

Cahier des charges plateforme de petites annonces

Au cours de ces 10 prochains jours, vous allez devoir développer une plateforme de petites annonces avec le framework Laravel 5.

Le site doit permettre une fois inscrit de:

- Publier des annonces de recherche d'emplois.
- Publier des annonces de recherche de compétences.
- Evaluer les autres membres sur 5 tout en leur laissant un commentaire.

Vous demeurez libre du choix de la charte graphique et pouvez si vous le souhaitez travailler avec des frameworks css tel que bootstrap.

Pour une meilleure organisation de travail il est recommandé de suivre les étapes de réalisation suivantes :

- 1) Création du template parent en 2 colonnes (sidebar + main).**
- 2) Formulaire d'inscription (page 2)**
- 3) Formulaire d'édition de son profil (page 3)**
- 4) Annuaire des membres (page 4)**
- 5) Formulaire de création et édition des annonces (page 5)**
- 6) Page d'une annonce (page 6)**
- 7) Annuaire des petites annonces (page 7)**
- 8) Profil d'un membre (page 8)**
- 9) Module de notation (page 8)**

Page inscription : (*localhost/plateform/public/index.php/register*)

[ACCUEIL](#) | [ANNONCES](#) | [MEMBRES](#) [CONNEXION](#) / **[INSCRIPTION](#)**

Texte incitant à s'inscrire

Je suis ☐ un homme ☐ une femme

☐ J'accepte les CGU et CGV

[CGU](#) | [CGV](#) | [CONTACT](#)

Champs de la table users :

id, name, firstname, lastname, description, gender, email, zipcode, city, lat, lng, rate, birthday, password, remember_token, avatar, created_at, updated_at

Le champ gender :

Le champ gender contient **m** ou **f** (m = male , f = female) .

Le champ name :

Doit faire 5 caractères mini, ne doit contenir que des caractères alphanumériques et être unique.

Le champ birthday :

Il faudra vérifier que l'âge de l'utilisateur est supérieur à 18.

Pour se faire vous pouvez intégrer cette règle de validation dans la méthode validator de la classe UserController : **'birthday' => 'required|date|before:-18 years'**

Les champs lat et lng :

La latitude et longitude seront récupérées en fonction du code postal grâce à l'api de google maps. Il faudra donc extraire au moment du traitement de l'envoi du formulaire d'inscription les informations de la page <http://maps.googleapis.com/maps/api/geocode/json?address={{code postal}}%20france> via la fonction <http://php.net/manual/fr/function.file-get-contents.php> ; et le convertir en tableau associatif avec la fonction <http://php.net/manual/fr/function.json-decode.php> pour en extraire les données (de la même manière qu'en TP sur l'api Google map en JS).

Exemple de la marche à suivre :

Supposons que le contenu renvoyé par la page <http://site.com/api.json> soit celui-ci:

```
{ "info1" : "Foo",  
  "info2" : "Bar",  
  "info3" : ["hello","word"]  
}
```

Si l'on souhaite récupérer le contenu de cette page en php afin d'en extraire les données qui nous intéresse voici la marche à suivre :

```
<?php $json = file_get_contents('http://site.com/api.json');
```

```
/*
```

\$json contiendra l'intégralité du contenu renvoyé par la page <http://site.com/api.json> :

```
{ "info1" : "Foo",  
  "info2" : "Bar",  
  "info3" : ["hello","word"]  
}
```

```
*/
```

\$data = json_decode(\$json,true); // On converti l'objet json en tableau associatif interprétable par php

```
/*
```

\$data contiendra ce tableau associatif :

```
[ "info1" => "foo",  
  "info2" => "bar",  
  "info3" => ["hello","word"]  
]  
*/
```

Si je veux afficher "hello" (première élément du tableau de la propriété "info3"):

```
$hello = $data["info3"][0] ; // contient « hello »
```

Le champ city : il sera récupérer en même temps que la latitude et longitude

Le champ rate : il s'agira de la note moyenne reçu par l'utilisateur

Editer mon profil : (localhost/platform/public/index.php/user/edit)

[ACCUEIL](#) | [ANNONCES](#) | [MEMBRES](#) [CONNEXION / INSCRIPTION](#)

Editer mon mot de passe :

Ancien mot de passe

Nouveau mot de passe

MODIFIER

Pseudo

Nom

Prénom

Je suis ☐ un homme ☐ une femme

Email

Code de postal

Date de naissance (JJ/MM/YYYY)

Description

Photo de profil *

EDITER MON PROFIL

[CGU](#) | [CGV](#) | [CONTACT](#)

*La photo de profil sera sauvegardée dans le dossier ressources/assets/images/avatars/ en 3 formats :

- Version originale. Nom du fichier : **{{ id user }}_nom_image.jpg**
- Version « découpée » en 150 x 150 px. Nom du fichier : **thumb_{{ id user }}_nom_image.jpg**
- Version moyenne : 800px de large nom du fichier : **medium_{{ id user }}_nom_image.jpg**

Le nom du fichier sera enregistré dans le champ avatar de la table users.

Liens utiles :

Pour gérer l'upload de fichiers avec Laravel : <https://laravel.com/docs/5.3/filesystem#file-uploads>

Pour le redimensionnement et cropping des images vous pouvez installer la librairie suivante : http://image.intervention.io/getting_started/installation

Annuaire des membres : (localhost/platform/public/index.php/members)

ACCUEIL | ANNONCES | MEMBRES

CONNEXION / INSCRIPTION

RECHERCHER

Dans un rayon de :

30

Km

Mot clés :

Damien

Age :Mini

18

ans

Maxi

58

ans

RECHERCHER

1

2

3

Trier par distance ▼

PHOTO

Prénom NOM

Description courte (200 caractères)

Date d'inscription | age | Ville - distance

PHOTO

Prénom NOM

Description courte (200 caractères)

Date d'inscription | age | Ville - distance

PHOTO

Prénom NOM

Description courte (200 caractères)

Date d'inscription | age | Ville - distance

PHOTO

Prénom NOM

Description courte (200 caractères)

Date d'inscription | age | Ville - distance

PHOTO

Prénom NOM

Description courte (200 caractères)

Date d'inscription | age | Ville - distance

CGU | CGV | CONTACT

La photo affichée sera celle au format 150 x 150 px

Il faut pouvoir trier les utilisateurs par âge croissant, date d'inscription croissante, note moyenne décroissante et distance décroissante.

Exemple : localhost/platform/public/index.php/members?orderBy=age triera les membres par age croissant.

Au sujet de la pagination :

On affichera 5 utilisateurs par page. La pagination pourra s'effectuer aisément grâce aux méthodes **paginate** et **links** de la laravel : <https://laravel.com/docs/5.2/pagination>

Formulaire de publication / modification d'une annonce :

Url création d'annonce : localhost/platform/public/index.php/advert/create

Url édition d'annonce : localhost/platform/public/index.php/advert/{id}/edit

[ACCUEIL](#) | [ANNONCES](#) | [MEMBRES](#) [CONNEXION / INSCRIPTION](#)

Mes annonces :

TITRE ANNONCE 1

TITRE ANNONCE 2

☒ Je recherche ☐ Je propose

Titre

Description

Type :
☐ JOB ☐ STAGE ☐ ITERIM ☐ CDD ☐ CDI ☐ PRESTA

TARIF

XX

 € / hr

Choisir catégorie ▼

PUBLIER

[CGU](#) | [CGV](#) | [CONTACT](#)

Champs de la table adverts :

id, title, content, search, type, category_id, user_id, hourly_wage, created_at, updated_at

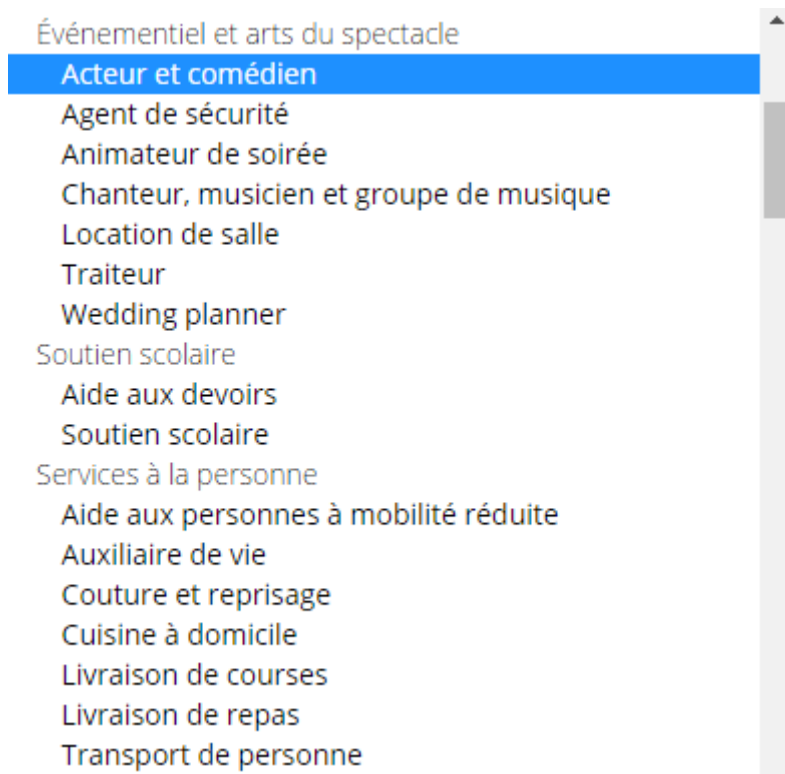
Nb : Le champ *search* est un booléen qui sera égal à **0** si l'utilisateur indique Je propose et à **1** s'il indique je recherche.

Le champ *hourly_wage* correspond au taux horaire (tarif).

Le champs *type* contiendra les valeurs *JOB / STAGE / ITERIM / CDD / CDI ou PRESTA*

Le champ select pour les catégories sera généré dynamiquement via la base de données (table categories). Vous pouvez mettre en place cette table en important simplement ce fichier SQL : <http://conception.website/3wa/PHP/Laravel/categories.sql>

**Les catégories devront être regroupées par selon leur catégorie parent et par ordre alphabétique.
Une catégorie parent ne pourra pas être sélectionnée :**



Champ select du formulaire permettant de sélectionner une catégorie

Extrait du code html à générer dynamiquement à partir du model Category :

```
<select name="category" placeholder="Choisissez un service">
  <optgroup label="Culture et loisirs">
    <option value="55">Accordage d'instruments</option>
    <option value="164">Association</option>
    <option value="52">Cameraman et photographe</option>
    <option value="127">Wedding planner</option>
  </optgroup>
  <optgroup label="Soutien scolaire">
    <option value="109">Aide aux devoirs</option>
    <option value="110">Soutien scolaire</option>
  </optgroup>
</select>
```

ACCUEIL ANNONCES MEMBRES		CONNEXION / INSCRIPTION
------------------------------	--	-------------------------

Auteur de l'annonce :

PHOTO

Prénom NOM
Age
Sexe
Ville + distance en km

Note : ★ ★ ★ ★ ★ (36)

EDITER

SUPPRIMER

TITRE DE L'ANNONCE

Description longue

"Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum."

Catégorie / TYPE (CDD etc.) / RECHERCHE ou PROPOSE

Date de publication

Tarif : XX€ / hr

CGU | CGV | CONTACT

NB : Seul l'auteur d'une annonce doit pouvoir l'éditer ou la supprimer.

ACCUEIL | **ANNONCES** | MEMBRES

CONNEXION / INSCRIPTION

RECHERCHER

Dans un rayon de :

30

 Km

Mot clés :

Webmaster freelance

☒ Propose ☐ Recherche

Type d'annonce :

☐ CDI

☐ CDD

☐ ITERIM

☐ JOB

☐ STAGE

Tarif mini :

5

 € / hr

Tarif maxi

30

 € / hr

Catégorie :

Choisir catégorie ▼

RECHERCHER

123

Trier par prix ▼

Titre annonce 1XX€ / h
Description courte (200 caractères)
Nom catégorie / Type de l'annonce (CDI, Job etc) / Auteur / Date

Titre annonce 2XX€ / h
Description courte (200 caractères)
Nom catégorie / Type de l'annonce (CDI, Job etc) / Auteur / Date

Titre annonce 3XX€ / h
Description courte (200 caractères)
Nom catégorie / Type de l'annonce (CDI, Job etc) / Auteur / Date

Titre annonce 4XX€ / h
Description courte (200 caractères)
Nom catégorie / Type de l'annonce (CDI, Job etc) / Auteur / Date

Titre annonce 5XX€ / h
Description courte (200 caractères)
Nom catégorie / Type de l'annonce (CDI, Job etc) / Auteur / Date

CGU | CGV | CONTACT

Il doit y avoir 5 annonces par page.

Au sujet du moteur de recherche des annonces :

La recherche dans un rayon de X km s'effectuera grâce aux coordonnées GPS de l'auteur de l'annonce (lat et lng) et celle de l'utilisateur qui effectue la recherche.

Pour développer cette fonctionnalité nous utiliseront cette extension Laravel :

<https://packagist.org/packages/jackpop/geodistance>

Il faut pouvoir trier les annonces par prix décroissant, distance croissante ou date décroissante.

Le champ select pour les catégories sera généré dynamiquement via la base de données (table categories).

ACCUEIL ANNONCES MEMBRES		CONNEXION / INSCRIPTION
------------------------------	--	-------------------------

PHOTO

Prénom NOM
Age
Sexe
Ville + distance en km

Note : ★ ★ ★ ★ ★ (36)

☆☆☆☆☆

Ecrire commentaire...

Noter

Notes reçus :
☆☆☆☆☆
"Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod »
Auteur / Date

☆☆☆☆
"Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod »
Auteur / Date

PROFIL DE {{ firstname }} {{ lastname }}
Description longue
"Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum."

ANNONCES PUBLIEES (4) :

Titre annonce 1 **XX€ / h**
Description courte (200 caractères)
Nom catégorie / Type de l'annonce (CDI, Job etc) / Auteur / Date

Titre annonce 2 **XX€ / h**
Description courte (200 caractères)
Nom catégorie / Type de l'annonce (CDI, Job etc) / Auteur / Date

Titre annonce 3 **XX€ / h**
Description courte (200 caractères)
Nom catégorie / Type de l'annonce (CDI, Job etc) / Auteur / Date

Titre annonce 4 **XX€ / h**
Description courte (200 caractères)
Nom catégorie / Type de l'annonce (CDI, Job etc) / Auteur / Date

CGU | CGV | CONTACT