



- Utilisation de SwiftMailer pour l'envoi de mails
- Dans un projet php simple
- Avec Symfony

Utilisation de SwiftMailer

Plan

- Utilisation dans un projet simple
- Utilisation avec Symfony

SwiftMailer

Dans cette présentation nous allons voir comment utiliser **SwiftMailer**, un outil open source et facile d'utilisation, vous permettant d'intégrer à votre site web un serveur d'envoi de mail.

SwiftMailer dans un projet php simple

- Installation

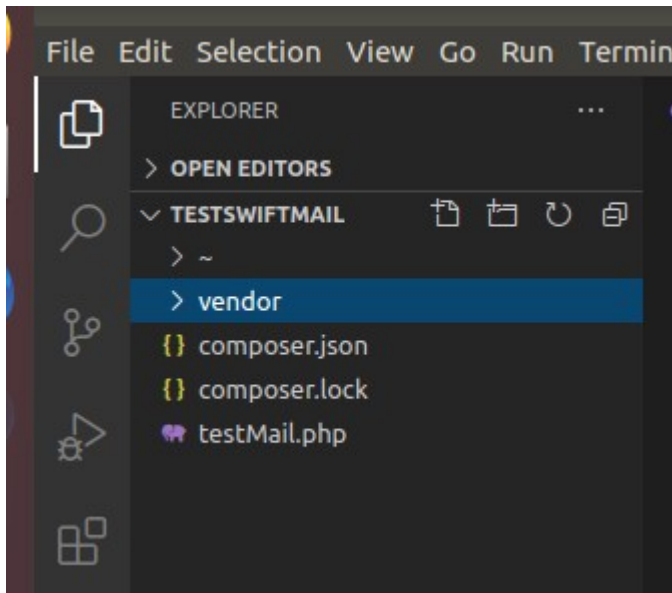
NB : composer est requis pour installer swiftMailer.

Positionnez-vous dans le dossier de votre projet et taper la commande suivante :

```
1 $ composer require "swiftmailer/swiftmailer:^6.0"
```

SwiftMailer dans un projet php simple

- Configuration



voici la structure de notre projet de test. Le fichier *testMail.php* contient le formulaire et le code php permettant d'envoyer le message venant du formulaire.

SwiftMailer dans un projet php simple

- Configuration

```
<body>
<h3 class="bg-info mt-3 text-center p-3 col-sm-8">Envoi de mail avec Swiftmail</h3>

<form class="form-horizontal" action="#" method="POST">
  <div class="form-group">
    <label class="control-label col-sm-2" for="email">Email:</label>
    <div class="col-sm-8">
      <input type="email" class="form-control" id="email" name="email" value="dabo.danfakha.sa@gmail.com">
    </div>
  </div>
  <div class="form-group">
    <label class="control-label col-sm-2" for="obj">Objet:</label>
    <div class="col-sm-8">
      <input type="text" class="form-control" id="obj" name="objet" value="Test Swiftmail">
    </div>
  </div>
  <div class="form-group">
    <label class="control-label col-sm-2" for="cnt">Contenu:</label>
    <div class="col-sm-8">
      <textarea name="mail_body" class="form-control" name="mail_body" id="cnt" cols="30" rows="5"></textarea>
    </div>
  </div>
  <div class="form-group">
    <div class="col-sm-offset-2 col-sm-8">
      <button type="submit" class="btn btn-success float-right px-5" name="envoi">Envoi</button>
    </div>
  </div>
</form>
```

le formulaire

SwiftMailer dans un projet php simple

- Configuration

```
<?php
if(isset($_POST['envoi'])) {
    require_once 'vendor/autoload.php';

    // Créer Transport
    $transport = (new Swift_SmtpTransport('smtp.gmail.com', 587, 'tls'))
        ->setUsername('nom_dutilisateur_admin@gmail.com')
        ->setPassword('votreMotDePasse')
    ;

    // Créer le Mailer en utilisant le Transport créé en haut
    $mailer = new Swift_Mailer($transport);

    // Créer le message
    $message = (new Swift_Message($_POST['objet']))
        ->setFrom(['nom_dutilisateur_admin@gmail.com' => 'username'])
        ->setTo($_POST['email']) //l'adresse du destinataire
        ->setBody($_POST['mail_body']) //le corp du message
    ;

    // Envoi du message
    $result = $mailer->send($message);

    //On teste si le message est bien envoyé ou pas
    echo $result===1?'Votre message a été envoyé.': 'Echec d envoi.';
}
?>
```

Ici nous avons utilisé le serveur smtp de gmail, c'est-à-dire tous nos messages seront envoyés à partir d'une adresse de gmail.

NB : Pour pouvoir envoyer des mails avec votre compte gmail il faudra se rendre dans vos paramètres de sécurité de votre compte Google et activer l'option « accès moins sécurisé des app... »

On peut aussi envoyer à plusieurs personnes en même temps car **setTo()** prend un tableau d'adresses mail en paramètre.

<https://myaccount.google.com/security>

SwiftMailer dans un projet php simple

- Test

Envoi de mail avec Swiftmail

Email:

Objet:

Contenu:

Envoi



Votre message a été envoyé.

Envoi de mail avec Swiftmail

Email:

Objet:

Contenu:

Envoi



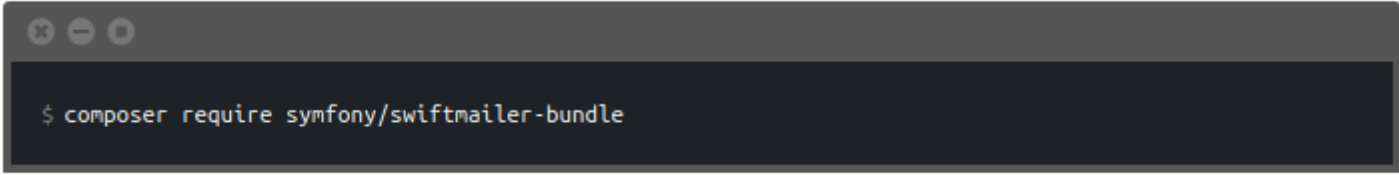
La boîte du destinataire

SwiftMailer avec symfony

- Installation

Symfony fournit une fonction de messagerie basée sur la bibliothèque Swift Mailer populaire via **SwiftMailerBundle**. Cet outil de messagerie prend en charge l'envoi de messages avec vos propres serveurs de messagerie ainsi que l'utilisation de fournisseurs de messagerie populaires tels que Gmail, Mandrill, SendGrid et Amazon SES.

On installe le bundle avec la commande :

A terminal window with a dark background and light gray text. The window has standard macOS window controls (close, zoom, and full screen buttons) in the top left corner. The command being executed is displayed in a monospaced font.

```
$ composer require symfony/swiftmailer-bundle
```

SwiftMailer avec symfony

- Configuration

toujours nous allons utiliser Gmail pour envoyer les mails. Pour cela, après l'installation, nous allons nous rendre dans le fichier `.env` et remplacer la ligne

```
MAILER_URL=null://localhost
```

par :

```
MAILER_URL=gmail://usernameAdmin@gmail.com:motDePassAdmin@localhost
```

SwiftMailer avec symfony

- Test

On peut directement tester sur la ligne de commandes en tapant *php bin/console swiftmailer:email:send*

```
root@al-amiine:/opt/lampp/htdocs/file_rouge_gp1# php bin/console swiftmailer:email:send

SwiftMailer's Interactive Email Sender
=====

From:
> laminedabo88@gmail.com

To:
> douvewane85@gmail.com

Subject:
> Test SwiftMailer

Body:
> Bonjour M. ceci est un message test de swiftMailer avec symfony

[OK] 1 emails were successfully sent.

root@al-amiine:/opt/lampp/htdocs/file_rouge_gp1#
```

Ln 45, Col 1 (69 selected) Spaces: 4 UTF-8 LF Plain Text

SwiftMailer avec symfony

- Intégration dans le code

Maintenant que tout est okey, on peut intégrer swiftMailer dans le code de notre projet.

Dans notre cas, il s'agit d'envoyer à chaque apprenant après son ajout dans une promo, ses informations de connexion notamment son *nom d'utilisateur* et son *mot de passe*.

SwiftMailer avec symfony

- Intégration dans le code

```
public function addPromo(Request $request, \Swift_Mailer $mailer, à compléter...){

    // Actions avant...

    foreach ($value['apprenants'] as $key => $val) {
        $Apprenant = new Apprenant();
        if ($Apprenant = $apprenant_repo -> findOneByEmail($val['email'])) {

            $message = (new \Swift_Message('Sélections Sonatel Academy'))
                ->setFrom('usernameAdmin@gmail.com')
                ->setTo($Apprenant -> getEmail())
                ->setBody('Bonjour cher(e) '. $Apprenant -> getPrenom() .' '. $Apprenant -> getNom() .'
                Félicitations!!! vous avez été sélectionné(e) suite à votre test d'entrée à la Sonatel Academy.
                Veuillez utiliser ces informations pour vous connecter à votre Promo. username: '. $Apprenant -> getUsername() .'
                password: <<CommonPassword>>. A bientôt.')
            ;

            $mailer->send($message);

        }

    }

    // Actions après...
```

l'apprenant une fois
connecté devra
obligatoirement
changer son mot de
passe.

Thanks to
SwiftMailer :)

MERCI