

Document technique – Site Web TRT-CONSEIL

02/04/2023

Caglar YASAR – ECF d'Entrainement

Graduate Développeur Web Full Stack – Promo ZEMANEK

SOMMAIRE

Table des matières

1.	Maquettage	3
1.1.	Page principale	3
1.2.	Page d'accueil de l'administrateur	3
1.3.	Page d'accueil du consultant	4
1.4.	Page d'accueil d'un candidat ou recruteur	4
2.	Modélisation de la base de données.....	5
2.1.	Utilisation de phpMyAdmin en local	5
2.2.	Diagramme de classe.....	6
3.	Développement	7
3.1.	Environnements de développement	7
3.2.	Développement Backend : Symfony	7
3.3.	Développement Frontend : Bootstrap.....	7
4.	Déploiement de l'application.....	8
4.1.	Test en local	8
4.2.	Déploiement de l'application sur Heroku	8

Ce document technique décrit mes réflexions initiales sur le projet (wireframes et diagrammes de classe), ainsi que le choix et la mise en place des technologies choisies.

1. Maquettage

A partir des besoins du client, j'ai créé les wireframes sur Adobe Xd, où j'ai représenté les écrans principaux. Vous pouvez retrouver le lien Adobe Xd pour voir le wireframe en ligne à l'adresse suivante : <https://xd.adobe.com/view/c994b39d-74d0-4147-a0c6-455fa1f57987-9cc7/>

1.1. Page principale

La page d'accueil présente la liste des annonces proposées par les recruteurs.

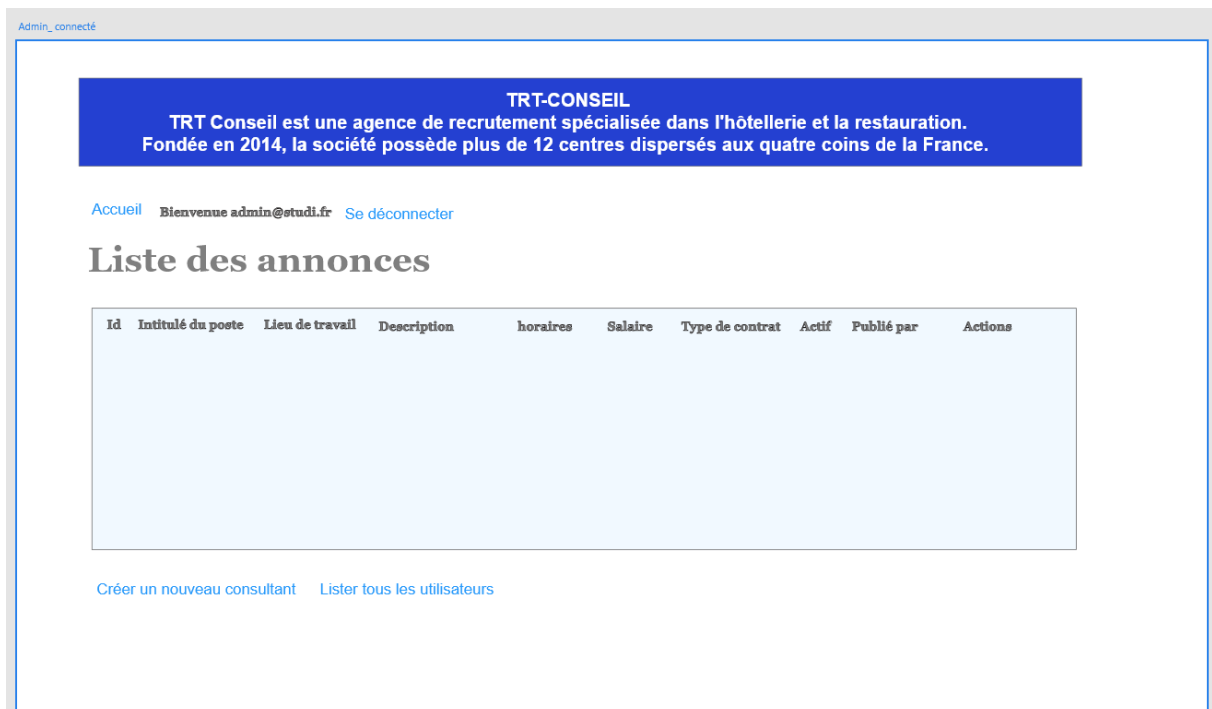
Pour pouvoir postuler, il faut se connecter ou s'inscrire si l'on n'a pas encore de compte.



1.2. Page d'accueil de l'administrateur

La page administrateur permet de créer des comptes avec un profil **consultant**.

Il pourra également lister tous les comptes actuellement créées.



1.3. Page d'accueil du consultant

Le consultant permet de valider l'activation de comptes candidat et recruteur, l'activation d'une annonce publiée par le recruteur, ainsi que la validation lorsqu'un candidat répond à une offre.



1.4. Page d'accueil d'un candidat ou recruteur

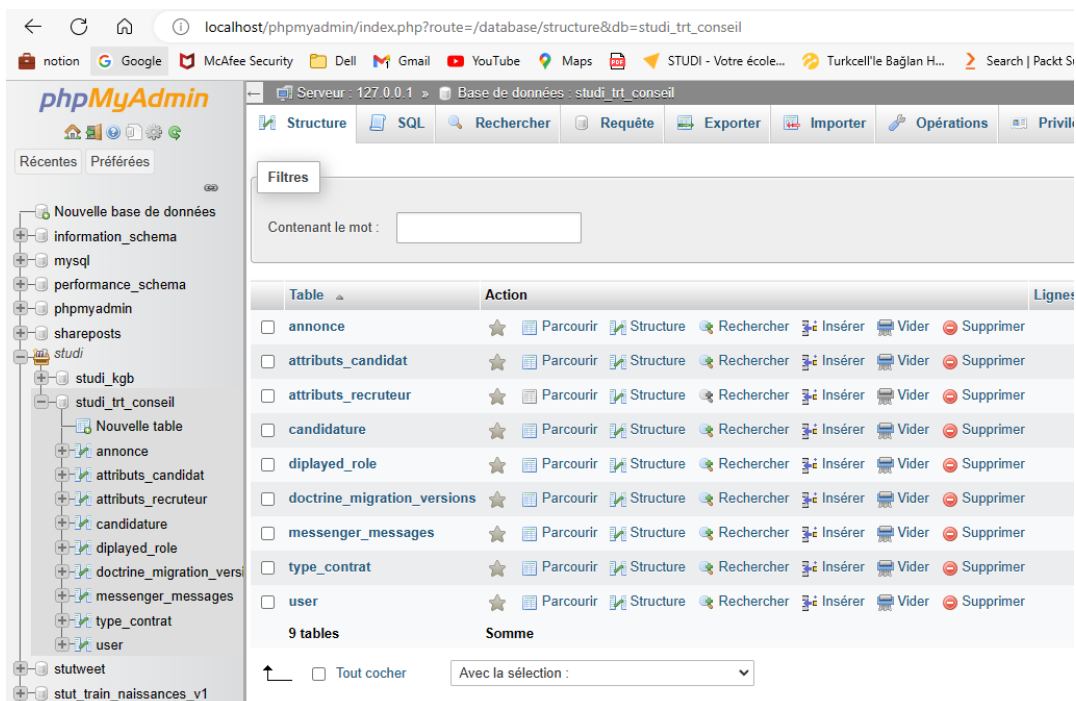
Les profils Candidat et Recruteurs ont les mêmes caractéristiques : possibilité de s'inscrire en toute autonomie (une activation devra cependant être faite par un consultant). Le bouton « Mon profil » permet de renseigner des infos supplémentaires sur les candidats et les recruteurs.



2. Modélisation de la base de données

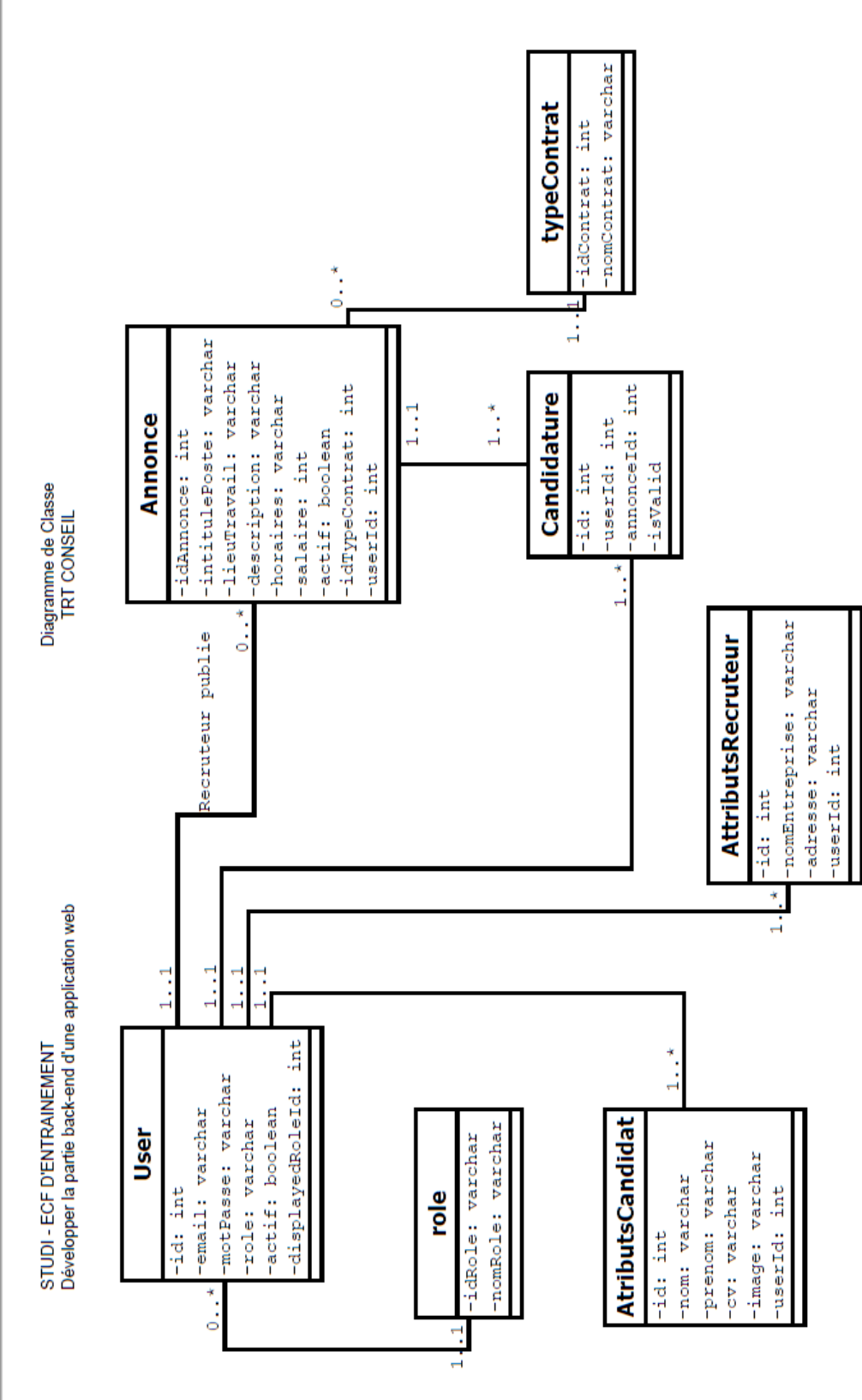
2.1. Utilisation de phpMyAdmin en local

J'ai utilisé MySQL en local pour la mise en place de la base de données en local. L'environnement utilisé est XAMPP et l'utilisation de phpMyAdmin 5.2 m'a grandement aidé dans la mise en place de la base données et l'export vers la base de données en ligne sur Heroku.



2.2. Diagramme de classe

La page suivante représente le diagramme de classe du projet TRT-Conseil.



3. Développement

3.1. Environnements de développement

J'utilise Visual Studio Code pour le développement du code et Github pour le versioning.

Pour les tests en local, j'utilise phpMyAdmin comme bases de données. J'utilise les Datafixtures pour créer le compte administrateur et également pour la création d'entrées dans les tables pour avoir une base de données initiale.

Pour l'envoi de mail, j'utilise Mailgun comme service. Il faut au préalable aller sur son compte pour paramétrer le serveur SMTP, puis dans le fichier .env, il faut renseigner les valeurs récupérées sur son compte Mailgun comme indiqué ci-dessous :

```
MAILER_DSN=mailgun+smtp://admin@yasardev.fr:a73efb26c9-30344472-312542e7@smtp.eu.mailgun.org:587?region=eu
```

3.2. Développement Backend : Symfony

J'utilise Symfony CLI pour le développement de mon application, ce qui permet d'écrire un minimum de code pour la réalisation de notre projet.

Les lignes de commande suivantes permettent de créer notre projet et de démarrer le serveur web local :

```
symfony new trt-conseil --webapp
cd my_project symfony server:start -d --port=8050
```

La création de la base de données se fait à l'aide du fichier .env à la racine :

```
DATABASE_URL=mysql://root:123456@127.0.0.1:3306/stud_trt-conseil
```

Il faut ensuite taper la ligne suivante pour créer la base de données dans MySQL :

```
php bin/console doctrine:database:create
```

Création des entités (tables) via la commande suivante

```
php bin/console make:entity
```

Lancement de la migration et création des tables dans la base de données

```
php bin/console doctrine:migrations:diff
php bin/console doctrine:migrations:migrate
```

Création des utilisateurs

```
php bin/console make:user
```

3.3. Développement Frontend : Bootstrap

Bootstrap permet de gérer le style du site web sans écrire une seule ligne de css.

4. Déploiement de l'application

4.1. Test en local

Pour tester le code en local, cliquez sur le lien ci-dessous [contactarberen/studi_trt_conseil_v3 \(github.com\)](https://github.com/contactarberen/studi_trt_conseil_v3) pour récupérer le code (par exemple allez dans Code puis download zip).

Un fois le code dézippé, ouvrez avec Visual Studio Code le répertoire et allez dans le fichier .env pour modifier la ligne suivante avec les informations de votre base de données :

```
DATABASE_URL=mysql://root:123456@127.0.0.1:3306/studi_trt-conseil
```

Taper ensuite la commande suivante pour installer tous les composants nécessaires au fonctionnement de l'application web :

```
composer update
```

Tapez à nouveau les commandes du paragraphe 3.2 pour recréer la base de données et lancer la commande suivante pour créer les entrées dans la base de données notamment l'administrateur :

```
php bin/console doctrine:fixtures:load --append
```

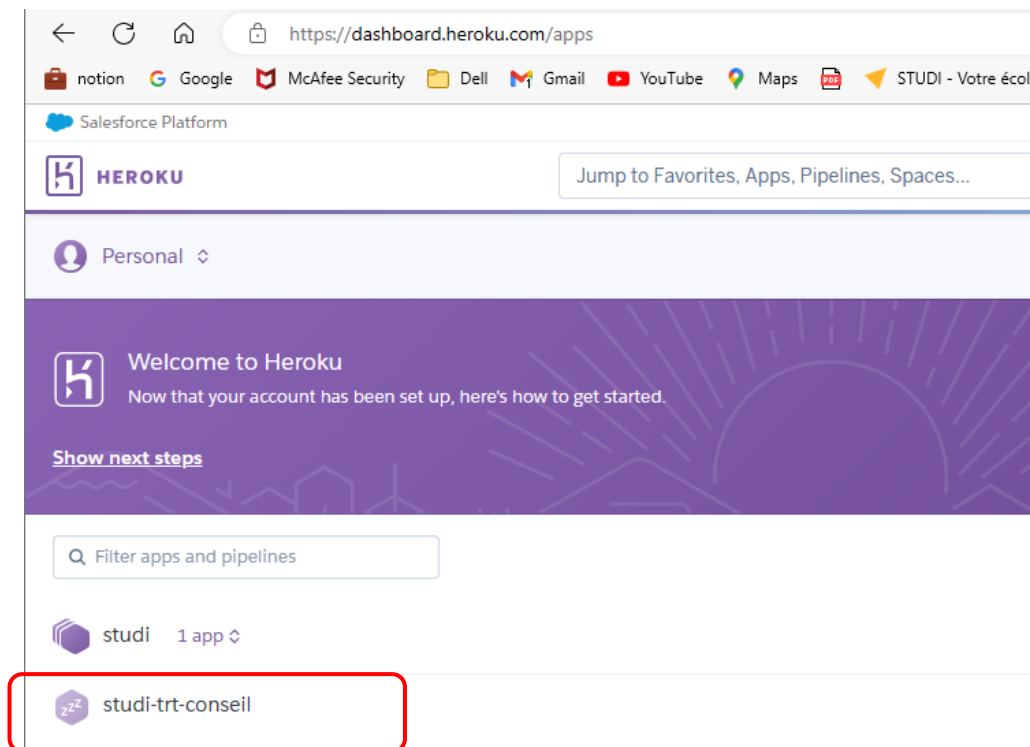
Lancer le serveur web en local pour tester le code comme indiqué au paragraphe 3.2.

4.2. Déploiement de l'application sur Heroku

Le site web en ligne se trouve à l'adresse suivante :

[Listes des annonces \(studitrconseil.herokuapp.com\)](https://studitrconseil.herokuapp.com/)

Le déploiement est facilité en utilisant Heroku CLI comme indiqué dans les cours Studi.



Via Heroku CLI, nous devons nous loguer (heroku login), créer l'espace heroku distant (heroku create), créer nos variables d'environnement (heroku config:set APP_ENV=prod, ...).

Pour la base de données en ligne, j'utilise jawsdb qui est gratuit pour une base de données inférieure à 5 Mo.

The screenshot shows the Heroku dashboard for the application 'studi-trt-conseil'. The top navigation bar includes links for Overview, Resources, Deploy, Metrics, Activity, Access, and Settings. The main content area is divided into three columns:

- Installed add-ons:** Shows 'JawsDB MySQL' (Kitefin Shared, jawsdb-crystalline-62344) with a price of \$0.00/month. A red box highlights this add-on.
- Dyno formation:** Shows 'eco dynos' with a price of \$0.00/month. The configuration is 'web heroku-php-apache2 public/' and the status is 'ON'.
- Collaborator activity:** Shows the user 'yasarcaglar@yahoo.fr' with 2 deploys.


The right sidebar displays the 'Latest activity' log, showing a series of events including configuration variable updates (MAILER_DNS, MAILER_DSN, DATABASE_URL) and successful builds, all performed by 'yasarcaglar@yahoo.fr'.

Nous retrouvons dans la fenêtre Settings nos variables d'environnement précédemment crée :


App Information

App Name

studi-trt-conseil

Region  United States

Stack heroku-22

Framework  PHP











Slug size 26.9 MiB of 500 MiB

Heroku git URL <https://git.heroku.com/studi-trt-conseil.git>

Config Vars

Config vars change the way your app behaves. In addition to creating your own, some add-ons come with their own.

Config Vars [Hide Config Vars](#)

APP_ENV	prod	 
APP_SECRET	da5ea98320df83a71ed04af3c356	 
DATABASE_URL	mysql://cgcih5wkvny22boy:sj6	 
JAWSDB_URL	mysql://cgcih5wkvny22boy:sj6	 
MAILER_DSN	mailgun+smtp://caglar@yasarc	 
KEY	VALUE	Add