

## Resume Generator

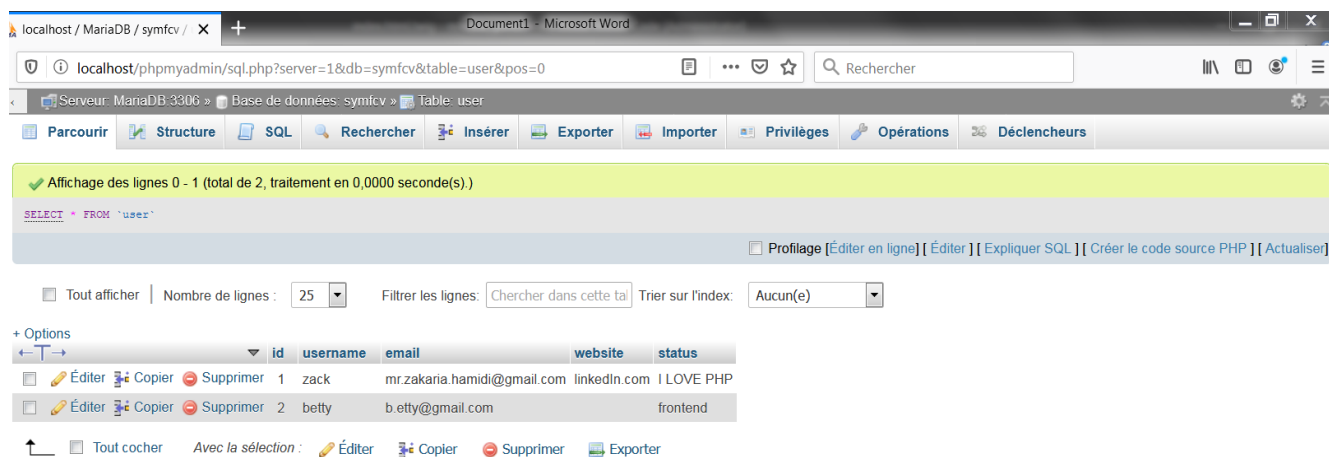
1. Create new skelaton

-> `composer create-project symfony/website-skeleton resume_generator`

2. I used server to execute (**php bin/console server:run**)

-> `composer require server --dev`

3. Created a controller
4. Generate user Entity
5. I made the migration to database with doctrine, I used *mariaDB* with *WampServer* where I added user for examples



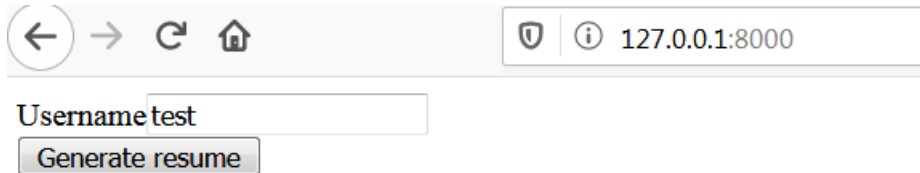
6. I created a method to generate the form and redirection with username as a POST request
7. I used twig
8. I used validator constraints as assert so any user can't generate the resume without typing at least 5 characters

```
private $username;

/**
 * @ORM\Column(type="string", length=255)
 * @Assert\NotBlank
 * @Assert\Length(min=5,minMessage = "At least 5 characters ")
 */
private $username;
```

9. I made a condition if the user exist then it will generate the resume otherwise it will show a message that he doesn't

10. Condition if web site exists by using «not empty»



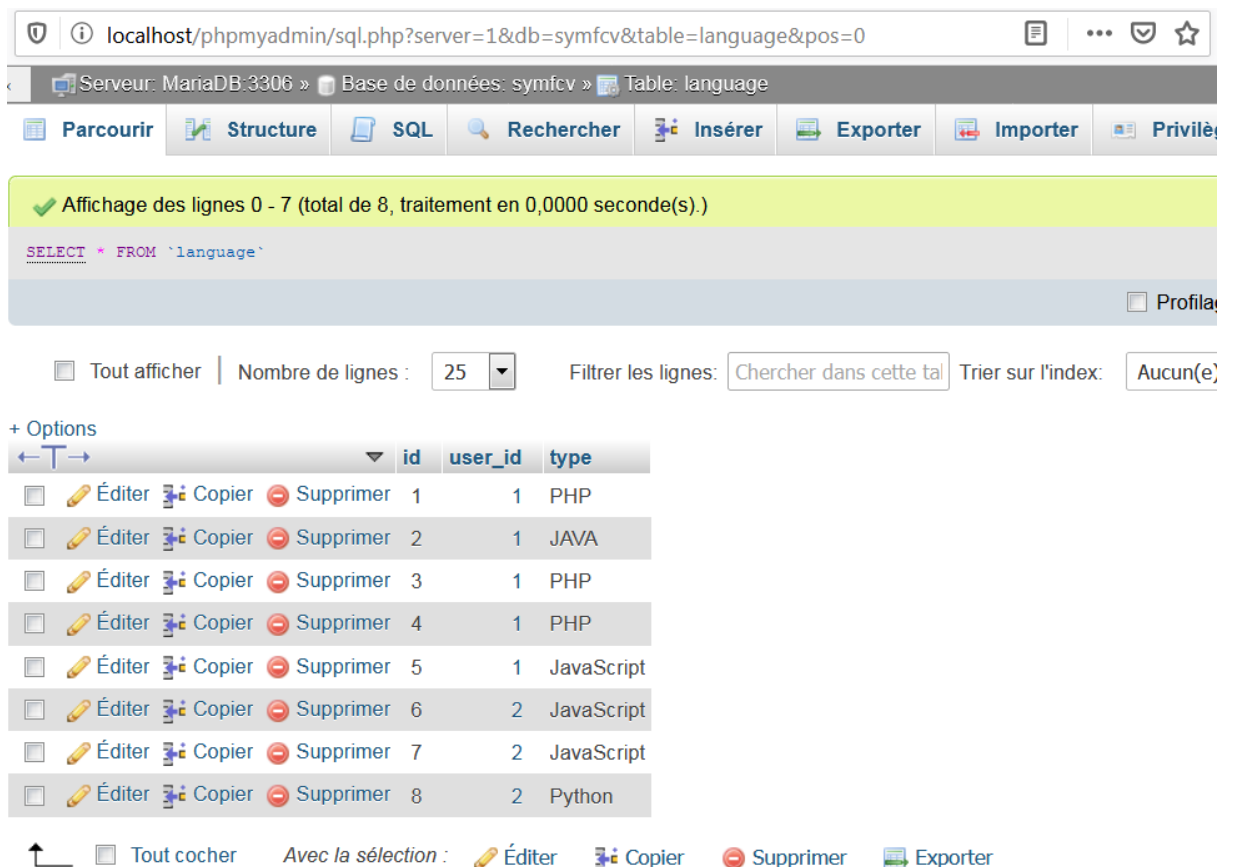
**no user with that username**

```
{# check if the website existe or not #}  
{% if user.website is not empty %}  
<tr><td><h2>Website </h2></td><td></td><td><h2> {{user.website}}</h2></td></tr>  
{% endif %}
```

11. Go back link to refresh and go back to generate another resume

12. I added Language entity with a relation ManyToOne to our User entity

```
/**  
 * @ORM\ManyToOne(targetEntity="App\Entity\User", inversedBy="languages")  
 * @ORM\JoinColumn(nullable=false)  
 */  
private $user;  
  
public function getId(): ?int  
{  
    return $this->id;  
}
```



localhost/phpmyadmin/sql.php?server=1&db=symfcv&table=language&pos=0

Serveur: MariaDB:3306 » Base de données: symfcv » Table: language

Parcourir Structure SQL Rechercher Insérer Exporter Importer Privileges

✓ Affichage des lignes 0 - 7 (total de 8, traitement en 0,0000 seconde(s).)

SELECT \* FROM `language`

Profilage

Tout afficher | Nombre de lignes : 25 | Filtrer les lignes: Chercher dans cette table | Trier sur l'index: Aucun(e)

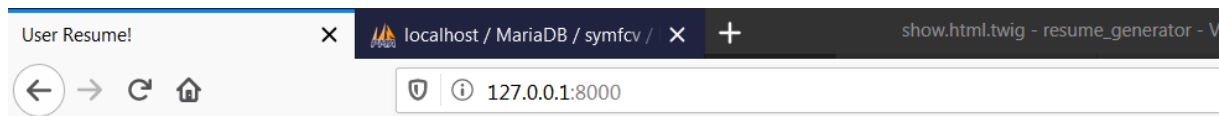
+ Options


	id	user_id	type
<input type="checkbox"/> Éditer Copier Supprimer	1	1	PHP
<input type="checkbox"/> Éditer Copier Supprimer	2	1	JAVA
<input type="checkbox"/> Éditer Copier Supprimer	3	1	PHP
<input type="checkbox"/> Éditer Copier Supprimer	4	1	PHP
<input type="checkbox"/> Éditer Copier Supprimer	5	1	JavaScript
<input type="checkbox"/> Éditer Copier Supprimer	6	2	JavaScript
<input type="checkbox"/> Éditer Copier Supprimer	7	2	JavaScript
<input type="checkbox"/> Éditer Copier Supprimer	8	2	Python

Tout cocher Avec la sélection : Éditer Copier Supprimer Exporter

13. I had to use twig to remove get unique languages ( remove duplicated languages)

```
{# get languages and remove duplicated languages #}
{% set newArray = [] %}
{% for language in user.languages %}
    {% if language.type not in newArray %}
        <tr><td>{{language.type}}</td><td>()</td></tr>
        {% set newArray = newArray|merge([language.type]) %}
    {% endif %}
{% endfor %}
</table>
```



**Name zack!** 

Status : I LOVE PHP

**Email** : mr.zakaria.hamidi@gmail.com

**Website** : linkedIn.com

## Languages:

PHP ()

JAVA ()

JavaScript ()

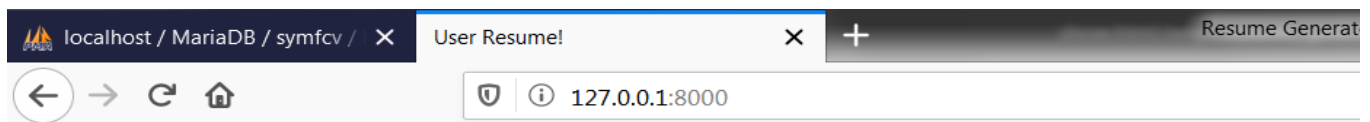
[go back](#)


14. I used twig conditions and loops to count how many time the languages were used, I used round to ceil the percentage

```

27
28     {# if there are no languages #}
29     {% if user.languages is not empty%}
30         <h2>Languages:</h2>
31         {# total of languages #}
32         {% set totallang= user.languages|length %}
33         ({{totallang}}) Times in total
34         <table style="width:10%"><tr>
35             {# get languages and remove duplicated languages then use the secon array to count how many time the vales are used #}
36             {% set newArray = [] %}
37             {% set i=0 %}
38             {% for language in user.languages%}
39                 {% if language.type not in newArray %}
40                     <td>{{language.type}}</td><td>|</td>
41                     {% set newArray = newArray|merge([language.type]) %}
42                 {% endif %}
43             {% endfor %}</tr><tr>
44                 {% for count in newArray %}
45                     {% for language in user.languages|sort %}
46                         {% if language.type==count %}
47                             {% set i=i+1 %}
48                         {% endif %}
49                     {% endfor %}
50                     <td>({{((i/totallang)*100)|round(2,'ceil')}})%</td><td>|</td>
51                     {% set i=0 %}
52                 {% endfor %}</tr>
53             </table>
54         {% endif %}
55

```



**Name zack!** 

Status : I LOVE PHP

**Email** : mr.zakaria.hamidi@gmail.com

**Website** : linkedIn.com

## Languages:

(5) Times in total  
 PHP | JAVA | JavaScript |  
 (60%) | (20%) | (20%) |


[go back](#)

15. I created The repository Entity
  16. Then I did the same as Languages, relation with user, after that I inserted data in DB
  17. I added a condition if there are no languages or repositories they won't appear
- First example

User Resume!

show.html.twig - resume\_generator -

127.0.0.1:8000

**Name : betty!** 

Status : frontend

**Email : b.etty@gmail.com**

**Languages:**  
(3) Times in total  
JavaScript | Python |  
(66.67%) | (33.34%) |


[go back](#)

Second example and the whole realization

DB / symfony / x User Resume!

show.html.twig - resume\_generator - Visual Studio Code [Administrator]

127.0.0.1:8000

**Name : zack!** 

Status : I LOVE PHP

**Email : mr.zakaria.hamidi@gmail.com**

**Website : linkedIn.com**

**Languages:**  
(5) Times in total  
PHP | JAVA | JavaScript |  
(60%) | (20%) | (20%) |

**Repositories:**  
Name of repository : Symfony resume  
Link : <https://symfony.com/index.php/what-is-symfony>  
Description : Framework that is going to make your programming easier  
Name of repository : facebox  
Link : <http://defunkt.io/facebox/>  
Description : This repository has 1969 stars and 422 forks. If you would like more information about this repository and my contributed code, please visit the repo on GitHub.

[go back](#)

Thank you for your Time

Waiting for a positive feedback, I am looking forward to discuss the code with you and get to the next step of recruiting process.