Table des matières

[1. Cadrage du projet 2](#_Toc512206640)

[Charte projet 2](#_Toc512206641)

[Cahier des charges (A2) 2](#_Toc512206642)

[Planning (A1) 6](#_Toc512206643)

[2. Réalisation du projet 9](#_Toc512206644)

[Définition de l’architecture (A1) 9](#_Toc512206645)

[Sprint Meeting (A repeter) (A1) 10](#_Toc512206646)

[Développement du code en fonction des cas de tests (A repeter) (A1) 10](#_Toc512206647)

[3. Clôture du projet 10](#_Toc512206648)

[Validation des tests fonctionnels (A1) 10](#_Toc512206649)

[REX du projet (A1) 10](#_Toc512206650)

[Bilan (A1) 10](#_Toc512206651)

[4. Annexes 11](#_Toc512206652)

[Outils 11](#_Toc512206653)

# Cadrage du projet

## Charte projet

### Objectifs (A1)

Les objectifs de ce projet sont la création d’un site web permettant au BDE du CESI de gérer efficacement les activités proposées aux élèves et une boutique de goodies. Ce site devra pouvoir permettre aux différentes personnes du CESI de s’inscrire et de pouvoir accéder à différents contenus suivants s’ils sont élèves, membre du BDE ou membre de l’administration.

### Enjeux (A2)

Etant un projet à titre éducatif il n’y a pas de réels enjeux économiques. De plus il n’y a pas de réelle partie prenante au projet. D’après l’intitulé, la seule partie prenante fictive est le BDE du CESI.

### Acteurs (A1)

Les acteurs impactant le projet sont :

- Du coté client : Le BDE et l’administration de l’école

- Du coté prestataire : Equipe de 3 élèves de l’Exia.CESI (Florian VOUTERS - Robin MAISANO - Pierre GILLY

Les acteurs pouvant être impactés par le projet :

- Toutes les personnes du CESI

### Définition des responsabilité (A1)

Le groupe prestataire est composé de 3 personnes

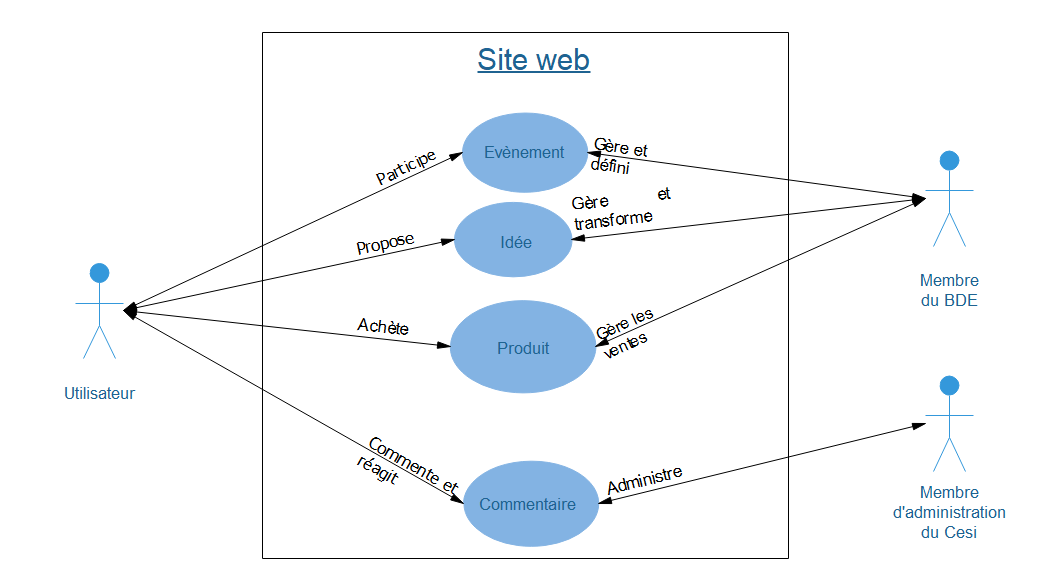
- Florian VOUTERS -> Chef de projet - Responsable du BackEnd

- Robin MAISANO -> Responsable du FrontEnd

- Pierre GILLY -> Responsable de la Base de Données

## Cahier des charges (A2)

Diagramme de cas d’utilisation



### Besoins

**Quoi ?** Nous devons réaliser un site web

**Qui ?** Le client est le BDE de l’école CESI

**Où ?** Le projet est pour tous les CESI : soit 24 campus répartis dans l’hexagone

**Quand ?** Le projet commence le lundi 09/04 et finit le vendredi 20/04

**Comment ?** Le projet sera réalisé grâce au framework « Symfony » et la plateforme de versioning « Github »

**Pourquoi ?** Le projet doit répondre au problème de la communication dans les écoles ainsi qu’à la simplification de vente de goodies

### Contraintes

Les contraintes de ce projet sont :

- Le site doit être adapté pour tous les types d’appareils (ordinateur, tablette ou smartphone)

- La charte doit être cohérente avec cohérente avec celle du CESI, de l’EI ou de l’EXIA

- Le site doit être rapidement visible dans Google

### Critères de réussites

Le projet pourra être réussit lorsque toutes les fonctionnalités demandées seront validées.

### Tests fonctionnels

En tant qu’**étudiant** de l’école, je peux m’inscrire sur le site en fournissant mon nom, prénom et adresse mail ainsi qu’un mot de passe contenant au moins une majuscule et au moins un chiffre.

**Test : Utiliser la méthode addUser puis appeler le getter de chaque attribut pour vérifier les informations.**

En tant qu**’étudiant**, après avoir réalisé l’inscription, je peux me connecter.

**------Test : Utiliser un getter pour vérifier l’inscription puis tester sur le Front End avec le compte voulu**

En tant qu**’étudiant**, je peux proposer une idée d’activité aux membres du BDE. Cette idée sera visible de tous les visiteurs connectés au site dans la partie « boite à idées ».

**Test : Utiliser la fonction addActivity puis utiliser un getter pour vérifier le résultat dans la BDD**

En tant qu’**étudiant connecté**, je peux accéder à la liste des activités proposées par le BDE et m’inscrire à l’une ou l’autre d’entre elles.

**------Test : Utiliser la fonction registerTo utiliser un getter pour vérifier le résultat dans la BDD**

En tant qu’**étudiant connecté,** je peux accéder à la « boite à idée » et voter pour les activités proposées par les autres étudiants.

**Test : Utiliser la fonction voteTo puis utiliser un getter pour vérifier le résultat dans la BDD**

En tant qu’**étudiant** **connecté**, je peux ajouter des photos sur les événements passés pour lesquels j’étais inscris.

**Test : Utiliser la fonction addReactionToActivity puis utiliser un getter pour vérifier le résultat dans la BDD**

En tant qu’**étudiant**, je peux voir les photos d’un événement passé, les liker et les commenter.

**Test : Utiliser les fonctions registerToActivity et addReactionToComment puis utiliser un getter pour accéder aux résultats de la BDD**

En tant que **membre du BDE**, je peux poster une manifestation avec une description, une image et une date, dans une partie « événement du mois ». Cette manifestation peut être ponctuelle ou récurrente, payante ou gratuite.

**Test : Utiliser la fonction setActivity et grâce à getActivityByID, vérifier toutes les informations**

En tant que **membre du BDE**, je peux accéder à l’ensemble des manifestations proposées par les étudiants dans la partie « boîte à idées ».

**------- Test : GivenAnAdminMemberShouldViewAllSuggestions**

En tant que **membre du BDE**, je peux choisir une manifestation proposée dans la partie « boîte à idées » proposée par les étudiants de l’école. Une fois cette manifestation sélectionnée, je dois lui attribuer une date et éventuellement modifier l’image et la description. L’étudiant ayant proposé cette idée sera notifié que son idée a été retenue.

**----- Test : Fonction GivenActivityWhenAdminMemberShouldAcceptActivity**

En tant que **membre du BDE**, pour chaque manifestation proposée, je peux accéder à la liste des inscrits et la télécharger au format PDF ou CSV.

**----- Test : Visuel**

En tant que **membre du BDE**, je peux administrer la partie photo et commentaire de la partie « événements passés ».

**----- Test : Fonction GivenAButtonDeleteWhenAdminMemberShouldDelete**

En tant que **personnel CESI**, je peux notifier l’ensemble des membres du BDE que certaines photos, commentaires, manifestations peuvent nuire à l’image de l’école. Dans ce cas les éléments ne peuvent être rendus public.

**---- Test : Utiliser la fonction addReport puis utiliser un getter pour vérifier l’état de visibilité du commentaire**

En tant que **personnel CESI**, je peux naviguer sur l’ensemble du site avec les mêmes droits que les étudiants.

**Test : Fonction memberModeratorShouldBeUserToo**

En tant que **personnel CESI**, je peux télécharger l’ensemble des photos postés par les étudiants et les membres du BDE

**Test : GivenAModeratorMemberWhenDownloadPicturesShouldDownloadAll**

Dans cette partie **les membres du BDE** peuvent ajouter, supprimer des produits avec nom, description et prix, en les classant par catégories (ces dernières peuvent être ajoutées également par les membres du BDE).

**Test : Utiliser addProduct, vérifier avec les getters le produit en BDD puis le supprimer avec deleteProduct**

**Les étudiants connectés** peuvent passer commande par l’intermédiaire d’un panier. Si la commande n’est pas finalisée par l’étudiant alors le panier est sauvegardé et l’étudiant retrouve sa liste de produits à sa prochaine reconnexion.

**Test : GivenAMemberWhenPurchasingShouldKeepInDB**

Lorsqu’une commande est passée par **un étudiant**, **les membres du BDE** reçoivent une notification par mail. Ils doivent alors donner un RDV à l’étudiant pour la transaction et la remise des goodies. Par la suite, un compte PayPal sera mis en place. Si vous pouvez préparer le terrain, ce serait un plus très apprécié.

**Test : Utiliser la fonction basketSold puis vérifier l’envoie de mail**

Dans la partie boutique, une partie affichant les 3 articles les plus commandés doit être visible et l'on doit pouvoir naviguer dans la liste d'article en filtrant par catégorie.

**Test :**  **GivenMarketPlaceWhenOnTopShouldReturnThreeProduct**

Afin de faciliter la recherche des goodies, il faut également offrir une fonctionnalité de recherche : bar simple avec auto complétion ou formulaire avec filtre sur le prix et les catégories

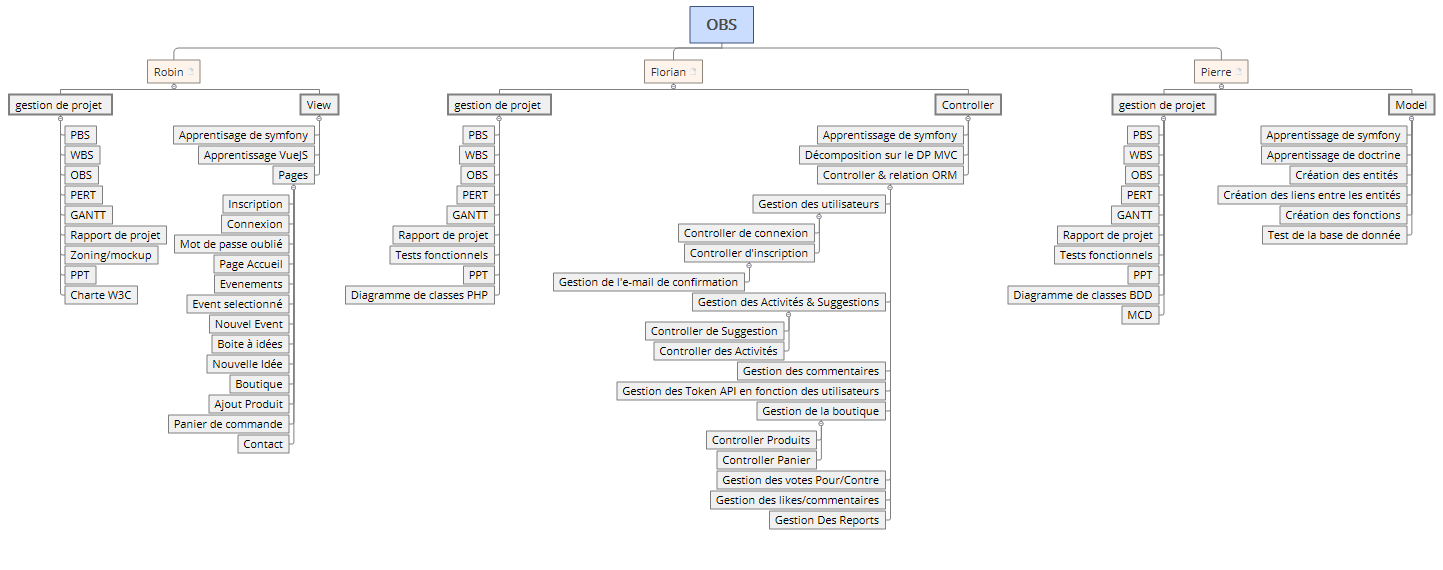
**Test : GivenMarketPlaceWhenLookingForShouldShowProducts**

Les membres du BDE souhaitent avoir un site vitrine attrayant, ergonomique et utilisant des technologies modernes pour une navigation fluide et homogène quel que soit le support utilisé (ordinateur, tablette ou smartphone). La charte graphique doit être cohérente avec celle du CESI, de l’EI ou de l’EXIA.

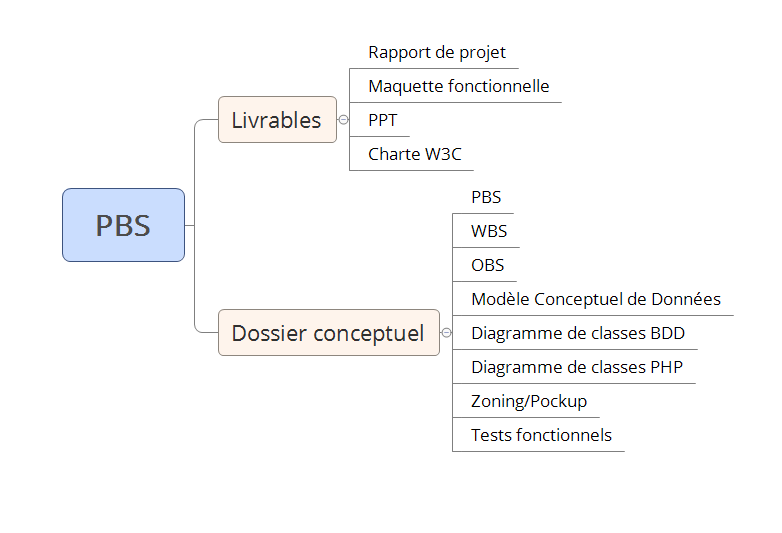
**Test : Visuel**

## Planning (A1)

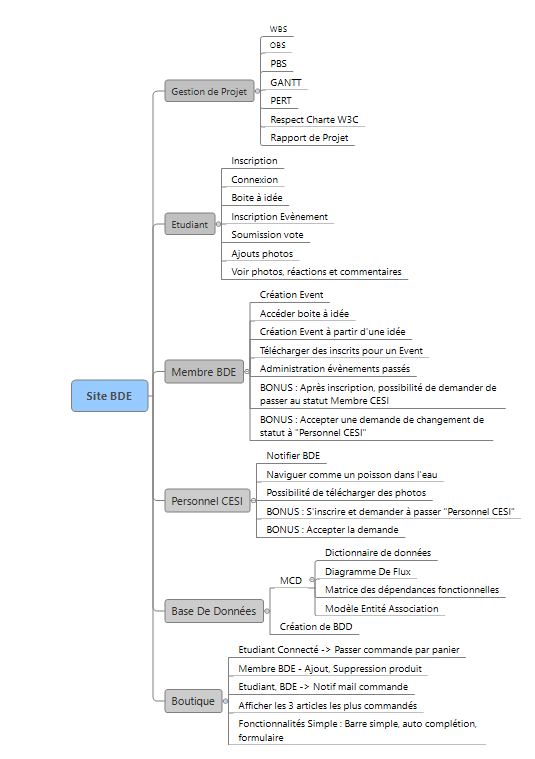
### OBS



### PBS



### WBS



### PERTT

**Voir le MS Project en Annexe**

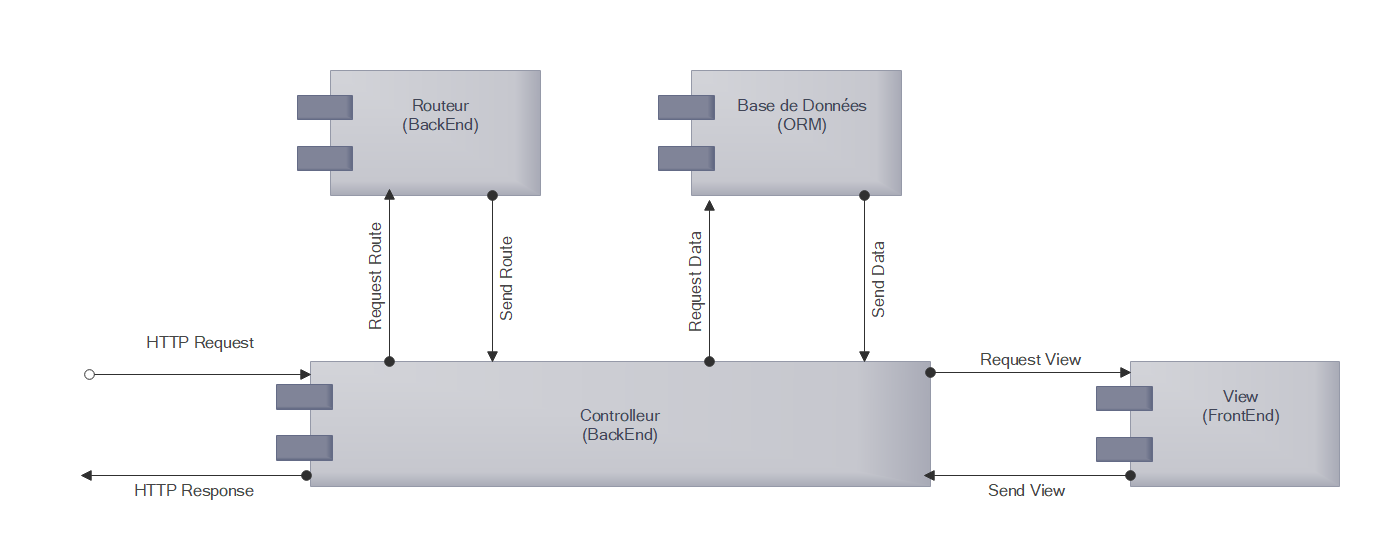
### GANT

**Voir le MS Project en Annexe**

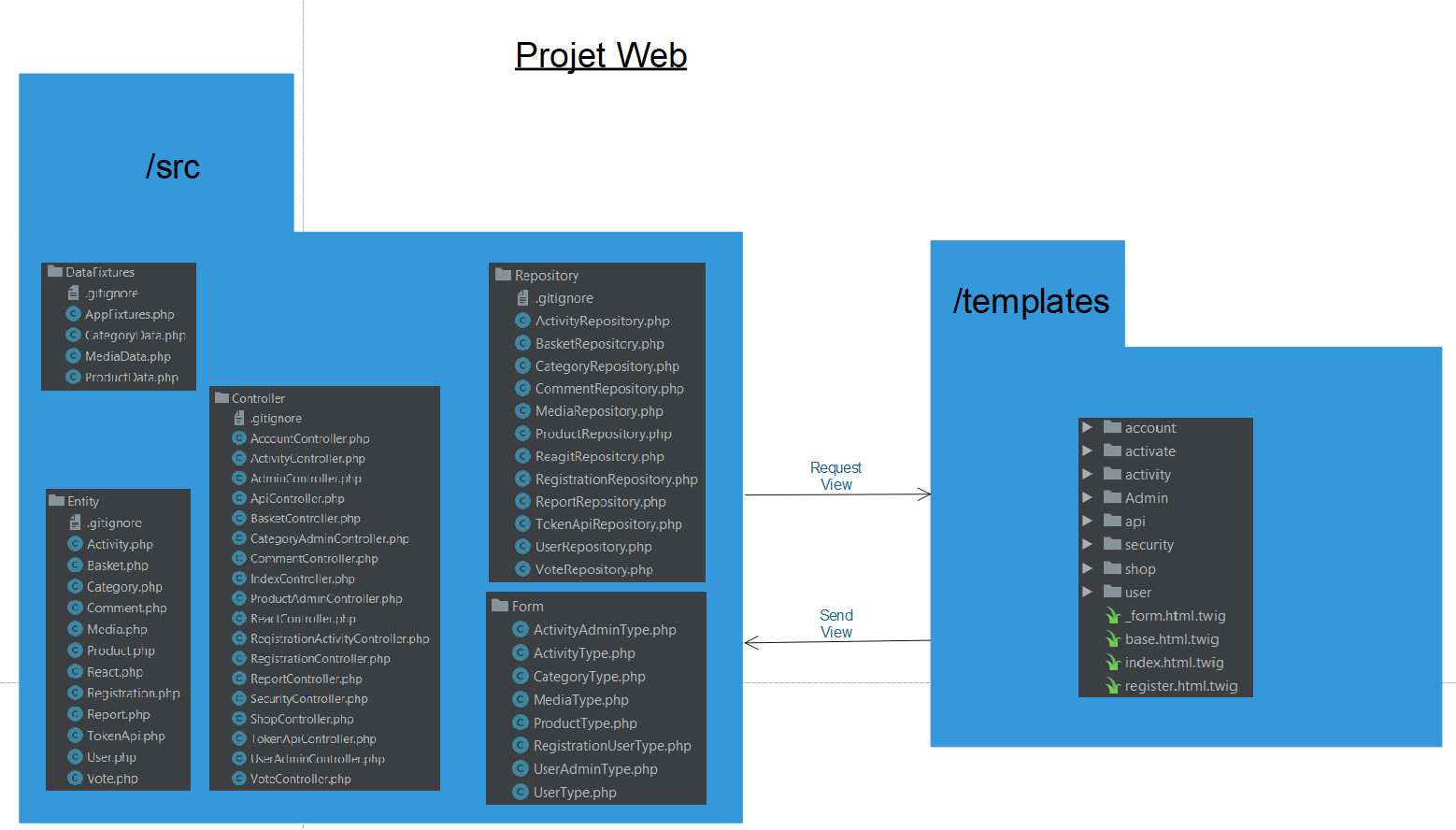
# Réalisation du projet

## Définition de l’architecture (A1)

### Diagramme de composant



### Diagramme de paquetage



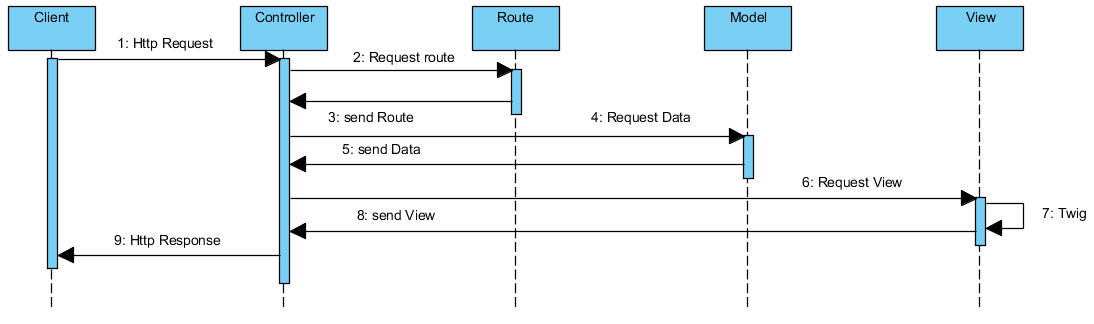
## Sprint Meeting (A repeter) (A1)

### Modélisation et création des tests

Diagramme de classe

**Voir Diagramme de classe en Annexe**

Diagramme de séquence



Création des jeux de tests unitaires

## Développement du code en fonction des cas de tests (A repeter) (A1)

Une branche par feature.

Pull request pour mise à jours de la branche principal.

# Clôture du projet

## Validation des tests fonctionnels (A1)

Tous les tests n’ont pas été validé lors de la soutenance.

## REX du projet (A1)

Ce projet à été un projet très intéressant à réaliser. Grâce à lui nous avons beaucoup appris sur le développement de site web et plus particulièrement sur le Framework « Symfony ». De plus nous avons acquis beaucoup de connaissances sur les Framework « Bootstrap » et « Doctrine ».

## Bilan (A1)

Bien qu’ayant beaucoup travaillé, nous n’avons pas pu mener a bien ce projet car nous avons rencontré beaucoup trop de difficultés sur les Framework « Symfony » et « Doctrine ». De plus l’assemblage de toutes les parties nous a pris trop de temps pour rendre une maquette complètement fonctionnelle.

# Annexes

## Outils

### Versionning

Github : https://github.com/FlorianVouters/ProjetWeb

### Gestion des exigences

Trello : https://trello.com/b/GaFBcEKh/projet-web

### Plannification

Project ou Trello