



Refonte du logiciel du foyer des élèves

Rapport de projet

Alyacoubi Reda
Brault Clément
Champion Antoine
Orain Alexandre

État de l'art

Le foyer des élèves est au centre de notre vie aux Arts et Métiers.

L'objectif de ce projet est d'améliorer sa gestion en effectuant une refonte du logiciel actuellement utilisé.

Actuellement, ce logiciel permet seulement de décompter les consommations des personnes sur leurs comptes, il permet également d'avoir un historique très sommaire des consommations et l'argent restant sur les comptes.

En l'état, il s'agit d'une interface front-end liée à quelques macros Access, le logiciel est très basique et ne permet pas d'avoir une gestion des stocks précise, ce qui est pénalisant pour la comptabilité.

L'idée serait donc de créer un logiciel qui permettrait une gestion des stocks et des approvisionnements automatisée et poussée, mais aussi avec une interface web adaptée aux ordinateurs et aux smartphones avec la possibilité pour un utilisateur de recharger son compte et de consulter son historique, et pour les administrateurs, avoir accès à un panneau de contrôle complet.

I – Organisation et gestion de projet

Afin de cadrer le projet et de ne pas partir en aveugle, nous avons utilisé plusieurs séances afin de définir la portée du projet à l'aide de différents outils.

Notre premier objectif était de définir la portée du projet, les évolutions par rapport au logiciel précédent et de mettre en lumière des liens entre les idées des différents membres. Nous avons donc réalisé une Mind Map (carte des idées).

Dans la continuité de cette Mind Map, nous avons réalisé une liste exhaustive des fonctions que notre plateforme allait offrir. En parallèle, nous avons soumis nos idées à la promotion et réalisé des sondages afin d'avoir le plus de retours possibles.

Nous leur avons soumis deux questions, et voici les réponses qui revenaient le plus souvent :

Q : Que faudrait-il changer dans le système actuel ?

R : Davantage de précisions sur l'historique personnel de consommations (type de produit, quantité, date)

Recharge du compte autonome et en ligne

Changement du design de l'interface utilisateur

Fusion avec compte Borgia

Protection du compte par mot de passe personnel

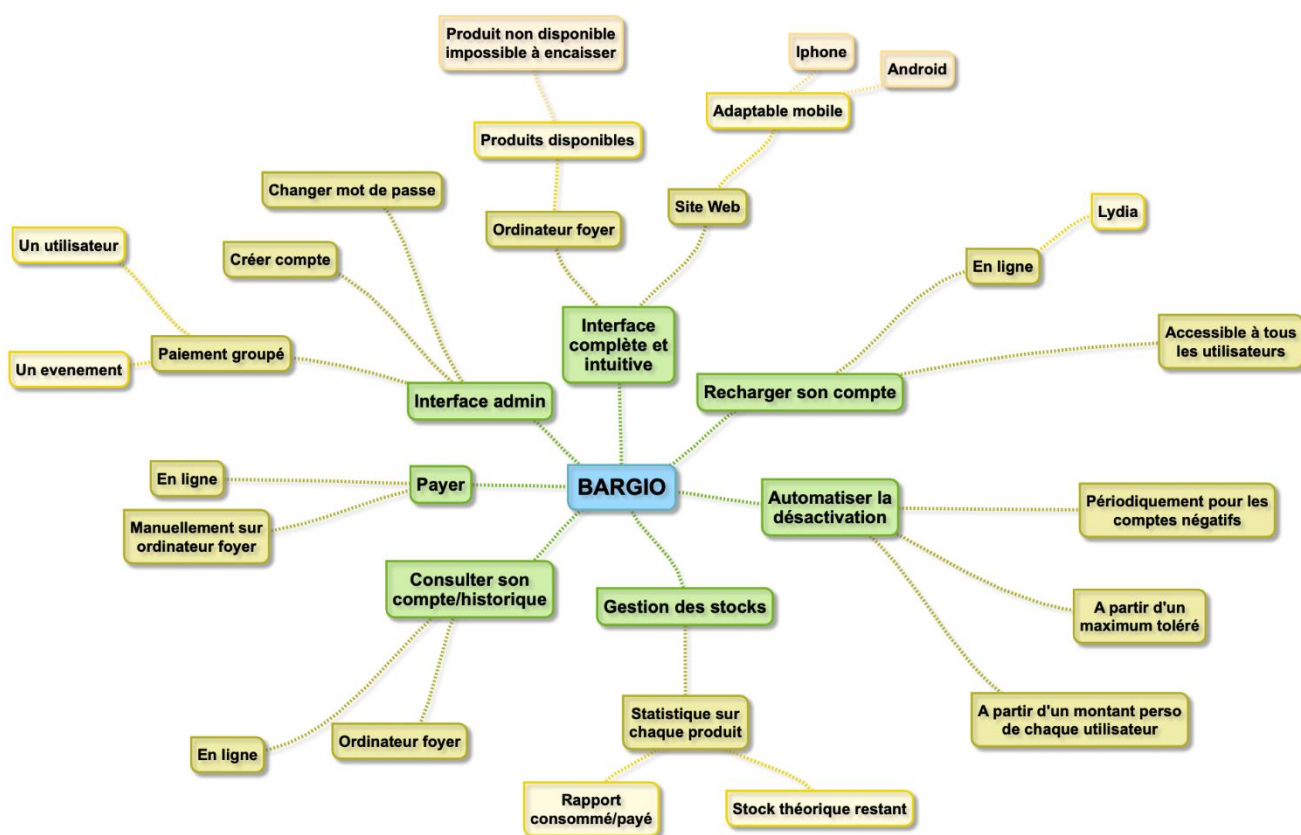


Fig 1. Mind Map

Q : Quelle(s) fonctionnalité(s) supplémentaire(s) voudriez-vous ajouter dans ce programme ?

R : Déblocage du compte automatique suite au rechargement
Afficher des statistiques (grands consommateurs, records journaliers...)
Recevoir une notification sur téléphone portable lors d'une consommation
Seuil de blocage de compte personnalisable
Pouvoir annuler une consommation due à une faute de frappe

Une fois terminée, cette liste des fonctionnalités a pu être exploitée afin d'entamer une réflexion sur quelles technologies utiliser.

Une plateforme web se sépare en deux parties : le front-end qui concerne tout ce qui est exécuté côté client (design, layout, scripting...), et le back-end qui est exécuté côté serveur (gestion du hardware, bases de données, sécurité...).

A la fois pour le front et le back-end, il existe des *frameworks*, qui sont des infrastructures logicielles spécialisées permettant de créer les fondations d'autres projet. Nous avons dû étudier et comparer différents frameworks afin de trouver le plus pertinent. Nous

avons choisi ASP.NET pour le back-end (cf fig. 2), et l'association de bibliothèques Bootstrap/jQuery pour le front-end (cf fig. 3).

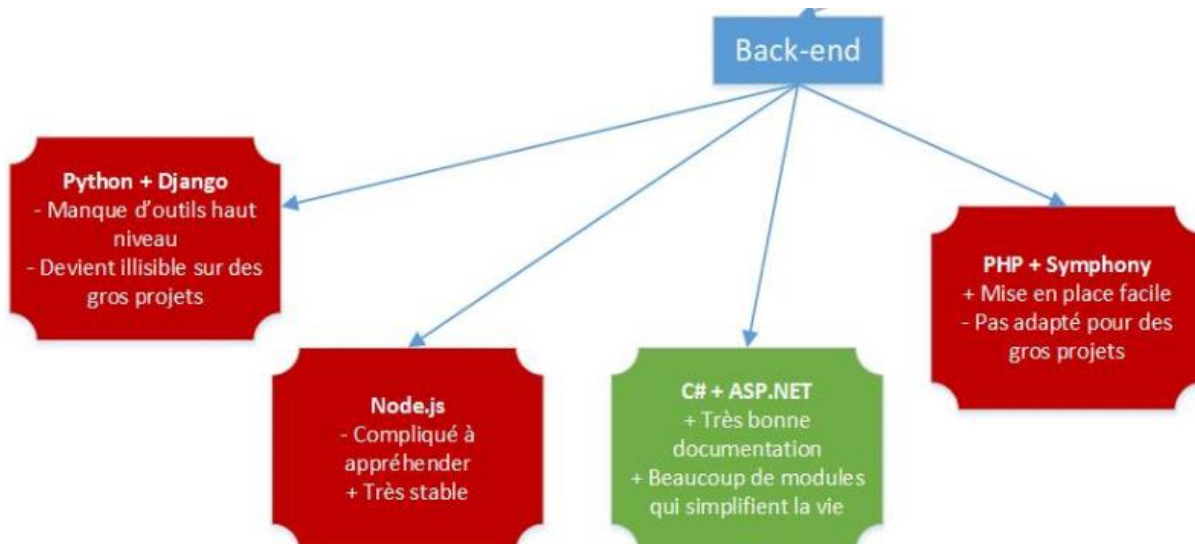


Fig. 2 : Arbre des technologies (framework back-end)

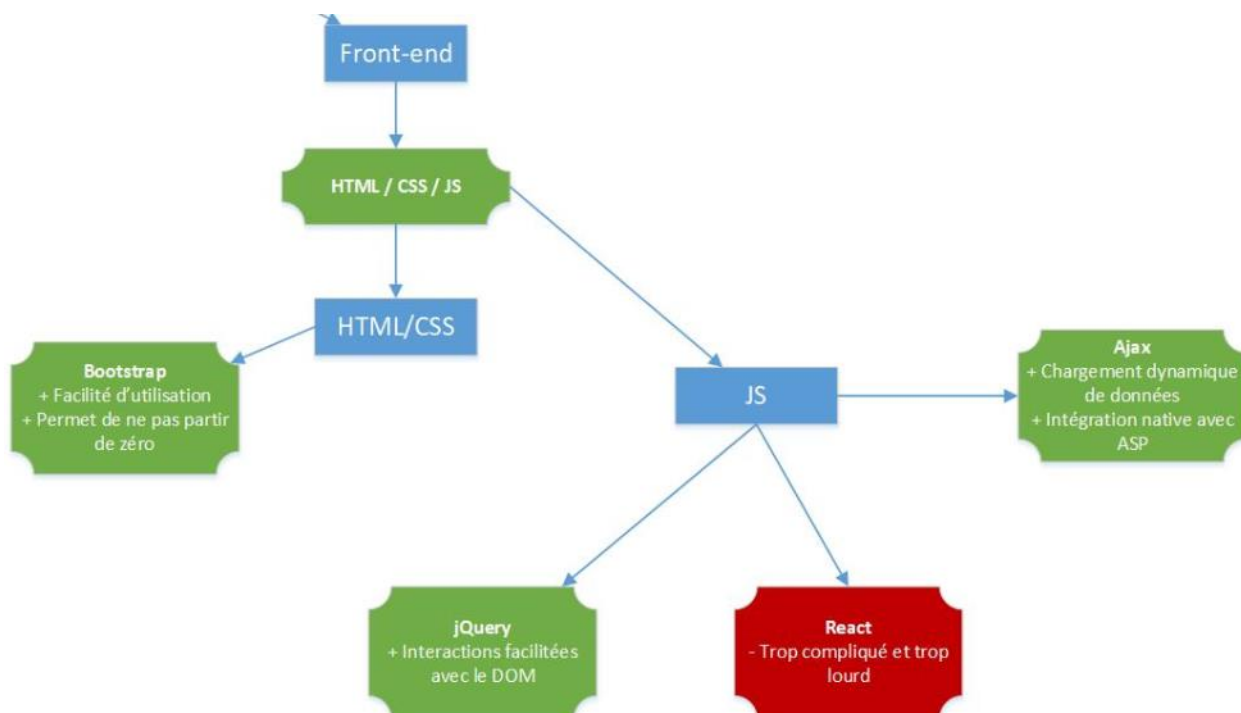


Fig. 3 : Arbre des technologies (framework front-end)

Une fois le framework déterminé, il a fallu choisir un *design-pattern*, qui représente une manière de programmer, d'organiser son code et de le hiérarchiser. Effectivement, notre framework en proposait deux différents, nous avons choisi Razor Pages (cf fig. 4).

Ces choix étant effectués, il ne nous restait plus qu'à choisir un système de bases de données, ainsi qu'un modèle d'hébergement. Nous sommes partis sur SQL Server couplé à son framework Entity, et un hébergement partagé (cf fig. 4,5).

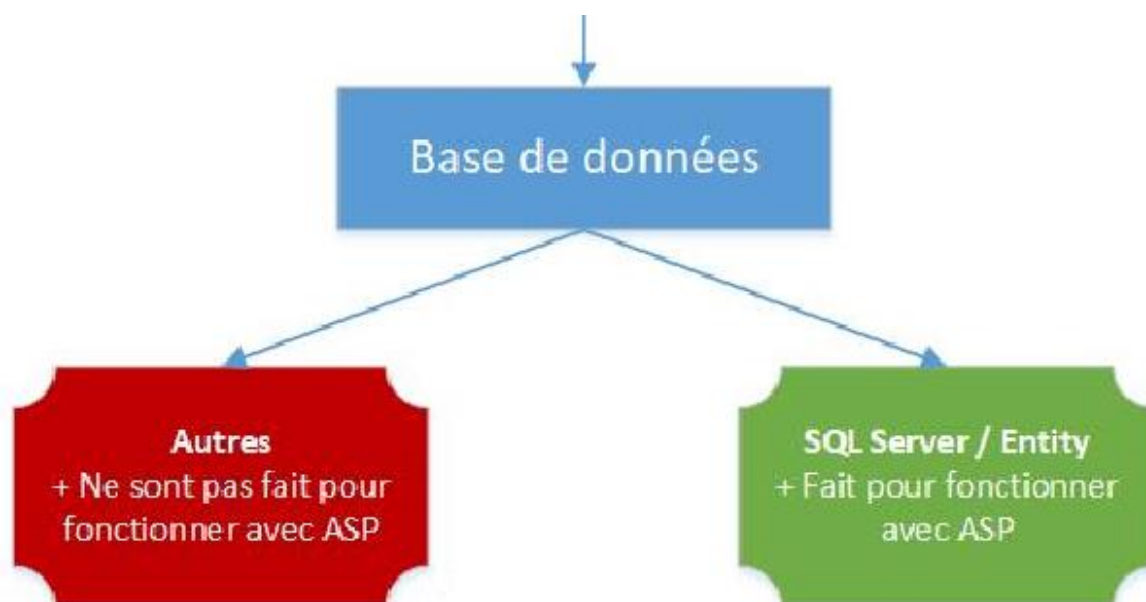


Fig. 4 : Arbre des technologies (bases de données)

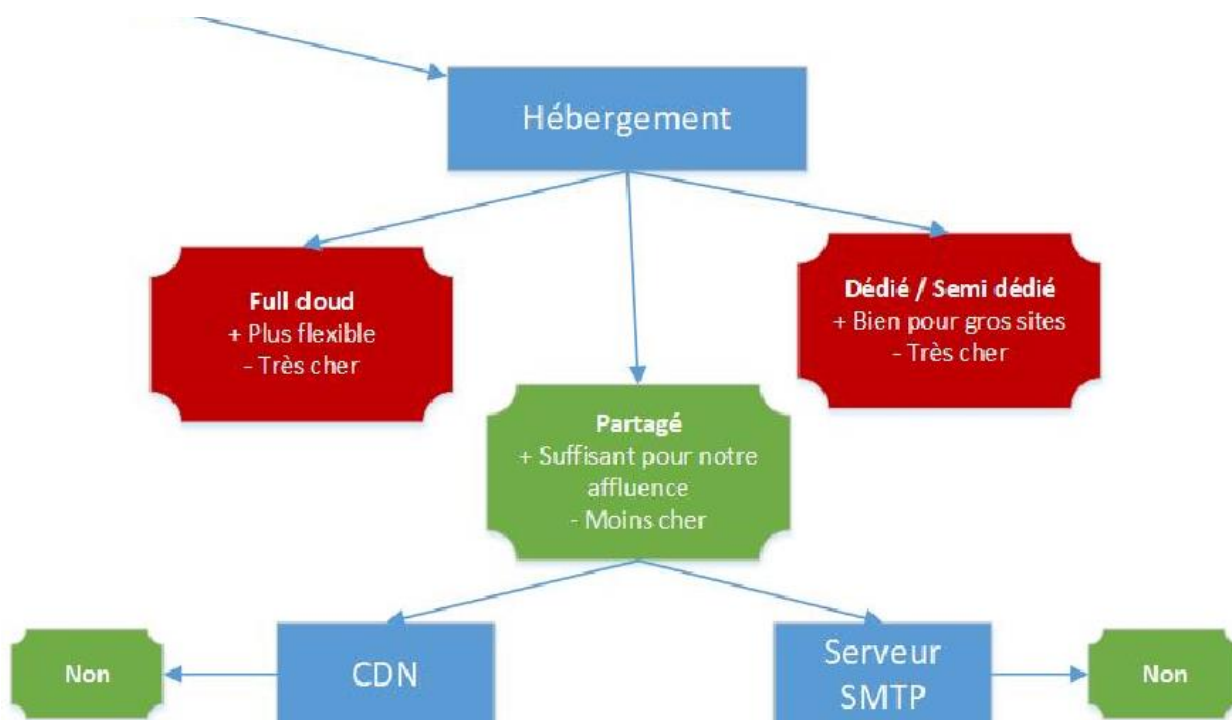


Fig. 5 : Arbre des technologies (hébergement)

II – Répartition des tâches et familiarisation avec les outils

A ce stade, nous avons une liste exhaustive des fonctionnalités à implémenter ainsi qu'un arbre des technologies complet.

La prochaine étape a donc été de réaliser un diagramme Gantt afin de relier les fonctionnalités aux technologies et de planifier le travail (cf fig. 6).

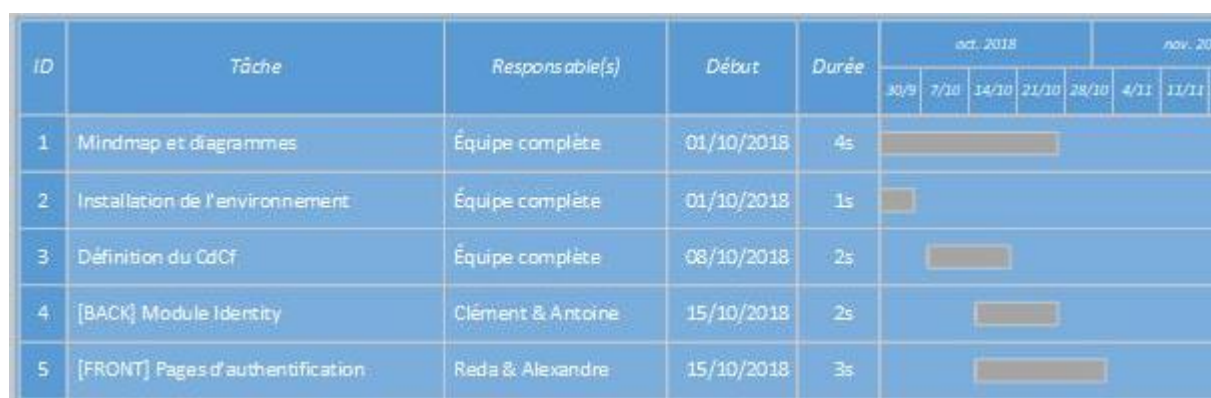


Fig. 6 : Vue partielle du diagramme Gantt

Le diagramme étant bien fourni (25 tâches sur 36 semaines), nous ne vous proposons qu'une vue partielle. Le diagramme entier est disponible en annexes.

Lors de la réalisation de ce Gantt, nous avons constaté qu'il était compliqué de paralléliser différentes tâches : le back-end a besoin que le front-end soit au même avancement que lui pour pouvoir en réaliser une implémentation. Cela nous a amené à une logique de travail en 2 équipes, la première s'occupant exclusivement du front-end (MM. Alyacoubi et Orain), et la seconde exclusivement du back-end (MM. Brault et Champion). Les étapes intermédiaires telles que la tenue de projet, les phases de test et le passage en production sont cependant gérées par l'équipe entière.

Afin de réaliser ce diagramme, nous avons dû beaucoup nous documenter afin d'anticiper au maximum les différentes étapes du projet, et de savoir comment les appréhender.

Tous les membres de l'équipe n'ayant pas forcément la même expérience avec les outils de programmation, un temps d'appréhension et de familiarisation avec les outils de programmation a également été mis en place, et nous avons commencé avec les modules les plus simples à réaliser, c'est-à-dire la création de comptes et les pages de connexion/déconnexion.

Pour le back-end, nous avons codé en C# avec ASP.NET Core 2 et Javascript sur Visual Studio, et pour le front-end, en HTML+CSS et Javascript sur Adobe Brackets.

Suite à cette première période, nous avons réalisé une arborescence complète de notre site afin de faire apparaître clairement la navigation au sein de celui-ci (cf fig. 7).

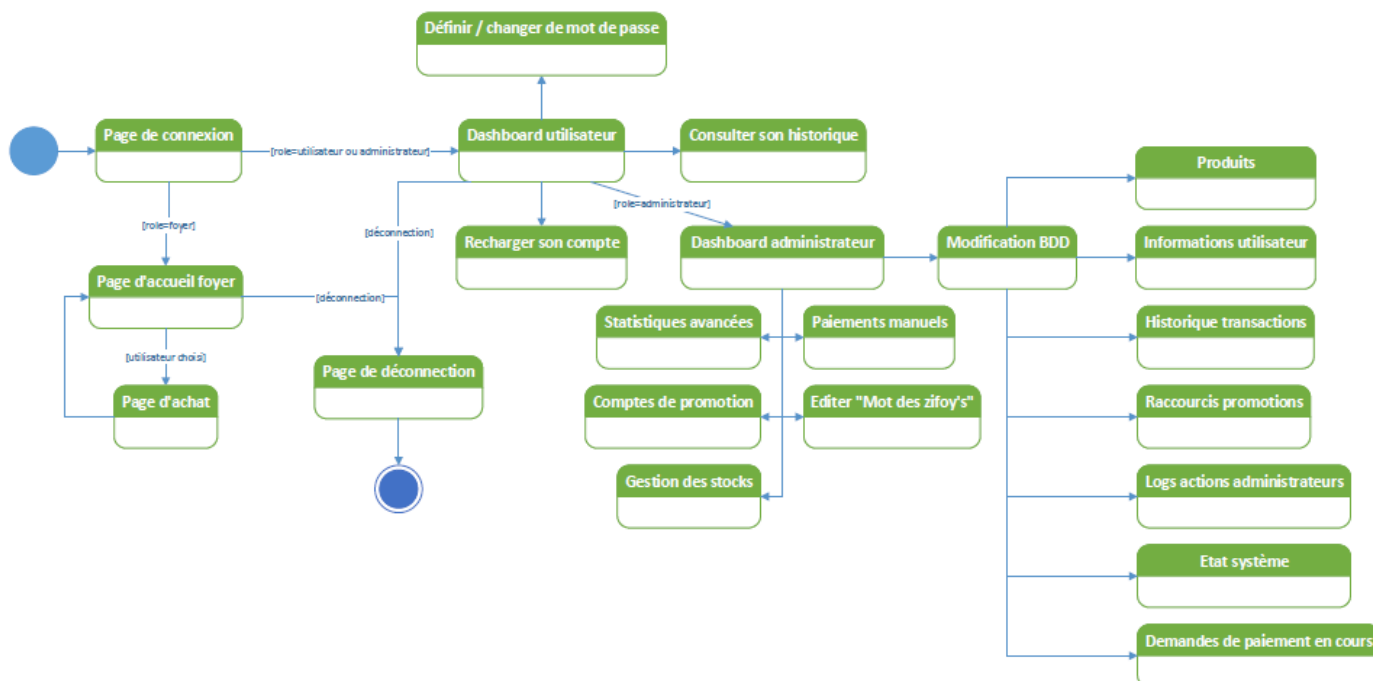


Fig. 7 : Arborescence du site


III – Module d'authentification

La première étape que nous avons réalisée concernait le module d'authentification : créer des bases de données d'utilisateurs qui peuvent se connecter à l'aide d'un mot de passe optionnel, gérer ce mot de passe et se déconnecter.

IV – Pages d'authentification en ligne (Front-end)

Après s'être familiarisé avec le langage choisi, nous nous sommes dans un premier temps intéressé à l'interface d'authentification utilisateur en ligne. Ces pages doivent avoir un format adaptable aux mobiles et aux ordinateurs.

La première page sur laquelle nous avons travaillé est la page d'accueil en ligne qui permet aux utilisateurs ou aux administrateurs de se connecter à leur Dashboard depuis leur mobile ou leur ordinateur personnel (Fig. 8). L'authentification de l'utilisateur nécessite d'entrer un identifiant sous la forme Num'ssXxProm'ss, qui est unique à chaque étudiant Gadz'Art ainsi qu'un mot de passe, si celui-ci a préalablement été défini. Ce mot de passe étant optionnel.



BARGIO


Connexion à mon compte

☒ Mot de passe optionnel

CONNEXION

Fig. 8 : Page d'accueil en ligne

Nous avons ensuite codé les pages de création de mot de passe ainsi que celle de changement de mot de passe (Fig.9 et Fig. 10).



BARGIO

Définir mon mot de passe optionnel

CONFIRMER

Fig. 9 : Page de création du mot de passe optionnel

**BARGIO**

Changer mon mot de passe optionnel

Ancien mot de passe...

Nouveau mot de passe...

Confirmer nouveau mot de passe...

CONFIRMER

Fig. 10 : Page de changement du mot de passe

Bien que ces pages soient visuellement simples et peu nombreuses, ce résultat a été difficile à obtenir. En effet, il s'agissait des premières pages que nous réalisions entièrement et concrètement : il était parfois difficile d'identifier nos erreurs ou de réaliser le codage permettant d'obtenir le rendu que l'on souhaitait. De plus, il fallait choisir et concevoir un design simple et soigné que l'on retrouverait sur toutes les futures pages qui composeront la plateforme de Bargo.

V – Intégration des pages d'authentification sur le serveur

ASP.NET propose un système d'authentification intégré appelé Identity. Ce module contient un type générique `UserManager<IdentityUser>` qu'il faut appeler par injection de dépendances. Cette classe permet déjà de :

- Créer un nouvel utilisateur avec mot de passe
- Changer le mot de passe d'un utilisateur
- Faire se connecter et se déconnecter un utilisateur

La classe `IdentityUser` ne permet cependant pas de désactiver le mot de passe par défaut, nous avons alors dû créer une classe dérivée.

VI – Création de la base de données

Nous avons ensuite créé le schéma de notre base de données en fonction de nos besoins :

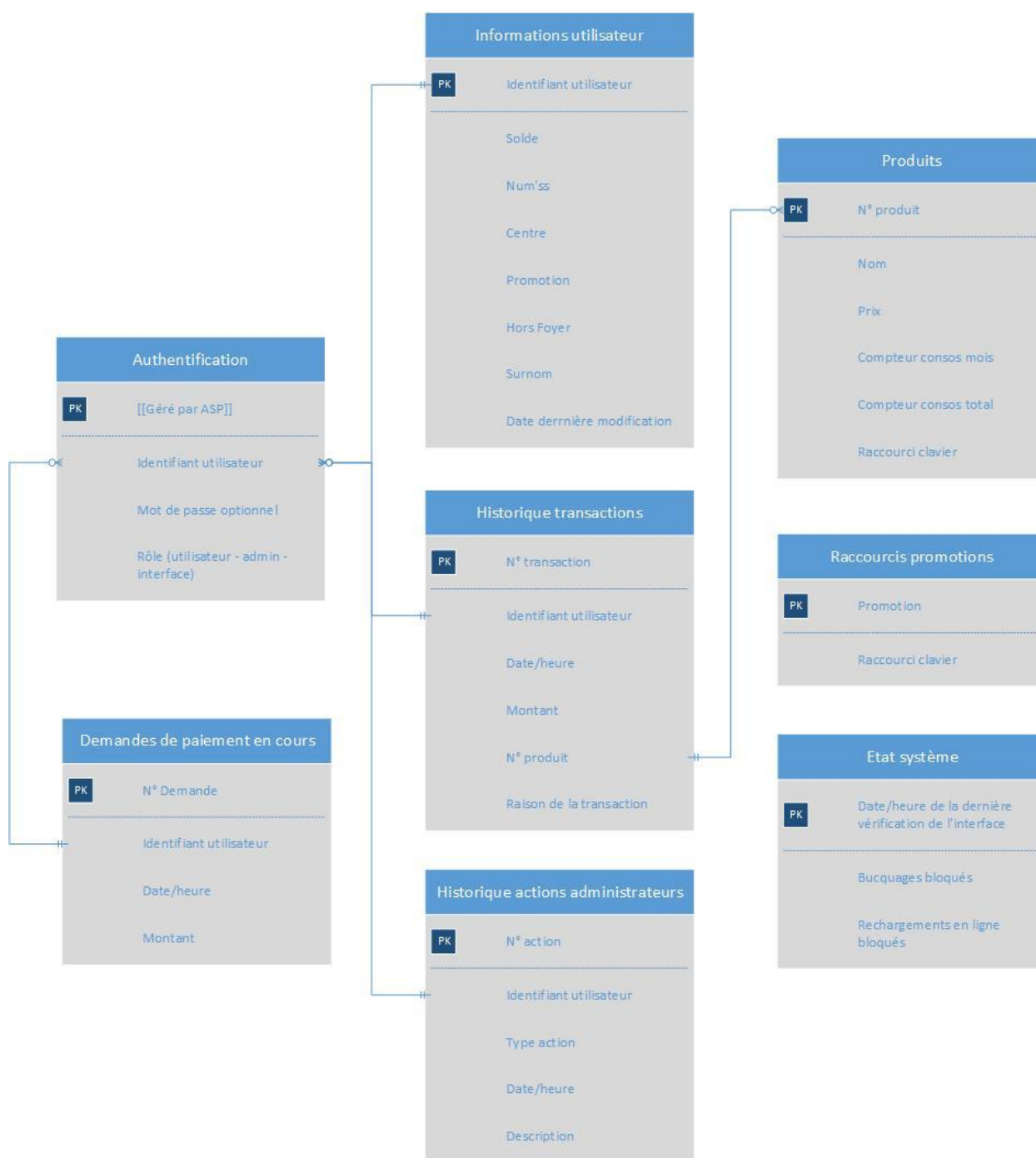


Fig. 11 : Schéma de la base de données

Pour ce faire, nous avons utilisé le moteur de base de données SQL Server intégré à ASP.NET ainsi que sa surcouche d'abstraction Entity Core. La philosophie d'Entity est très différente des moteurs SQL usuels : il n'existe pas de tables mais des « entités », pouvant stocker des variables classiques ainsi que des objets C#, mais aussi des données non-structurées à la manière de NoSQL. Les requêtes s'effectuent via LINQ.

Par exemple, si l'on désire récupérer le total des transactions effectuées entre une date A et une date B, nous effectuons la requête suivante :

```
var total = ApplicationDbContext.HistoriqueTransactions
    .Where(o => o.Date > A && o.Date < B)
    .Select(o => o.Montant)
    .ToList()
    .Sum();
```

VII – Interface sur l'ordinateur du foyer des élèves

Contrairement à la plateforme en ligne qui sera une nouveauté, une interface d'achats existe déjà sur l'ordinateur au sein du foyer des élèves. Un des objectifs définis pour notre projet était de refaire cette interface, afin de réduire les bugs et d'améliorer le visuel et l'ergonomie. Cette interface permet la connexion des utilisateurs et des administrateurs ainsi que l'achat de consommations, et elle reste en permanence ouverte sur l'ordinateur du foyer.

Les éléments présents avant la connexion d'un utilisateur sont les suivants (visible sur la Fig. 12) :

- Espace de renseignement du code de connexion de l'utilisateur (Num'ss de babasse)
- Tarifs des produits disponibles
- Une zone d'information : « Actualités & Évènements »
- Une zone dédiée au message à transmettre de la part des gérants du foyer : « Le mot des Zifoy'ss »
- Les raccourcis des différentes promotions
- Achat réalisé par le dernier utilisateur : « Dernier buquage »

Le fonctionnement pour l'utilisateur est le suivant : l'utilisateur entre son code de connexion (qui est indépendant des identifiants de connexion en ligne), qui est par défaut son numéro de Gadz'Art, dans l'espace défini. Il effectue la connexion à son compte en renseignant sa promotion au sein de l'école à l'aide du clavier (par exemple la touche F11 pour la promotion Li217). En renseignant sa promotion, l'utilisateur reste sur la même page. Des informations supplémentaires apparaissent :

- Si son solde n'est pas suffisant (généralement lorsqu'il est négatif) pour pouvoir utiliser son compte, une photo apparaît annonçant : « Tu es hors babasse » ainsi que le montant de son solde. L'utilisateur doit donc recharger son compte afin de procéder à un achat.
- Si son solde est suffisant, il accède à son solde, son historique et son encours d'achat. Il peut alors procéder à des achats. Pour cela, il sélectionne les produits qu'il souhaite consommer à l'aide de la souris sur la zone des tarifs, ou bien en utilisant les raccourcis sur le clavier. L'encours augmente sans modifier le solde pour le moment. L'utilisateur appuie ensuite sur la touche *Entrée* afin de valider ses achats.

Malgré la demande de certains élèves, nous avons décidé de ne pas afficher sur cette interface les statistiques de consommation de l'utilisateur. Nous avons pensé qu'il s'agirait d'un moyen de se défier entre élèves et donc de pousser certains à la surconsommation dans le seul

but d'avoir des « bonnes » statistiques, ce qui peut être dangereux. De plus, cela relève du privé.

Les éléments présents sur cette page restent les mêmes que ceux actuellement présents. Nous avons amélioré l'esthétique et la visibilité des informations.

Autres
F1

Li208
F2

Li209
F3

Li210
F4

Li211
F5

Li212
F6

Li213
F7

Li214
F8

Li215
F9

Li216
F10

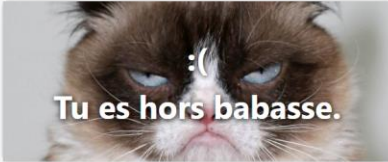
Li217
F11

Li218
F12

Tarifs

Bière F1	F1	Prix
Bière F2	F2	Prix
Bière F5	F5	Prix
Bière bout'ss	F4	Prix
Cidre	F6	Prix
Dose d'alcool fort	F3	Prix
Zabulle	F10	Prix
Biffut	F12	Prix
Pain choc'ss	F7	Prix

Num'ss de babasse



Tu es hors babasse.

86Li217
Alexandre la salamandre
+123,00€
-86,00€

Actualités & Événements
Jeudi 8 Novembre - Zbeuuuuuuul
Vendredi 9 Novembre - Remise de fam'ss
WBP - fin'ss - ce genre de choses...

Historique

Conso	Date
Conso	Date
Conso	Date
Conso	Date
Conso	Date
Conso	Date
Conso	Date

Le mot des Zifoy'ss
L'endroit parfait pour caler des rhopes hyper ayat'sssssssss.

Dernier bucuage : F5 le XX/XX/XX à XX:XX
BARGIO

Fig. 12 : Interface du foyer

Nous n'avons pour le moment réalisé que le front-end, l'implémentation sur le serveur reste à définir au second semestre.

VIII – Interface de paiement en ligne

Le but premier de notre projet était de rendre possible le rechargement de son compte en ligne. D'une part, le rechargement est plus facile pour l'utilisateur et plus rapide, d'une autre part, il décharge les responsables du foyer de recharger les comptes à chaque fois qu'un utilisateur le demande. En effet, jusque maintenant, l'utilisateur qui voulait recharger son compte devait aller voir un responsable et lui effectuer un virement. Ce dernier devait aller sur l'ordinateur du foyer pour renseigner le montant payé par l'utilisateur.

Nous avons donc réalisé une page de rechargement du compte du foyer (Fig. 13). Le rechargement se fait via Lydia, qui est une société spécialisée dans le paiement en ligne. Pour disposer d'un compte il suffit d'un numéro de téléphone qui est donc unique pour chaque compte Lydia. Notre page en question demande à l'utilisateur son numéro de téléphone (qui n'est pas stocké, simplement transmis à Lydia).

En back-end, un jeton unique de transaction est créé et stocké (dans l'entité Demandes de paiement), puis transmis à Lydia via son API, accompagné du montant et du numéro de téléphone. Cette API attend que l'utilisateur valide le paiement, puis envoie une requête POST sur une de nos pages avec le jeton de transaction et l'état du paiement (finalisé ou annulé).

Nous pouvons alors recharger l'utilisateur du montant qu'il avait renseigné.

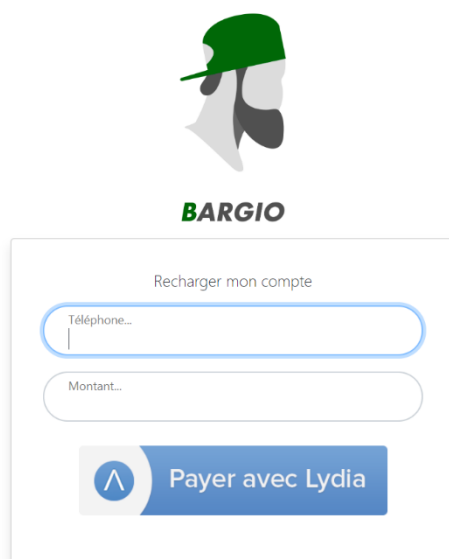


Fig. 13 : Page de rechargement en ligne

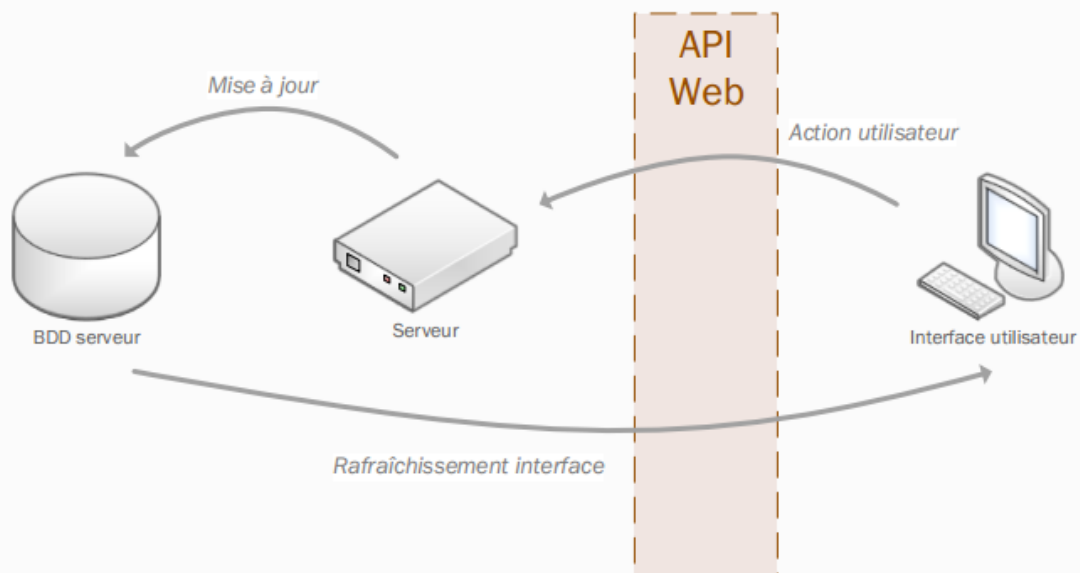
IX – Synchronisation interface foyer et site

Une des plus grosses contraintes du projet est de faire en sorte que les gens puissent acheter des consommations même quand il n'y a pas internet sur l'ordinateur du foyer. Cela nous a amené à concevoir un système de synchronisation entre la page d'achat qui reste en permanence allumée sur l'ordinateur du foyer, et nos serveurs (cf fig. 14).

De cette manière, en créant une base de données côté client, même si il y a une coupure d'internet au foyer, les personnes pourront quand même payer leurs consommations et le tout sera resynchronisé dès qu'internet sera revenu.

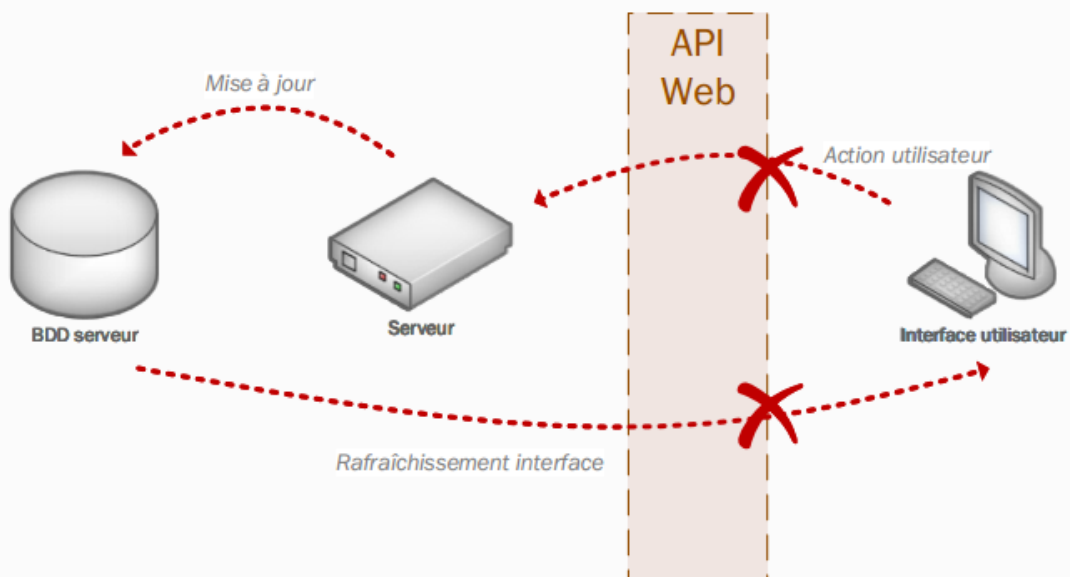
Le code de cette interface sera effectué au second semestre, nous utiliserons du javascript asynchrone avec Dexie.js comme moteur de base de données locale.

Interface serveur-client classique

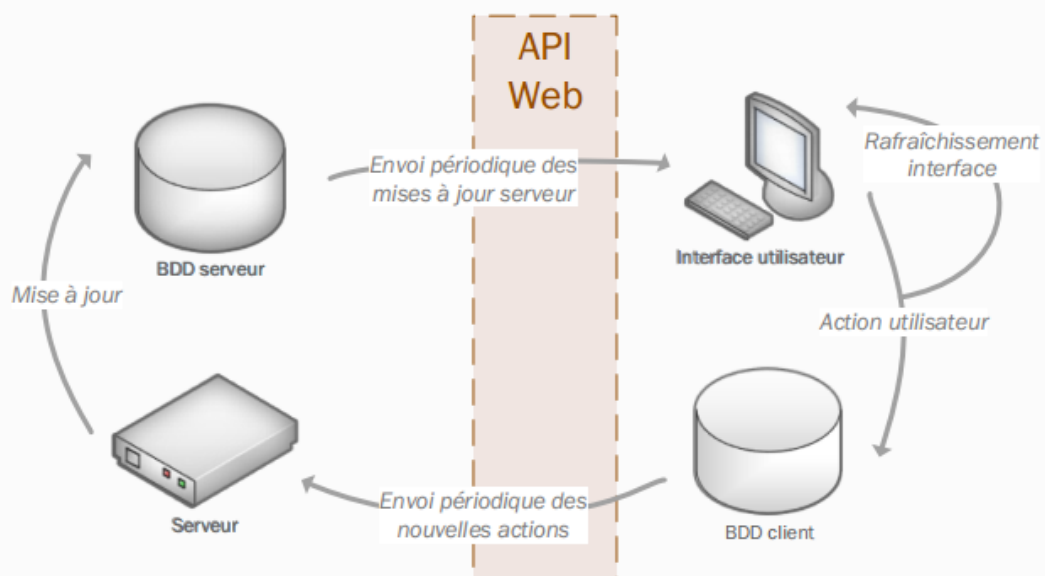


Interface serveur-client classique

Coupure d'internet



Interface serveur-client asynchrone



Interface serveur-client asynchrone

Coupure d'internet

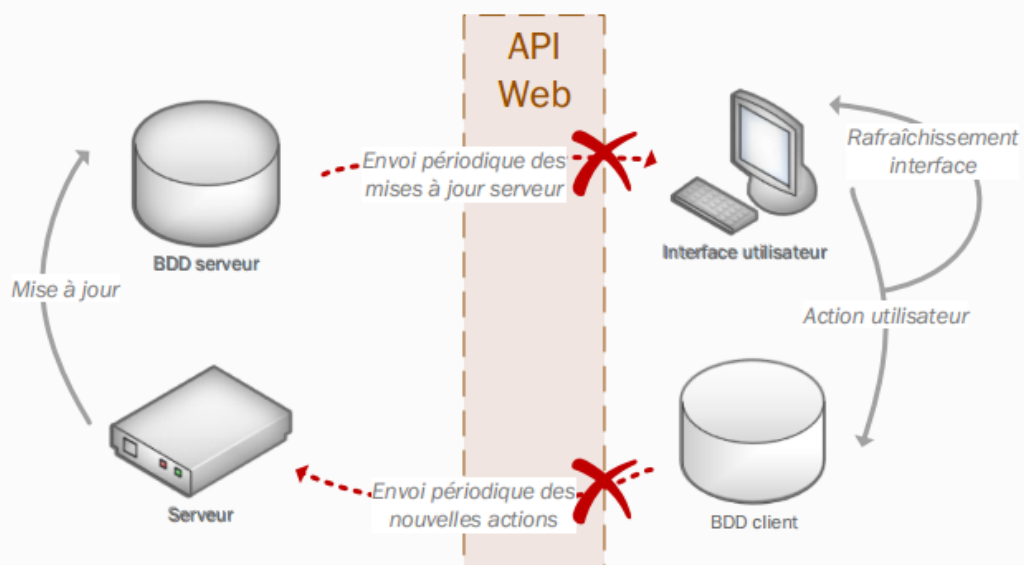


Fig 14. : Interface asynchrone pour la page d'achat

X – Première phase de test utilisateurs

Nous avons rassemblé des volontaires devant la page d'achat en leur demandant des actions simples afin de vérifier que l'ergonomie était bonne, puis devant la navigation utilisateur (connexion, gestion de mot de passe et rechargement), afin de vérifier cette fois à la fois l'ergonomie et les potentiels bugs.

Les deux tests ont été tout à fait concluant, les utilisateurs ayant trouvé le site pratique à utiliser, et seulement des bugs mineurs ont été détectés (problèmes d'affichage sur certains navigateurs...), qui ont été depuis corrigés.

XIII – Objectifs de la seconde phase du projet

La 1^{ère} phase de test utilisateur a été l'occasion pour notre groupe de prendre le temps de faire un point sur l'avancement de notre projet et des prochaines échéances afin d'assurer la bonne réalisation de notre projet. Comme on peut le constater avec l'arborescence du site, il manque de nombreuses pages à réaliser en front-end. En effet, il manque pour le moment les Dashboard utilisateurs et administrateurs, ainsi que les modules de gestion des stocks.

La partie la plus chronophage sera celle de la réalisation de la partie administrateur du projet, en front-end et back-end, l'objectif étant de donner plus d'outils aux gérants du foyer pour en assurer la bonne gestion, ainsi que le Javascript pour l'interface asynchrone. Ces outils pourront permettre notamment d'optimiser les produits à commander, identifier plus facilement les déficits potentiellement engendrés par des approvisionnements, faciliter la gestion des comptes utilisateurs, avoir des statistiques d'achat auprès de nos fournisseurs et partenaires, etc...

La finalité de notre projet est d'intégrer le site dans la vie de notre école, le dernier objectif sera donc d'effectuer une dernière phase de tests et les éventuelles corrections afin de pouvoir lancer le site sur nos serveurs. Bien sûr, une fois lancé, nous effectuerons les maintenances nécessaires et de l'amélioration continue afin de parfaire notre création.

ID	Tâche	Responsable(s)	Début	Durée	2018												2019																			
					oct. 2018				nov. 2018				déc. 2018				janv. 2019				févr. 2019				mars 2019				avr. 2019				mai 2019			
					30/9	7/10	14/10	21/10	28/10	4/11	11/11	18/11	25/11	2/12	9/12	16/12	23/12	30/12	6/1	13/1	20/1	27/1	3/2	10/2	17/2	24/2	3/3	10/3	17/3	24/3	31/3	7/4	14/4	21/4	28/4	5/5
1	Mindmap et diagrammes	Équipe complète	01/10/2018	4s																																
2	Installation de l'environnement	Équipe complète	01/10/2018	1s																																
3	Définition du CdCf	Équipe complète	08/10/2018	2s																																
4	[BACK] Module Identity	Clément & Antoine	15/10/2018	2s																																
5	[FRONT] Pages d'authentification	Rada & Alexandre	15/10/2018	3s																																
6	Réalisation arborescence du site	Équipe complète	29/10/2018	1s																																
7	[BACK] Intégration de JS sur le serveur	Clément & Antoine	29/10/2018	2s																																
8	Création entités de la base de données	Clément & Antoine	29/10/2018	2s																																
9	[BACK] Module de paiement	Antoine	05/11/2018	3s																																
10	[FRONT] Interface foyer	Rada & Alexandre	05/11/2018	5s																																
11	Module administration BDD en ligne	Clément	12/11/2018	4s																																
12	[FRONT] Interface de paiement	Rada & Alexandre	07/12/2018	4s																																
13	[BACK] Synchronisation interfaces foyer et site	Clément & Antoine	07/12/2018	4s																																
14	1ere phase de test utilisateur (retour sur ergonomie, détection de bugs)	Équipe complète	04/01/2019	2s																																
15	Point sur le projet et son avancement	Équipe complète	17/01/2019	1s																																
16	[FRONT] Module gestion des stocks	Rada & Alexandre	24/01/2019	2s																																
17	[BACK] Module gestion des stocks	Clément & Antoine	24/01/2019	2s																																
18	[FRONT] Dashboard utilisateur	Rada & Alexandre	07/02/2019	3s																																
19	[BACK] Dashboard utilisateur	Clément & Antoine	07/02/2019	3s																																
20	[FRONT] Dashboard administrateur	Rada & Alexandre	28/02/2019	3s																																
21	[BACK] Dashboard administrateur	Clément & Antoine	28/02/2019	3s																																
22	2e phase de test utilisateur	Équipe complète	21/03/2019	1s																																
23	Installation des serveurs	Équipe complète	28/03/2019	1s																																
24	Passage en production	Équipe complète	03/04/2019	2s																																
25	Maintenance et amélioration cont.	Équipe complète	18/04/2019	6s																																