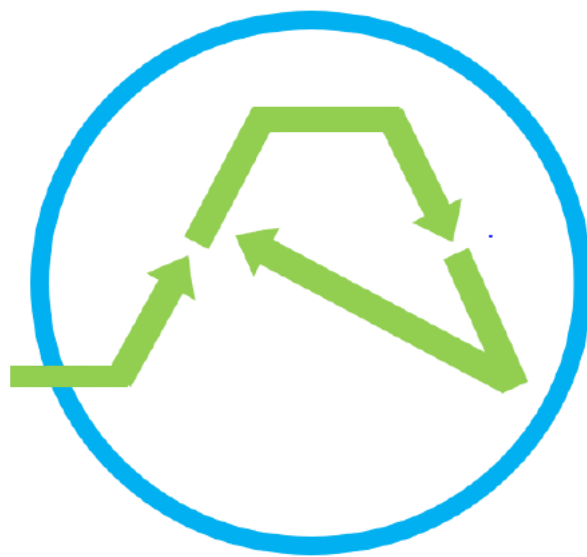


## Développement web



Association Avanza

Client/MOA : Association Avanza

Secteur d'activité : Écologie - Environnement

Prestataire/MOE : Yacine TAZDAIT

Secteur d'activité : Informatique - Développement web

08/10/2020

## Table des matières

I.	Présentation du projet	4
1.	Contexte	4
2.	Acteur du projet	4
3.	Objectif et raison du projet	4
4.	Les technologies utilisées	5
A.	Programmation et développement :	5
B.	API et service de paiement :	7
C.	Base de données	7
D.	Partage de données, communication et organisation	8
E.	Modélisation	8
F.	Edition et retouche d'image	9
II.	Développement	9
1.	Besoins fonctionnels	9
A.	Les non adhérents	9
B.	Les membres adhérents	10
C.	Les administrateurs	10
2.	Besoins non-fonctionnels	10
A.	Les utilisateurs :	10
B.	Les membres adhérents	11
C.	Les administrateurs	11
3.	La charte graphique	11
III.	Contraintes	11
1.	Contraintes techniques et organisationnelles	11
A.	Contraintes techniques	11
B.	Contraintes organisationnelles	11
2.	Contraintes temporelles	12
IV.	Gestion de projet	12
1.	Organisation	12
2.	Responsabilité	12
V.	Glossaire :	15

## Tables des illustrations

FIGURE 1- LOGO D'HTML5/CSS	5
FIGURE 2 - LOGO DE BOOTSTRAP	5
FIGURE 3 - ARCHITECTURE MVC	5
FIGURE 4 - LOGO DE PHP	5
FIGURE 5 - LOGO DE JAVASCRIPT ET DE JQUERY	6
FIGURE 6 - LOGO DE WEBSTORM	6
FIGURE 7 - LOGO DE PHPSTORM	6
FIGURE 8 - LOGO DE JETBRAINS	6
FIGURE 9 - LOGO DE WAMP SERVER	6
FIGURE 10 - LOGO D'APACHE	6
FIGURE 11 - LOGO D'HELLOASSO	7
FIGURE 12 - LOGO DE PHPMYADMIN ET DE MYSQL	7
FIGURE 13 - LOGO DE GIT	8
FIGURE 14 - LOGO DE SCRUM	8
FIGURE 15 - LOGO DE NOTION	8
FIGURE 16 - LOGO DE WHATSAPP	8
FIGURE 17 - LOGO DE GIMP	9
FIGURE 18 - LOGO DE POWER AMC	9

# I. Présentation du projet

## 1. Contexte

Le projet Avanza est une application web développée par un étudiant de l'université Claude Bernard Lyon 1 en étroite collaboration avec l'association Avanza. Celle-ci œuvre pour la préservation de notre environnement.

Ce projet émane principalement de la volonté de contribuer à l'écologie. Il permettra également au développeur de retrouver un cadre de travail identique à celui qu'il retrouvera dans le monde du travail avec un client et un prestataire. Ainsi, cela sera l'occasion de croiser le domaine de formation de l'étudiant avec une thématique qui est véritable enjeu aujourd'hui.

## 2. Acteur du projet

Comme nous l'avons mentionné, nous pourrions compter sur un étudiant en informatique. Il sera perçu comme le **maître d'œuvre (MOE)**. En d'autres termes, il s'agit du prestataire et responsable du développement de la plateforme web. Il disposera d'une certaine liberté dans les choix techniques mais aussi organisationnels.

En revanche, il devra prendre uniquement en compte les besoins des clients en faisant la liaison entre la partie développement et analyse du projet. De ce fait, ce dernier sera en contact permanent avec le **maître d'ouvrage (MOA)**. Il s'agit client final représenté par les membres d'Avanza. Leur rôle sera d'exprimer les besoins de l'association et de les partager avec le développeur. Ainsi, il devra les recueillir et les développer sur la **plateforme**. Par ailleurs, ces derniers participeront également aux phases de **test utilisateur**. Cela permettra au client d'avoir plusieurs versions du produit et de suivre l'avancée du projet. Ils pourront alors faire part de leurs avis et adopter leurs demandes au cours du développement.

À noter que le MOA ne sera en aucun cas attendu pour des missions en informatique. Aucune compétence technique ne sera demandée. Il faudra simplement qu'il fasse appel à leurs bons sens ainsi qu'à leurs capacités d'analyse. De plus, les tâches qui leurs seront attribuées seront secondaires et viendront enrichir l'**application web** via l'intégration de contenu (articles, images, etc.).

## 3. Objectif et raison du projet

L'application sera développée dans un cadre associatif et n'aura aucun but lucratif. En clair, elle permettra de sensibiliser davantage d'étudiants et de mener à bien ses actions pour la protection ainsi que la préservation de l'environnement.

De plus, elle permet créer un outil qui centralise les informations (news, événement etc.) ainsi que les **opérations** qu'Avanza réalise. Ainsi, nous pourrions réaliser un gain de temps et réduire l'usage de certains matériaux nuisibles à l'environnement tels que le papier, le carton ou encore le plastique.

D'un point de vue personnel, j'ai pu remarquer que la majorité de mes candidatures pour des emplois de diverses natures étaient rejetées. Aux yeux des employeurs, je manquais d'habileté dans les **technologies** qui étaient mentionnées dans les offres qui m'étaient parvenues. De ce fait, ce projet est l'occasion de monter en compétences, notamment en programmation ainsi qu'en développement web. Enfin, cela me permettra de justifier de certains projets personnels que je pourrais mettre en avant lors de mes entretiens.

## 4. Les technologies utilisées

### A. Programmation et développement :

#### a) HTML5/CSS3 :

Afin de garantir un design **ergonome** et intuitif, nous avons misé sur des outils que nous connaissons bien. HTML est un **langage balisé** permettant de représenter une application web.

L'HTML nous permettra de créer le fond de notre application. C'est grâce à cette technologie que nous pourrions afficher nos données et les structurer.

Le CSS sont des **feuilles de style** nous permettant de rendre l'application plus séduisante. Pour garantir une expérience agréable, le CSS sera utilisé pour la mise en forme des données.



Figure 1- Logo d'HTML5/CSS

#### b) Bootstrap :

HTML et CSS ne sont pas les seuls outils auxquels nous aurons recours pour la mise en forme des données. En effet, nous aurons également besoin de Bootstrap. Ce framework offre un bagage de composants et d'**éléments graphiques**. Elle permet également d'utiliser du JavaScript ainsi que du JQuery pour enrichir notre application web.

Dans le cadre de ce projet, nous utiliserons seulement Bootstrap pour améliorer le design CSS et enrichir le site web via du Javascript.



Figure 2 - Logo de Bootstrap

#### c) PHP – MVC :

PHP est un **langage de programmation orientée côté serveur**. Il permettra d'implémenter la **couche métier** de notre application (**traitement** et **calcul** avec la **base de données** etc.).

Par ailleurs, ce site sera développé à l'aide du modèle MVC. Il s'agit d'une architecture permettant d'organiser un code à travers 3 couches :

- Le modèle (ou couche métier/logique) : la couche métier qui permet de traiter les données (requête SQL ...).
- La vue : Elle permet de restituer les données en les mettant en forme.
- Contrôleur : Cette couche permet de faire la liaison entre la vue et le modèle



Figure 4 - Logo de PHP

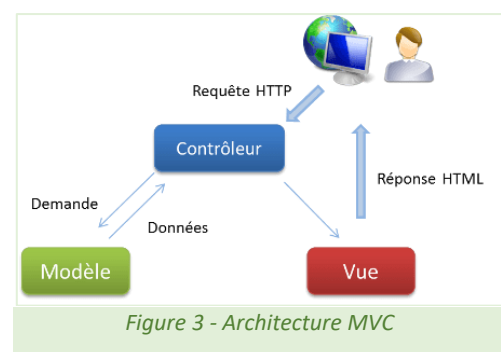


Figure 3 - Architecture MVC

d) JavaScript/JQuery :

Contrairement à PHP, JavaScript est un **langage de programmation orienté côté client** qui permet d'enrichir une page web en créant des animations.

Au fil du temps, de nombreuses extensions et frameworks ont été développés en se basant sur ce langage. Parmi ceux-ci, on peut citer JQuery qui possède une syntaxe différente du Javascript natif et permet de réaliser des animations et des interactions avec l'utilisateur.



Figure 5 - Logo de JavaScript et de JQuery

Nous utiliserons du Javascript ainsi que du JQuery pour donner une forme plus séduisante à notre application.

e) PHPStorm / WebStorm :

Ce sont des **IDE(s)** créés par la société Jet Brain. Le premier est, comme son nom l'indique, destiné à l'implémentation du **backend** en PHP tandis que le second sera utilisé pour du **frontend**, soit pour le développement de l'interface utilisateur en Javascript. Il supporte l'ensemble des frameworks construits en se basant sur ce langage.

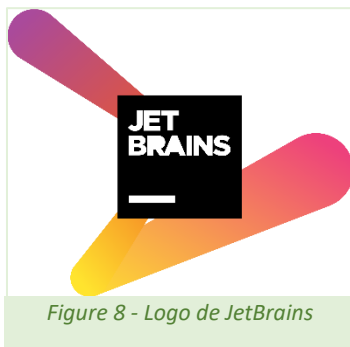


Figure 8 - Logo de JetBrains



Figure 7 - Logo de PHPStorm



Figure 6 - Logo de WebStorm

f) WampServer :

Ce serveur est installé en même temps que PHP, MySQL et apache via l'installation de WAMPServer. WampServer est une plateforme de développement Web de type WAMP, permettant de faire fonctionner localement des **scripts** PHP. WampServer n'est pas en soi un logiciel, mais un environnement comprenant trois serveurs, un **interpréteur** de script, ainsi que phpMyAdmin pour l'administration Web des **bases de données** MySQL.



Figure 9 - Logo de WampServer

Il sera utilisé pour réaliser faire tourner notre application en local et créer notre base de données.

g) Apache :

Apache est un **serveur web** permettant de faire tourner des applications web en local. Ces applications sont développées en PHP. Ainsi, en interprétant le code source pour l'afficher sur le navigateur, il permet de visualiser et corriger si nécessaire le rendu final. Il sera utilisé pour effectuer les tests de l'application.



Figure 10 - Logo d'Apache

## B. API et service de paiement :

### a) HelloAsso :

HelloAsso est une plateforme fournissant une multitude de services. Ceux-ci sont exclusivement réservés aux associations inscrites sur le site « helloasso.com ».

HelloAsso nous sera utile pour la mise en vente des paniers en ligne car elle nous procurera un service de paiement en ligne.



## C. Base de données

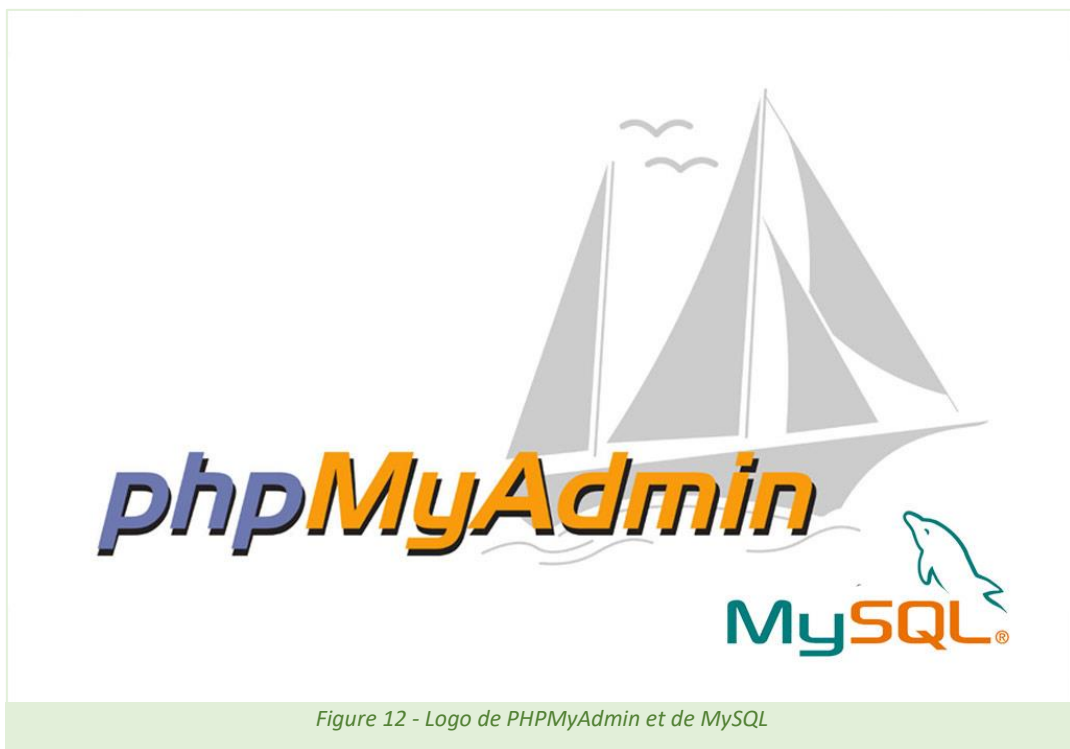
### a) MySQL :

MySQL est **SGBD (système de gestion de base de données) relationnelle** permettant de programmer et d'administrer des bases de données en **SQL**.

Ce SGBD nous permettra de créer notre base et de l'interroger pour restituer des données. Dans un premier temps, nous créerons une base de données relationnelle. Dans un second, si la quantité de données est trop importante et ralentit l'application alors nous migrerons vers du NoSQL. Celui-ci est un SGBD différent du modèle relationnel. Il est connu pour être plus performant et adapte ses fonctionnalités et ses performances en fonction de la quantité de données traitées.

### b) PhpMyAdmin :

PhpMyAdmin est une plateforme distribuée sous la licence **GNU/linux**. Elle permet d'administrer des bases de données MySQL. Notre base de données sera principalement **hébergée** sur cette plateforme puis mise en ligne lors de la mise en production.



## D. Partage de données, communication et organisation

### a) Git :

Git un logiciel de **gestion de version** développé et distribué sous la licence GNU/Linux. Il sera utile pour effectuer des sauvegardes du projet mais aussi pour retrouver des anciennes données que l'on souhaite réutiliser.



Figure 13 - Logo de Git

### b) Agile/ méthode Scrum :

Les méthodes agiles sont des groupes de pratiques de pilotage et de réalisation de projets.

Scrum est considéré comme un cadre ou « framework » de gestion de projet. Il permet d'organiser le projet de sorte à ce que le client puisse suivre son avancé et effectuer des modifications au cours du projet s'il le juge nécessaire.



Figure 14 - Logo de Scrum

### c) Notion :

Notion est une application qui fournit des composants tels que des bases de données, des **tableaux kanban**, des **wikis**, des calendriers et des rappels.

Cet outil nous permettra d'organiser et de prioriser nos tâches via des "To do list" et des tableaux Kanban. Ils permettront également de mettre en place un système de suivi des objectifs atteints tout au long du projet. Notion nous offre également un système de suivi des anomalies après avoir effectué des tests utilisateurs.



Figure 15 - Logo de Notion

### d) WhatsApp :

WhatsApp est une **application mobile multiplateforme** qui fournit un système de messagerie instantanée chiffrée de bout en bout aussi bien par Internet que par les réseaux téléphonie mobiles. WhatsApp a remporté un grand succès au tournant des années 2010.

Cette application nous permettra de communiquer et de partager des informations via le service de messagerie.



Figure 16 - Logo de WhatsApp

## E. Modélisation

### a) PowerAMC :

Il s'agit d'un logiciel de **modélisation** permettant de créer des diagrammes UML mais aussi toute sorte de **schéma** permettant de synthétiser les fonctionnalités de l'application. Un diagramme UML est un langage de modélisation graphique à base de pictogrammes conçu pour fournir une méthode normalisée pour visualiser la conception d'un système.

Power AMC sera donc utilisé lors de la phase d'analyse pour modéliser les problèmes qui englobent cette application et fournir des solutions.



## F. Edition et retouche d'image

### b) Gimp :

Gimp est un logiciel d'édition et de retouche d'image libre et gratuit. Il sera utilisé pour revoir créer des ou modifier des contenus graphiques.



Figure 18 - Logo de Power AMC



Figure 17 - Logo de Gimp

## II. Développement

### 1. Besoins fonctionnels

#### A. Les non adhérents

Bien que cette application web soit destinée au grand public, on distingue 3 types d'utilisateurs. En fonction du type d'utilisateur, des privilèges leur seront attribués.

Dans un premier temps, nous retrouverons les utilisateurs **non adhérents**. Il s'agit des personnes n'ayant pas adhéré à l'association. Ils disposeront de fonctionnalités classiques et accessibles à tous. Ces utilisateurs regroupent l'ensemble des tierces personnes ou bien des membres de l'association. Ainsi, ces derniers pourront consulter une page d'accueil où seront postées les annonces. En première page on retrouvera l'annonce du jour qui recouvrera l'ensemble ainsi que celles postées à des dates antérieures.

De même, les articles, les événements ainsi que les recettes seront consultables via la plateforme. Il ne leur sera pas possible de les modifier ou bien de les commenter car seuls les utilisateurs authentifiés peuvent effectuer ce type d'opération.

Les **ateliers** proposés par l'association sont également accessibles. Plus encore, il ne faut pas nécessairement que l'utilisateur soit connecté pour qu'ils puissent y participer. Il suffira de sélectionner l'atelier auquel le visiteur souhaite participer et de réserver une place. Pour chaque atelier, un nombre limité de places et un créneau seront attribués par les administrateurs.

Enfin, ils pourront adhérer à l'association en s'inscrivant en ligne. Pour ce faire, l'utilisateur doit remplir un formulaire avec ses informations personnelles. L'inscription est envoyée pour que les **administrateurs** vérifient si les informations saisies sont conformes (propos choquants, etc.). Si le compte est validé un mail est envoyé à la personne concernée avec un lien vers le compte « HelloAsso » d'Avanza pour réaliser un don de 3€. Cette contribution sera obligatoire afin d'officialiser la venue d'un nouveau membre en activant son compte. Elle montre son engagement et son dévouement au sein de l'association. En revanche, si l'inscription est rejetée, alors un mail sera envoyé l'utilisateur pour revoir son inscription avec le motif du refus.

## B. Les membres adhérents

Comme nous l'avons mentionné, les fonctionnalités énumérées précédemment resteront accessibles à tous. Cependant, d'autres opérations ne pourront être réalisées que par des membres de l'association. Ce sont des personnes (étudiants, etc.) faisant partie de l'association. Contrairement aux utilisateurs classiques, les **adhérents** pourront, rédiger et poster des **articles** ainsi que des **événements** (soirées d'intégrations, ateliers, présentations etc ...).

Ils auront également la possibilité de composer et poster des **recettes**.

N.B : Le contenu publié par d'autres membres pourra être commenté uniquement si l'utilisateur est authentifié.

Enfin, un utilisateur pourra recevoir des **newsletters** qui l'informeront des dernières informations relatives à l'association.

## C. Les administrateurs

Les **administrateurs** étant des membres d'Avanza, ils bénéficieront des mêmes droits que les autres utilisateurs (utilisateurs classiques et membres adhérents). Toutefois, ils auront un degré de **privilège** plus élevé sur l'application. De ce fait, ils pourront effectuer des opérations que les d'autres utilisateurs ne pourront réaliser.

Dans un premier temps, ces derniers pourront gérer les inscriptions des membres adhérents. Comme nous l'avons évoqué précédemment, ils auront le droit d'accepter ou de rejeter une candidature.

De même, les administrateurs auront également la main mise et pourront vérifier le contenu soumis par les membres avant qu'il ne soit posté. Une fois que ces derniers auront validé le **contenu**, il sera publié. Un mail sera envoyé à l'auteur du contenu afin de l'informer que celui-ci est bien en ligne. Dans le cas contraire, un mail lui sera également envoyé pour lui demander de revoir son contenu. Les contenus pourront être modifiés ou supprimés à tout moment par les administrateurs du site dès lors qu'ils seront postés.

Certains **événements** comme les AGs (assemblée générale) seront organisées par ces-derniers.

Les **annonces** seront rédigées, vérifiées et mises en ligne uniquement par les administrateurs.

Enfin, les **paniers** seront composés et mis en vente sur l'application par les administrateurs.

## 2. Besoins non-fonctionnels

### A. Les utilisateurs :

Au-delà des **opérations** évoquées dans la dernière partie, les utilisateurs auront la possibilité d'accéder au CQFD ainsi qu'au page Facebook et Instagram d'Avanza.

Les paniers composés par les administrateurs de l'association seront disponibles sur la plateforme afin de les consulter mais aussi d'en réserver un seul par personne.

## B. Les membres adhérents

Les membres pourront gérer leurs informations personnelles dans les paramètres tout comme ils pourront supprimer leurs comptes pour signifier leurs départs de l'association. Dès que le compte sera supprimé un email de confirmation sera envoyé à la personne concernée pour lui attester que son compte a été supprimé. Un email sera également envoyé aux administrateurs pour les informer du départ d'un membre. Ainsi, tous les contenus publiés par cette personne seront également supprimés.

Ces opérations ne nécessitent aucune demande au préalable auprès de l'administrateur.

## C. Les administrateurs

Les administrateurs pourront gérer les comptes des membres adhérents sans autorisation. Ils auront le droit de supprimer le compte d'un membre pour une tierce raison (renvoie etc.). De même, ils pourront augmenter les privilèges d'un utilisateur en lui donnant le statut d'administrateur.

Enfin, ces derniers seront responsables de la création des **newsletters** qui seront envoyées sur la boîte mail des adhérents.

## 3. La charte graphique

La **charte graphique** de cette application reprendra essentiellement les couleurs de l'association Avanza. En effet, les couleurs ainsi que la thématique de cette plateforme seront bleu clair, vert clair et blanc. Des couleurs froides et en relation avec l'écologie seront également ajoutés pour être en adéquation avec le projet.

# III. Contraintes

## 1. Contraintes techniques et organisationnelles

### A. Contraintes techniques

La principale **contrainte technique** sera principalement due au temps de formation sur les nouvelles technologies qui seront employées sur le projet. En effet, il faudra compter au total 1 mois de formation sur le **design pattern** MVC en PHP ainsi que sur du JQuery. De plus, les **bugs** et **anomalies** seront des éléments qui pourront freiner le projet au fur et à mesure qu'ils seront complexes.

### B. Contraintes organisationnelles

Étant en année de césure pour des raisons qui me sont propres, j'aurais davantage de temps à consacrer à ce projet. Toutefois, mon emploi ainsi que les autres activités extra-professionnelles (sport, cours de langue etc.) représentent des contraintes qui m'obligeront à répartir mon temps personnel afin de mener à bien ce projet.

Les périodes de partiels sont à prendre en compte dans l'organisation du projet. En effet, les membres de l'équipe Avanza seront parfois absents pour des raisons scolaires personnelles. De plus, il se peut que ces derniers ne soient pas sur Lyon lors des vacances ou bien pendant les weekends.

Enfin, la crise sanitaire que nous traversons est également un frein pour l'avancée du projet. Les transports peuvent se retrouver bloqués pour ceux qui sont en dehors de Lyon et nous obliger à utiliser d'autres canaux de communication comme hatsapp ou encore Discord.

## 2. Contraintes temporelles

Pour réaliser ce projet nous partons sur une base de 12 mois. En prenant en compte les différentes contraintes évoquées dans les paragraphes précédents ainsi que les éventuels événements inattendus (malades, congé etc.) il se peut que nous dépassions la durée totale du projet. Ainsi, nous nous laissons finalement une marge de manœuvre de 14 mois maximum afin de livrer le produit final.

À l'issue de la phase de développement, une **maintenance** sera effectuée tout au long de l'année qui suit afin de pouvoir régler les anomalies qui seront rapportées par le client.

Par ailleurs, il se peut qu'au fil du temps certaines **fonctionnalités** ne soient pas revues sans pour autant comporter des bugs. Il s'agira simplement de les revoir afin d'améliorer l'application et de la rendre plus **optimale** (**migration** vers un SGBD plus performant, insertion de fonctionnalités supplémentaires etc.).

## IV. Gestion de projet

### 1. Organisation

Le système organisationnel du projet sera basé essentiellement autour de la méthode Scrum. En effet, le développeur disposera de plusieurs périodes délimitées sur toute la durée du projet que l'on nomme des « **sprints** » (cf voir l'annexe A). Pendant ces sprints, il sera amené à développer les fonctionnalités du projet.

À l'issue d'une période, une « mêlée » avec l'ensemble des membres d'Avanza sera organisée. Il s'agit d'une réunion où une présentation du travail effectué ainsi que des tests utilisateurs seront réalisés. Dans un premier temps, cela permettra de faire un point sur l'avancée du projet. Dans un second, temps, je pourrais revoir certains points et aussi ajuster l'application en fonction de la demande de l'utilisateur. Enfin, je pourrai proposer et confronter mes idées avec l'ensemble de l'équipe pour que l'on soit d'accord sur certains choix. Ainsi, l'ensemble des membres de l'équipe se sentiront davantage concernés et proches du travail réalisé en amont.

### 2. Responsabilité

La charge de travail de la partie développement sera exclusivement réservée au développeur. Il sera responsable de :

- La phase d'analyse
- La phase de conception de l'application dans son intégralité
  - Base de données
  - Backend
  - Frontend
- La phase de **déploiement** de l'application
- La phase de maintenance

D'autres missions seront confiées à l'équipe Avanza. Ces derniers pourront être amenés à réaliser des tâches secondaires :

- Retouche d'images et réalisations graphiques
- Rédaction et insertion de contenus (articles, news etc ...)
- Promotion de l'application
- Tester de l'application
- Rapporter les bugs et autres au développeur

Ancêtre	Tâche	Libellés	Durée (en jours)
Début du projet – semaine du 01/10/2020			
I - Phase de formation – partie 1			
-	A	Formation en PHP/MVC	7
I - Phase d'analyse			
A	B	Réaliser le schéma entité/association	2
A	C	Réaliser le modèle relationnel	1
A	D	Réaliser les diagrammes UML	3
II – Phase de développement – partie 1 (Besoin fonctionnel)			
C, D	E	Implémenter le schéma de base de données	0,1
E	F	Connexion à la base de données et réalisation des requêtes SQL permettant d'interroger la base de données	7
F	G	<b>Implémentation</b> de la connexion utilisateur avec le système de gestion des droits d'accès	7
Présentation du travail effectuée - semaine du 10/11/2020			
G	H	Implémentations des fonctionnalités des utilisateurs non-adhérents à l'association	7
H	I	Implémentations des fonctionnalités des utilisateurs adhérents à l'association	30
Test des fonctionnalités primaires 1 - semaine 17/12/2021			
I	J	Implémentations des fonctionnalités des administrateurs	50
Test des fonctionnalités primaires 2 - semaine 03/02/2021			
III – Phase de développement – partie 2 (Besoin non-fonctionnel)			
J	K	Implémentations des fonctionnalités secondaires des utilisateurs non-adhérents à l'association	30
Test des fonctionnalités secondaire 1 - semaine 05/03/2021			
K	L	Implémentations des fonctionnalités secondaires des utilisateurs adhérents à l'association	60
Test des fonctionnalités secondaire 2 - semaine 03/06/2021			
L	M	Implémentations des fonctionnalités secondaires des administrateurs	40
Test des fonctionnalités secondaires 10/09/2021			
IV - Phase de formation - partie 2			
M	N	Formation en Javascript/JQuery	0,5
V - Phase développement – partie 3			
N	O	Rédiger le contenu statique de la plateforme (page à propos, etc ...)	5
O	P	Réaliser la mise en forme	40
Tests utilisateurs - semaine 30/09/2021			
VI - Phase développement – partie 4			
P	Q	Gestion des bugs et des problèmes de maintenance	30
Livraison – mi-octobre			
Durée totale			348,5

## V. Glossaire :

Ce glossaire a pour objectif de fournir une définition aux termes ainsi qu'aux expressions qui se révéleraient complexes dans ce cahier des charges. Il permet ainsi aux lecteurs de mieux le comprendre et de se situer au fur et à mesure qu'il prend connaissance de ce document.

**Adhérent** : Membre de l'association Avanza disposant de privilèges plus élevés que les non adhérents mais moins important que les administrateurs.

**Administrateur** : Membre de l'association Avanza disposant de privilèges plus élevés que les autres types utilisateurs (non adhérents et adhérents).

**Annonces** : Décisions, réformes et autres informations importantes permettant de changer le fonctionnement et l'organisation d'Avanza.

**Application mobile multiplateforme** : Application mobile disponible sur IOS et Android.

**Application web** : Site web qui permet à ses utilisateurs d'accomplir des tâches spécifiques. Une application gère donc généralement des utilisateurs et toutes sortes de données selon les requis spécifiques au projet.

**Article** : Contenu textuel, portant un sujet spécifique et publié sur l'application web afin de tenir au courant les lecteurs des informations relatifs à l'écologie.

**Atelier** : Événement ouvert à tous et proposant des activités manuelles et ludiques sur la thématique de l'environnement afin de promouvoir l'action d'Avanza.

**Backend (ou couche logique/métier)** : Partie du code qui exécuté par le serveur avant de les renvoyer sur la page web de l'utilisateur. Il s'agit également de la partie permettant de rendre le site web dynamique en réalisant une multitude d'opérations et de calculs en amont avec la base de données.

**Base de données** : Structure permettant de stocker et de retrouver l'intégralité des données brutes ou d'informations en rapport avec un thème ou une activité ; celles-ci peuvent être de natures différentes et plus ou moins reliées entre elles

**Modèle relationnel** : Manière de modéliser les relations existantes entre plusieurs informations, et de les ordonner entre elles.

**Bug/anomalie** : Défaut de conception d'un programme informatique à l'origine d'un dysfonctionnement.

**Charte graphique** : Document de travail qui contient l'ensemble des règles fondamentales d'utilisation des signes graphiques qui constituent l'identité graphique d'une organisation, d'un projet, d'une entreprise.

**Contenu** : Ensemble des articles, événements, annonces, recettes et paniers qui sont destinés à être publiés sur le site.

**Contrainte technique** : Limitations techniques (formation dans un langage par exemple) entraînant un retard du projet.

**Design pattern (ou patron de conception)** : Architecture modulaire conçue et reconnue pour être une bonne pratique afin de solutionner un problème donné.

**Déploiement :** Fait d'introduire une nouvelle solution/plateforme technique et des services dans une organisation d'une manière coordonnée.

**Élément graphique :** Ensemble de dessins, caractères typographiques, photos, couleurs, etc. utilisé pour réaliser le design d'une application.

**Ergonomie :** Branche de l'ergonomie, qui a pour objectif d'améliorer les interactions homme-machine, la facilité d'utilisation et d'apprentissage des produits interactifs.

**Événements :** Contenu posté sur l'application permettant d'informer un visiteur qu'un fait à l'intention d'être organisé (fête, soirée d'intégration etc.).

**Feuille de style :** Langage informatique qui décrit la présentation des documents HTML et XML.

**Framework :** Ensemble cohérent de composants logiciels structurels, qui sert à créer les fondations ainsi que les grandes lignes de tout ou d'une partie d'un logiciel.

**Frontend :** Frontend désigne les éléments d'un site que l'on voit à l'écran et avec lesquels on peut interagir depuis un navigateur.

**Gestionnaire de version :** Logiciel qui permet de stocker un ensemble de fichiers en conservant la chronologie de toutes les modifications qui ont été effectuées dessus.

**GNU/Linux :** Système d'exploitation permettant d'exécuter plusieurs autres applications simultanément.

**Hébergeur :** Espace de stockage en ligne.

**IDE (ou environnement de développement) :** Ensemble d'outils qui permet d'augmenter la productivité des programmeurs qui développent des applications.

**Implémentation :** Mise en place sur un ordinateur d'un système d'exploitation ou d'une application adaptée aux besoins et à la configuration informatique de l'utilisateur.

**Interpréteur de script :** Outil ayant pour tâche d'analyser, de traduire et d'exécuter les programmes écrits dans un langage informatique.

**Langage balisé :** Langage informatique le plus couramment utilisé en informatique afin de préciser à l'intérieur d'un fichier texte des éléments tels que les titres, les paragraphes, les acronymes, les citations, etc.

**Langage de programmation orientée côté client :** La programmation "côté client" (ou développement web frontal) consiste à produire du code (HTML, CSS, Javascript, ...) qui sera interprété par un navigateur (ou une fenêtre, pour les logiciels de bureau ou applications mobiles) et produira un rendu visuel : une interface utilisateur (UI), et dont certains éléments permettront une navigation, des liens, avec les autres pages (UX).

**Langage de programmation orientée côté serveur :** Un langage serveur, ou plus précisément un langage de script côté serveur (de l'anglais : server-side scripting) est un langage de programmation mis en œuvre sur un serveur HTTP pour produire une page Web dynamique.

**Maintenance :** La maintenance informatique regroupe toutes les tâches et actions nécessaires au bon fonctionnement d'un système informatique.

**Maître d'œuvre (MOE) :** Personne qui a pour mission de faire réaliser l'ouvrage prévu dans des conditions de coût et de délais préalablement définis.



**Maître d'ouvrage (MOA) :** Personne pour qui est réalisé le projet. Elle est l'entité porteuse d'un besoin, définissant l'objectif d'un projet, son calendrier et le budget consacré à ce projet.

**Migration :** Transformation de données, de programmes ou de logiciels afin de les rendre compatibles avec un autre environnement informatique.

**Modélisation :** Création de modèle permettant de représenter les solutions face à un problème via une syntaxe bien précise.

**Mêlée :** Réunion quotidienne de moins de 15 minutes permettant à l'équipe de développement de se synchroniser, identifier les obstacles éventuels et mesurer son avancement sur le Sprint en cours.

**Newsletter :** Lettre d'information périodique d'un site web.

**Non-adhérent :** Personne physique issue du grand public, n'ayant pas adhéré à l'association et possédant le degré de privilège le plus bas par rapport aux autres utilisateurs sur l'application.

**Optimisation :** Amélioration des performances d'une application.

**Opérations :** Fonctionnalités permettant à l'utilisateur de réaliser une action sur une application ou un système d'exploitation.

**Panier :** Fruits et légumes qui seront vendus à hauteur d'un prix compris entre 8 et 10 €.

**Plateforme web :** Environnement permettant la gestion et/ou l'utilisation de services applicatifs.

**Privilège/droit d'accès :** Délégation d'autorité sur un système informatique. Un privilège permet à un utilisateur d'effectuer une action.

**Recette :** Ensemble d'instructions permettant de la réalisation d'un plat posté sur l'application.

**Schéma entité/association :** Modèle de données ou diagramme pour des descriptions de haut niveau de modèles conceptuels de données. Il permet de modéliser l'architecture d'une base de données en restant cohérent avec la réalité.

**Schéma de base de données :** Structure permettant de décrire l'organisation des données au sein d'une base de données.

**Script :** Programme (ou un bout de programme) chargé d'exécuter une action prédéfinie quand un utilisateur réalise une action ou qu'une page web est en cours d'affichage sur un écran.

**Serveur web :** Un serveur web est, soit un logiciel de service de ressources web, soit un serveur informatique qui répond à des requêtes du World Wide Web sur un réseau public ou privé, en utilisant principalement le protocole HTTP.

**Site web :** Ensemble de pages web et de ressources reliées par des hyperliens, défini et accessible par une adresse web.

**Sprint (ou itération) :** Intervalle de temps court, pendant lequel l'équipe de développement va concevoir, réaliser et tester de nouvelles fonctionnalités.

**Système de gestion de base de données (ou SGBD) :** Logiciel informatique permettant le stockage, la consultation, la mise à jour, la structuration ou encore le partage d'informations dans une base de données.

**Kanban** : Méthode basée sur le management visuel des tâches. On appelle aussi « kanban » (avec un 'k' minuscule), le tableau des tâches utilisé pour suivre visuellement l'activité du Sprint.

**Technologies** : Ensembles des logiciels, langages et autres ressources qui seront déployés afin de pouvoir créer l'application.

**Test utilisateur** : Méthode permettant d'évaluer un produit en le faisant tester par des utilisateurs.

**Traitement/calcul** : Processus qui permettent d'extraire de l'information ou de produire du savoir à partir de données brutes. Ces processus, une fois programmés, sont le plus souvent automatisés à l'aide d'ordinateurs.

**Wikis** : Site web dynamique qui a pour principale particularité de laisser chaque utilisateur modifier son contenu. En d'autres termes, il s'agit, d'un site web collaboratif.