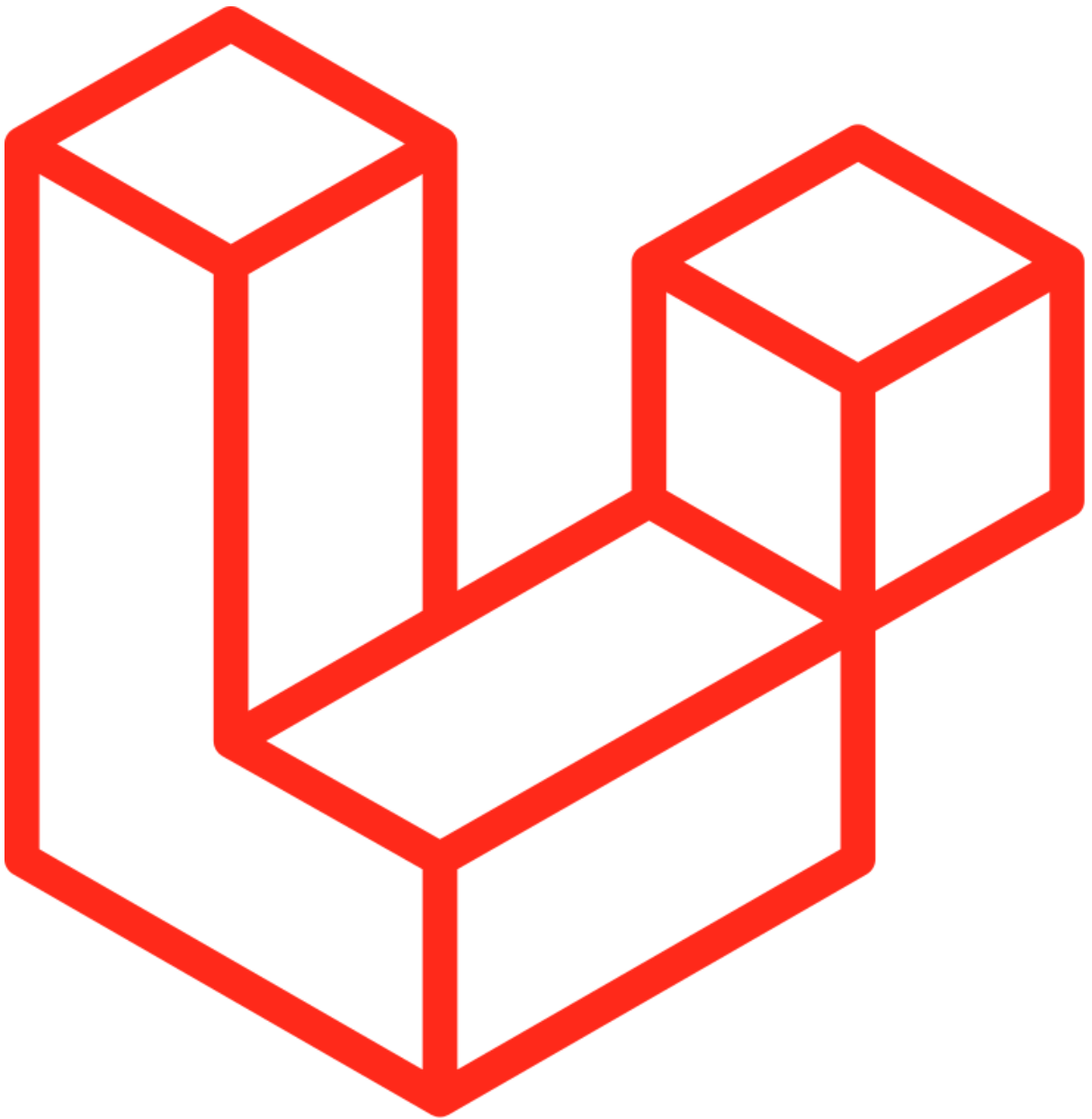


•

## Laravel Placeholder Image

Applicazione in Laravel per creare immagini segnaposto.

Docker Docker Compose Composer PHP Laravel JavaScript

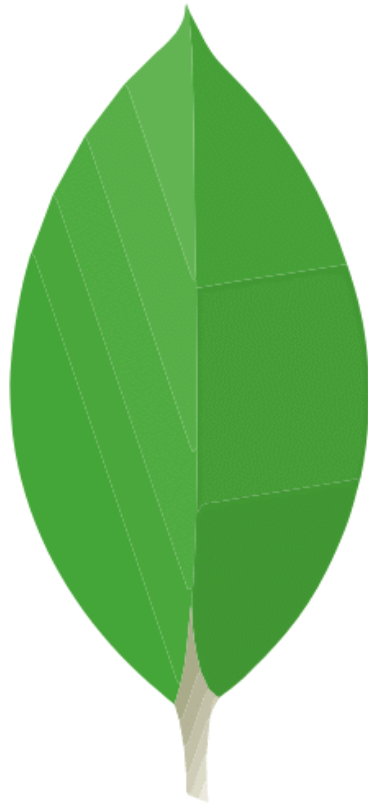


•

## **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