

# GABRIELE ROMANATO

## Come inviare email con PHP

Inviare email con PHP è un'operazione comune per molte applicazioni web, come l'invio di notifiche, conferme di registrazione o comunicazioni personalizzate. In questa guida, vedremo come inviare email utilizzando la funzione `mail()` di PHP e una libreria più avanzata come PHPMailer.

## Utilizzo della Funzione `mail()`

La funzione `mail()` è uno strumento base fornito da PHP per inviare email. Ecco un esempio:

```
<?php
$to = "destinatario@example.com";
$subject = "Oggetto del Messaggio";
$message = "Questo è il corpo dell'email.";
$headers = "From: mittente@example.com\r\n";

if (mail($to, $subject, $message, $headers)) {
    echo "Email inviata con successo.";
} else {
    echo "Errore nell'invio dell'email.";
}
?>
```

Nota che la funzione `mail()` richiede che il server abbia un servizio di posta configurato, come Sendmail su Linux.

## Utilizzo di PHPMailer

PHPMailer è una libreria più avanzata che offre maggiori funzionalità, come il supporto per SMTP e la gestione degli allegati. Segui questi passi per utilizzarla:

### Installazione

Puoi installare PHPMailer utilizzando Composer:

```
composer require phpmailer/phpmailer
```

### Esempio di Codice

Ecco un esempio di invio email con PHPMailer:

```
<?php
use PHPMailer\PHPMailer\PHPMailer;
use PHPMailer\PHPMailer\Exception;

require 'vendor/autoload.php';

$mail = new PHPMailer(true);

try {
    // Configurazione del server SMTP
    $mail->isSMTP();
    $mail->Host = 'smtp.example.com';
    $mail->SMTPAuth = true;
    $mail->Username = 'utente@example.com';
    $mail->Password = 'password';
    $mail->SMTPSecure = PHPMailer::ENCRYPTION_STARTTLS;
    $mail->Port = 587;

    // Impostazioni del mittente e destinatario
    $mail->setFrom('mittente@example.com', 'Mittente');
    $mail->addAddress('destinatario@example.com', 'Destinatario');

    // Contenuto dell'email
    $mail->isHTML(true);
    $mail->Subject = 'Oggetto del Messaggio';
    $mail->Body = '<p>Questo è il corpo del messaggio in formato HTML.
</p>';
    $mail->AltBody = 'Questo è il corpo del messaggio in formato plain
text.';

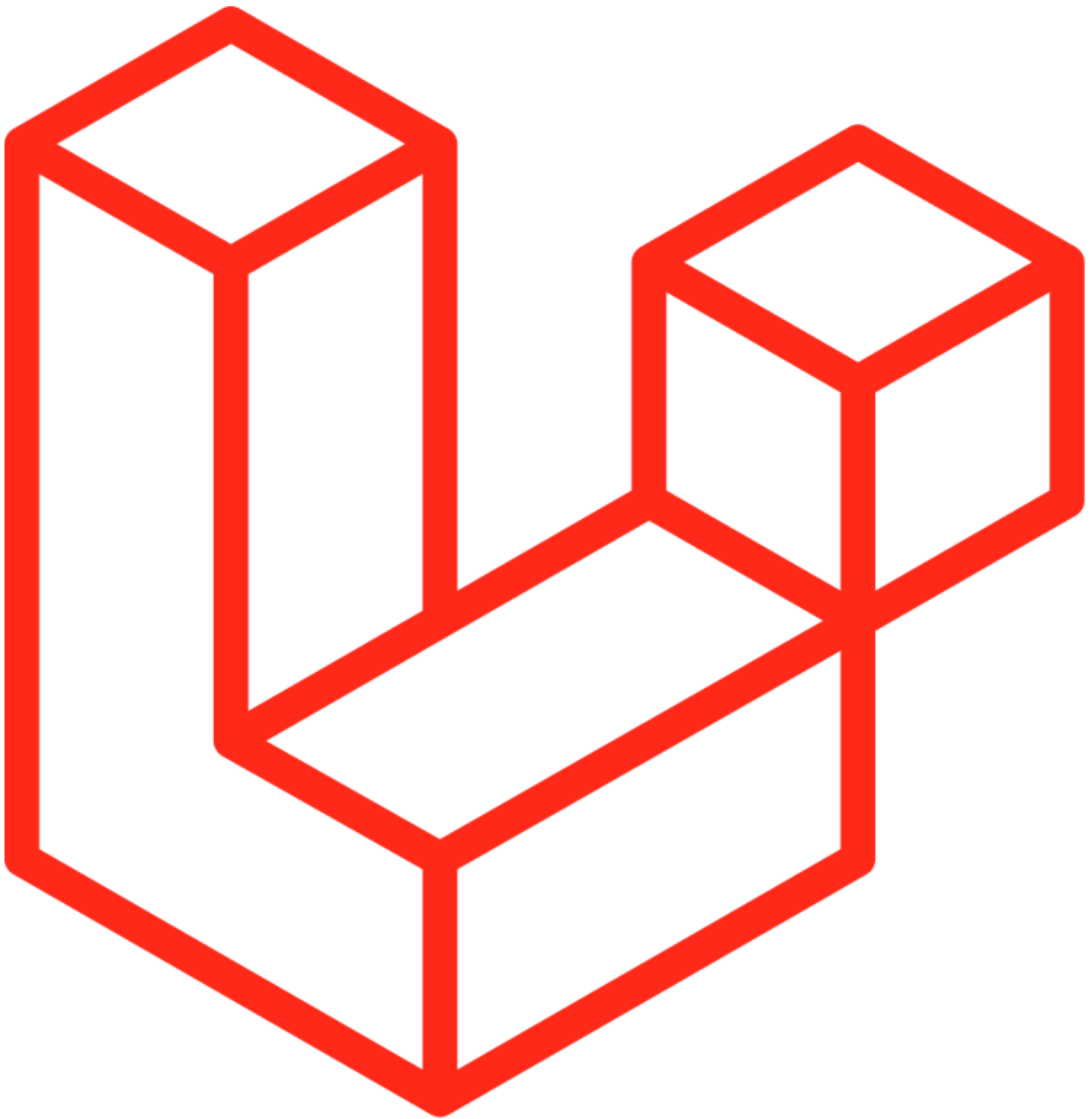
    $mail->send();
    echo 'Email inviata con successo';
} catch (Exception $e) {
    echo "Errore nell'invio dell'email: {$mail->ErrorInfo}";
}
?>
```

Questo approccio offre maggiore controllo e affidabilità rispetto alla funzione `mail()`.

## Conclusione

Inviare email con PHP può essere semplice o avanzato a seconda delle necessità. La funzione `mail()` è utile per scenari di base, mentre librerie come PHPMailer sono ideali per funzionalità più complesse. Scegli la soluzione più adatta al tuo progetto!

## Applicazioni Correlate

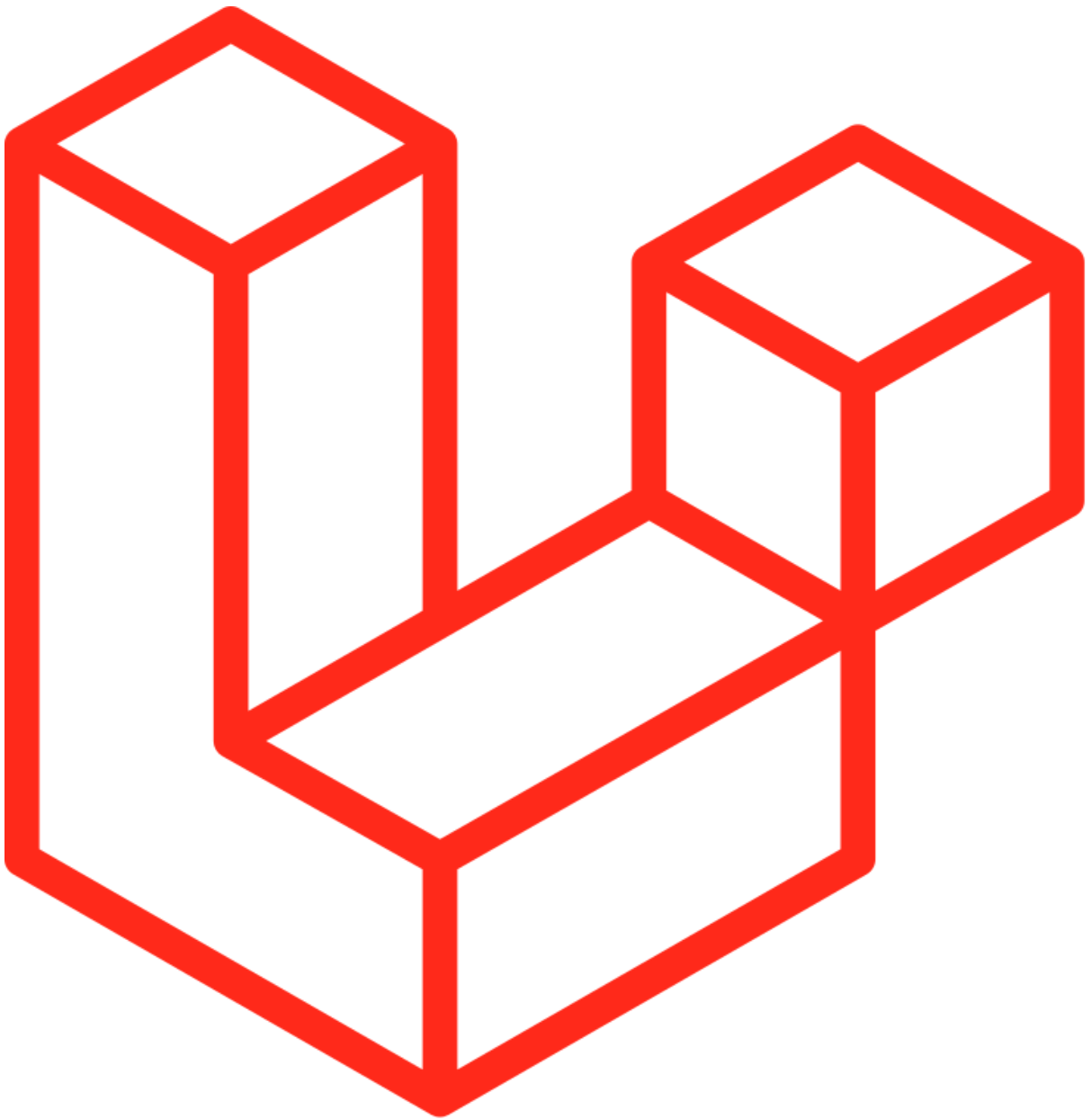


•

## Laravel Placeholder Image

Applicazione in Laravel per creare immagini segnaposto.

Docker Docker Compose Composer PHP Laravel JavaScript

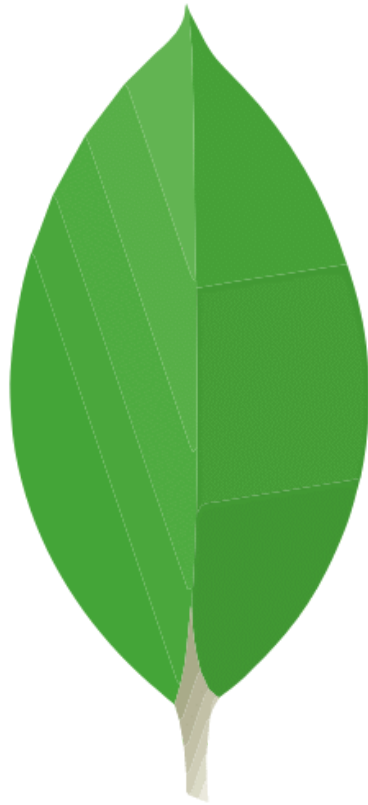


•

## **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