

DOCUMENTATION TECHNIQUE

E-SPORTIFY

Etienne COYEN – ECF Studi

Session mars-avril 2025

SOMMAIRE :

- A. Réflexions initiales technologique sur le sujet
- B. Configuration de mon environnement de travail
- C. Diagramme de classe
- D. Diagramme d'utilisation
- E. Diagramme de séquence
- F. Deploiement

A. Réflexions initiales technologique sur le sujet :

Langage Front-end :

- TWIG: j'ai choisi de l'utiliser car twig est utilisé avec le framework symfony 7.
- BOOSTRAP : j'ai choisi de l'utiliser afin d'avoir un rendu agréable et surtout un site responsive.Bootstrap est LE framework CSS a savoir utiliser, il permet de créer des sites web, sans avoir un code a plusieurs milliers de lignes pour un beau rendu avec un beau design.
- JAVASCRIPT : j'ai choisi de l'utiliser afin d'avoir un site web dynamique et agréable.

Langage back-end :

- Framework Symfony 7 : j'ai choisi de l'utiliser car ce framework est en php. Ce framework me sert a bien respecter le canevas MVC, ainsi que la sécurité d'un site web.Il est facile d'accès même en ayant que la base de connaissance en php. C'est aussi un framework open-source avec de la documentation détaillé.

Base de données :

- MariaDB : j'ai choisi de l'utiliser car il est un prérequis obligatoire pour tout développeur web qui se respecte. MariaDB ainsi que MySQL est la base et la référence quand nous parlons de base de données. Je l'ai choisi aussi car c'est une base de données relationnel et que Hostinger le propose dans son phpMyAdmin.

Déploiement :

- Hostinger: j'ai choisi cet hébergeur car c'est un hébergeur très simple d'utilisation et facile à comprendre. Une documentation ainsi qu'un forum est mise en place afin de répondre a nos questions.
- PhpMyAdmin : proposer par hostinger (Version : 5.2.1)

Sécurité :

J'ai utilisé la base de la sécurité c'est-à-dire :

- Des mots de passes hashés ;
- Une politique de mot de passe;
- Mettre un login et mot de passe a sécurité forte sur la connexion à la base de données ;
- Utilisé les jetons CSRF lors des formulaires ;
- Des rôles bien distinct avec différent droits ;
- Une hiérarchie de rôles ;
- Dans ma base de données, j'ai spécifié à chaque fois si c'était une date, INT, décimal, text ou varchar en spécifiant au maximum le nombre de caractère permis ;
- Dans les balises input, j'ai précisé au maximum ce qu'on attendait (exemple : dans un input ou on attend un email, type=email).

Exemple d'un code avec de la sécurité :

```
# Contrôle d'accès pour définir les permissions selon les chemins
access_control:
- { path: ^/admin, roles: ROLE_ADMIN }
- { path: ^/employee, roles: ROLE_EMPLOYEE }
- { path: ^/user, roles: ROLE_USER }

# Hiérarchie des rôles
role_hierarchy:
  ROLE_ADMIN: [ROLE_EMPLOYEE, ROLE_USER]
  ROLE_EMPLOYEE: [ROLE_USER]
```

Je compte perfectionner au maximum la cybersécurité car c'est important pour l'utilisateur de savoir qu'il peut naviguer en toute sécurité. Ainsi que je mettrai a jour ma base de donnée si une nouvelle version sort.

Je fais tester mon site sur snyk afin d'avoir une meilleur sécurité.

B. Configuration de mon environnement de travail

Mon IDE :

J'ai choisi comme IDE, Visual Studio qui est pour moi le plus simple éditeur de code ainsi que le plus complet. En effet avec ces nombreuses extensions qui peuvent nous faciliter la vie et ainsi moins faire d'erreur, ce IDE n'est pas négligeable.

Extensions principalement utilisées :

- Better Comments (sert a mettre différents commentaire selon leur importance ou non) ;
- HTMLHint ;
- Indent-rainbow (sert à mieux visualiser ou on en est dans les if, for, while,etc...) ;
- Material icon theme (sert à mettre différents icones sur vos fichiers, dossiers) ;
- Microsoft Edge Developer Tools for VS code (permet de visualiser directement son site sur l'éditeur).

Mon OS :

J'ai utilisé mon ordinateur personnel pour réaliser ce projet, il est en Windows 11 Famille avec une version 23H2.

Mon environnement applicatif :

J'ai déployé mon site web grâce a hostinger qui est simple d'utilisation.

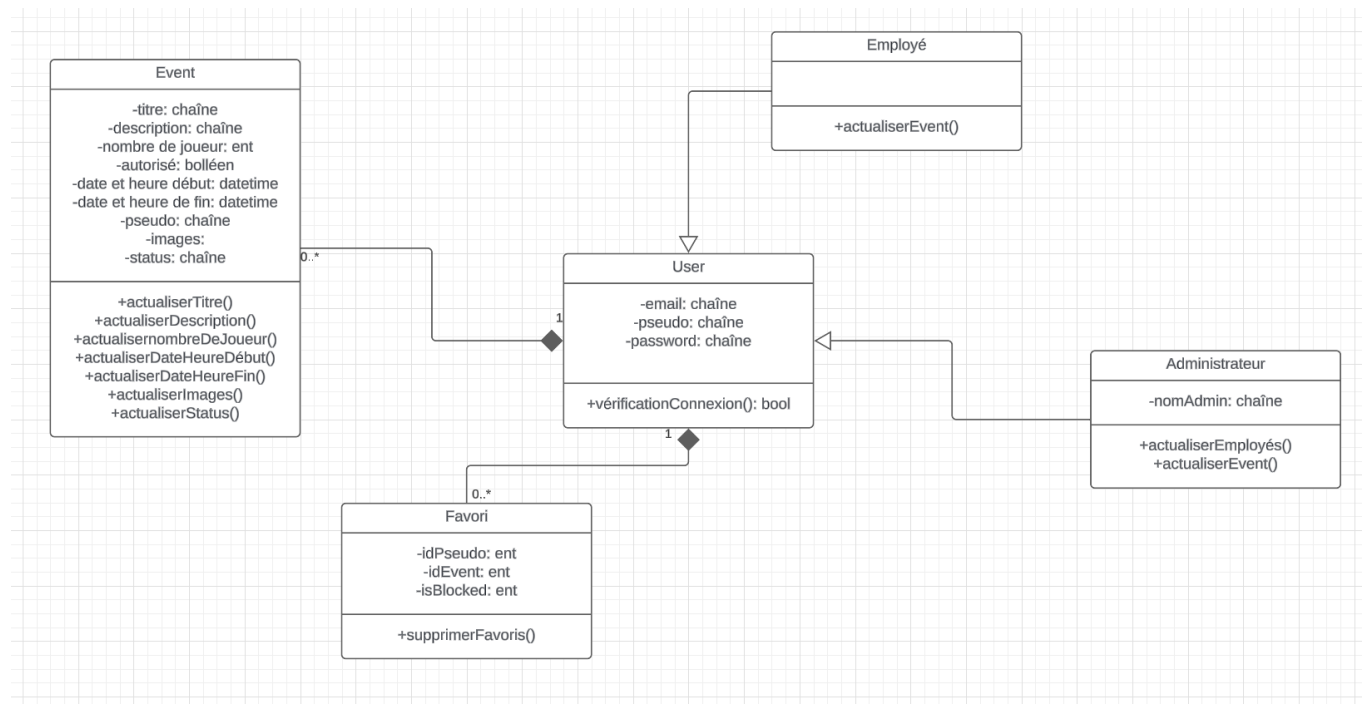
J'ai testé mon site web grâce à Docker qui peut faire des containers avec des images afin de pouvoir tout tester localement.

J'ai aussi testé avec différents outils mon code :

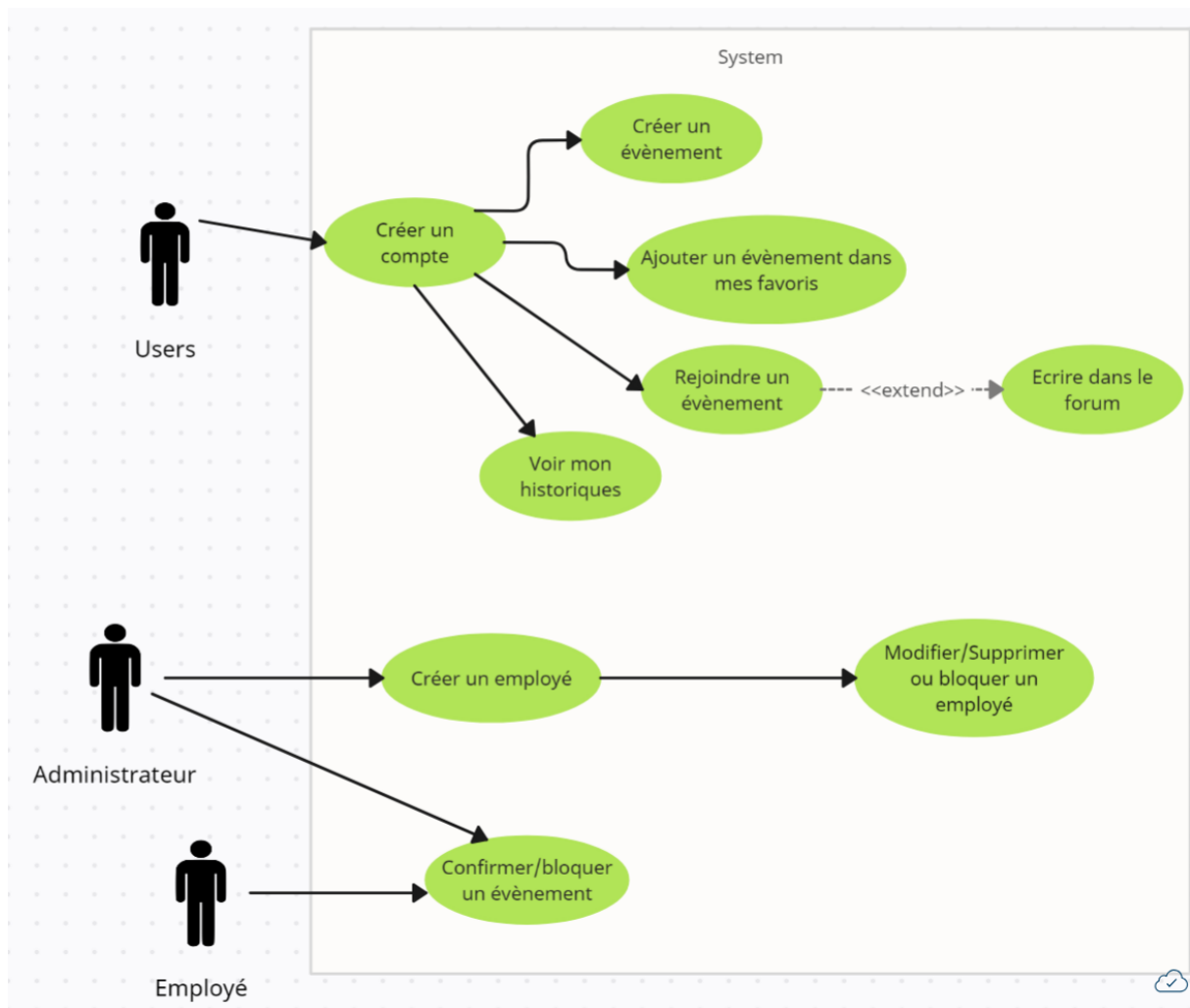
- Snyk qui est un outil qui permet de voir nos failles de sécurité ;
- J'ai aussi regardé directement dans l'éditeur que propose hostinger qui nous met des icones sur les morceaux de code qui n'ont pas d'utilité.

Sur l'hébergeur Hostinger, nous pouvons faire un test de vitesse afin de savoir si notre site web est trop lent ou trop lourd.

C. Diagramme de classe

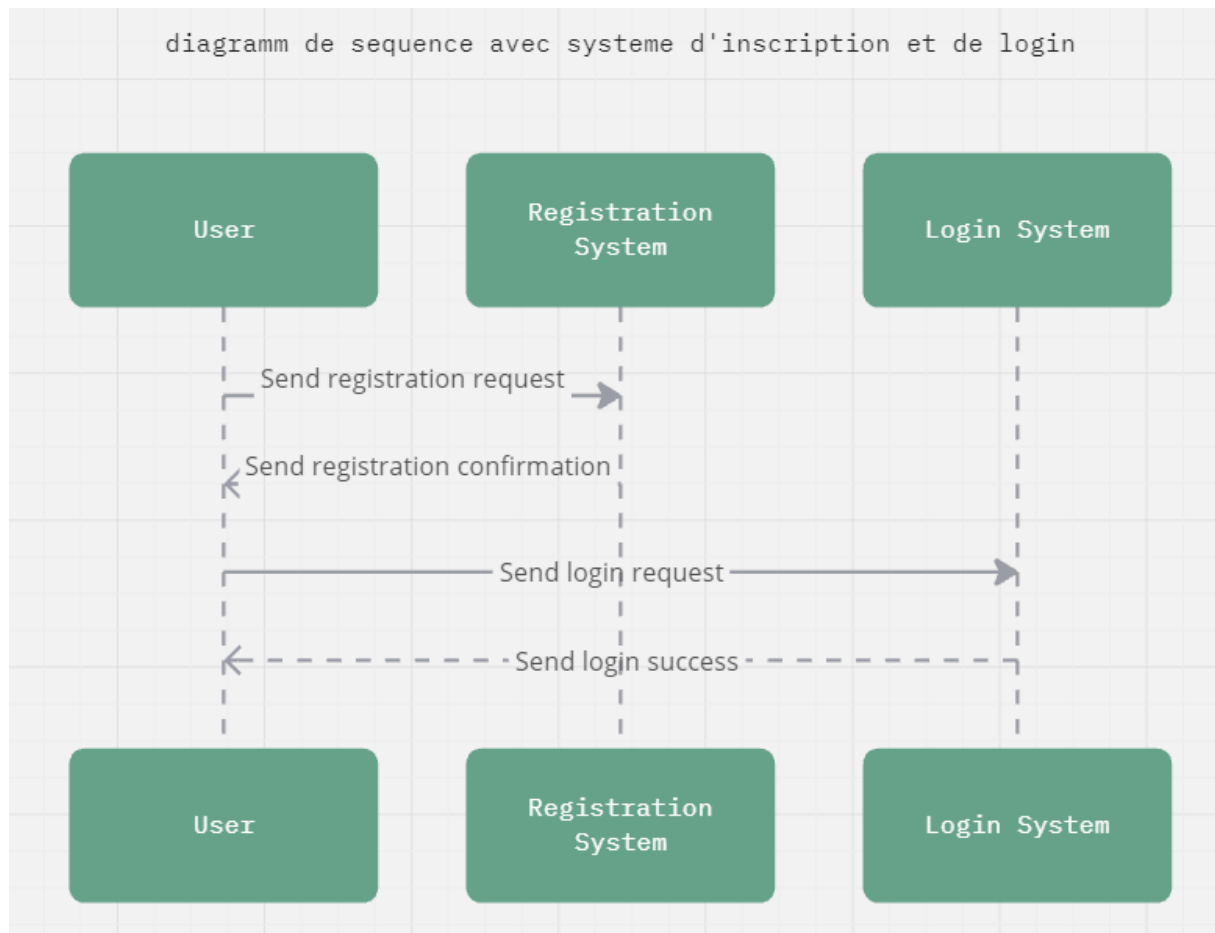


D. Diagramme d'utilisation



E. Diagramme de séquence

Pour une inscription et un login:



F. Déploiement

Pour mon déploiement, j'ai décidé après mes recherches de prendre Hostinger. Ce qu'y a décidé malgré le fait que ce soit payant sont ses nombreux avantages.

- Même si c'est payant, le prix reste complètement accessible ;
- Possibilité de transférer mon site via Git ou SSH ;
- Base de données via PhpMyAdmin ;
- Test de rapidité ;
- Boîte mail directement intégré dans mon espace.

Une fois que mon application a été testée complètement en local, j'ai pu envisager de le déployer sur internet. Les différentes étapes sont celle-ci :

1. Créer un compte et choisir mon forfait sur Hostinger ;
2. Transférer mes fichiers en SSH grâce à BitVise (pour ceci, dans mon espace hostinger j'ai pris un nom de domaine et j'ai activé le SSH) ;
3. J'ai ensuite créé ma base de données et ma boîte mail ;
4. Une fois transféré j'ai modifié différents paramètres
 - a. Le chemin d'index, afin que l'application fonctionne correctement
 - b. Mon .env qui permet de redistribuer au différents serveur (serveur mail, base de données...)
 - c. Mon chemin pour mon code css
5. J'ai intégré ma dataFixture à ma base de données afin d'avoir mon administrateur ;
6. J'ai testé mon site web afin de vérifier toutes les fonctionnalités
7. J'ai testé sa vitesse, afin de savoir si je pouvais l'améliorer
8. Enfin j'ai modifié dans mes paramètres dans le .env j'ai mis que mon application n'était plus en développement (dev) mais en production (prod)(ce qui est conseillé dans la documentation de symfony). Et j'ai désactivé le débogage de symfony afin d'avoir un site totalement propre.