

***Projet Cyclive***

---

**Caron Florian**  
**Yanmaz Armagan**

---

***IMA4***  
***2013/2014***

# *Sommaire*

I] Présentation du projet Cyclive

II] Choix de l'identification du produit

1) Code QR

2) Code NFC

3) Différences

III] Choix des outils

1) Les Frameworks

2) Framework Bootstrap

3) Framework Symfony

IV] Présentation du site

V] Conclusion

## **I]Présentation du projet**

Le projet « Cyclive » possède trois principaux buts :

1. Apporter un nouvel outil aux distributeurs de biens pour suivre, informer, conseiller ou encore récompenser et anticiper les attentes de leurs clients.
2. Introduire les consommateurs dans l'économie circulaire dans le cadre du développement durable et de l'économie verte.
3. Permettre la réparation, la revente, le troc ou encore le don et l'élimination d'un bien acheté.

Pour ce faire, nous avons mis en place un site mobile responsive d'adaptant aux différents terminaux, une application native étant beaucoup plus coûteuse et longue à réaliser.

Le projet se déroule en trois phases. Tout d'abord, il est nécessaire d'avoir un outil d'identification adapté du produit. Pour cela, nous avons le choix entre deux identifiants numériques : le code QR ou le tag NFC. Ainsi, la deuxième phase consiste à choisir le code le mieux adapté à la situation. Enfin, la dernière phase consiste à la création du site mobile responsive.

Ce dernier permettra notamment de :

- Créer un compte utilisateur
- Scanner un produit et l'ajouter dans son compte
- Possibilité pour l'utilisateur d'effectuer les actions :
  - Réparer
  - Vendre
  - Se débarrasser

Seule la fonctionnalité « Réparer » sera réalisée ici.

- Enregistrement de l'action de l'utilisateur dans la base de données
- Transmettre des données aux distributeurs de biens

Pour réaliser le site en lui-même, nous avons opté pour la partie HTML/CSS du site, le programme open source Bootstrap et le Framework Symfony pour la partie PHP, la base de donnée est quant à elle une MySQL.

## **II] Choix de l'identification du produit**

### 1) Le code QR

Le code QR est un code barre en deux dimensions. Il permet de stocker des informations de très grandes tailles (jusqu'à 7089 caractères numériques et 4296 caractères alphanumériques).

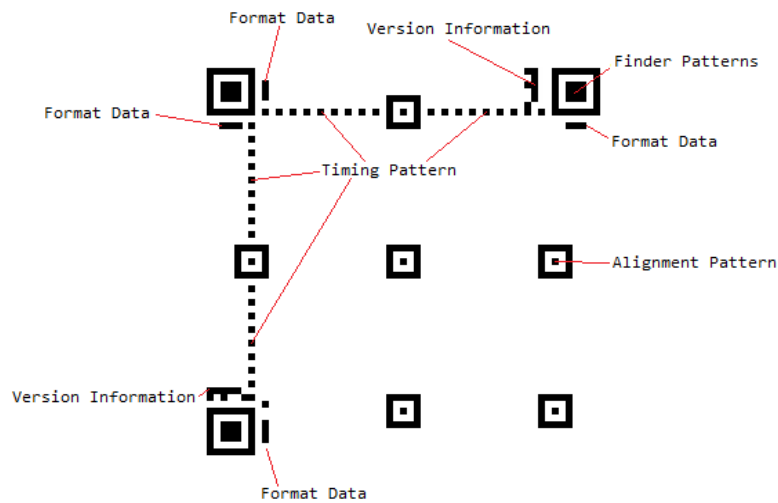
Il peut être lu par l'intermédiaire d'une webcam ou un Smartphone équipé d'une application de lecture. Le code QR est très simple à utiliser mais aussi très simple à créer. En effet de nombreux sites proposent la création du code avec l'information à coder et ceci sans frais. Un autre avantage du code QR est qu'il est tolérant aux salissures, dégradations etc... car il intègre un code correcteur et une redondance de l'information.

*Exemple de code QR du site [www.MyCyclelive.freeserver.me](http://www.MyCyclelive.freeserver.me)*



On note que le code QR est composé de carrés noirs et blancs qui s'appellent modules. Un module représente une valeur binaire. La valeur binaire du module noir est 0 et celle du blanc est 1.

## Structure détaillée d'un code QR



**Finder pattern** (« motifs de détection de position ») : au nombre de 3, ils fixent la rotation et les dimensions du code.

**Timing pattern** (« motifs de synchronisation ») : c'est une alternance de modules noirs et blancs qui permettent de densifier le code et de fournir des positions de référence pour le calcul des coordonnées de chaque module.

**Alignment pattern** (« motifs d'alignement ») : ils sont composés de 3 carrés concentriques superposés et de taille 5x5 modules pour le premier noir, 3x3 modules pour le blanc et un seul module central noir. Le nombre de ces patterns d'alignement dépend de la version de QR Code généré.

**Format data** (« données ») : ces modules contiennent deux informations : le niveau de correction d'erreurs et l'indice du masque mis sur le message original. En effet, la lecture du code étant crucial, il est nécessaire de stocker l'information à différents endroits.

## 2) Le tag NFC

Le NFC (« Near Field Communication ») ou « communication en champ proche » est une technologie sans fil qui permet de transférer des données comme l'adresse d'un site web, du texte ou encore des chiffres entre deux dispositifs NFC. Un tag NFC est un de ces dispositifs et est essentiellement une

micro puce qui possède une petite quantité de mémoire attaché à une antenne qui peut stocker une petite quantité d'information pour le transférer vers un autre dispositif, notamment les téléphones mobiles ou Smartphone. Le tag NFC se présente en général sous la forme d'un autocollant appliqué à un objet. Ainsi une personne peut aisément interagir avec son environnement.



### 3) Les différences

	Utilisation	Coût	Taille	Portée d'utilisation	Sécurité
Code QR	-Nécessité d'avoir une application adaptée pour lire le code -Viser le code avec la caméra de l'appareil	Le code QR est gratuit et coûte seulement les frais d'imprimerie	Doit avoir une taille minimale de 20mm x 20mm pour être lu correctement	Le code QR peut être lu par n'importe quels Smartphone	Le QR code n'offre pas de moyens de sécurité
Tag NFC	-NFC déjà intégré dans les Smartphone -Viser le tag	Le tag NFC est payant (20 à 30 centimes l'unité)	Taille de 10mm à 30 mm	Tous les Smartphone ne possèdent pas la technologie NFC	Le NFC intègre un algorithme de cryptage permettant de cacher les données

Pour une application mobile comme la nôtre, permettant seulement d'identifier un objet, il est préférable d'utiliser un code QR. En effet, ici il n'est pas question de sécurité et le critère de comparaison du coût l' « emporte » largement sur les autres critères.

### **III] Outils de conception**

Nous avons choisi d'utiliser pour la réalisation de ce site deux Frameworks beaucoup utilisés dans le développement web. Ceux-ci permettent de faciliter le développement en partant sur de bonnes bases. Ces deux Framework sont Symfony pour la couche back end du site, c'est à dire le traitement des données qui permettent au site d'être dynamique, et Bootstrap pour faciliter la gestion de l'aspect graphique du site. Nous allons dans cette partie vous présenter plus en détails ces deux outils. Mais tout d'abord, nous allons définir ce qu'est un Framework et quels avantages permettent leurs utilisations.

#### **1) Les Frameworks**

Pour commencer, le mot « *Framework* » vient de l'anglais et signifie « cadre de travail ». C'est un ensemble de composants qui servent à créer les fondations et l'architecture d'un logiciel ou dans notre cas un site. Ils sont très utilisés dans le développement et presque chaque langage de programmation possède ses Frameworks. Ceux-ci sont conçus par un ou plusieurs développeurs pour d'autres développeurs. Contrairement à certains scripts tels que WordPress, Joomla ou autres, un Framework n'est pas utilisable tel quel. Il demande beaucoup de travail de personnalisation.

L'utilisation d'un Framework a des avantages et des inconvénients. Le premier avantage est comme dit précédemment, l'amélioration de la productivité. Il en existe beaucoup d'autres que l'on peut classer en plusieurs catégories : le code, le travail et la communauté.

Tout d'abord, un Framework va aider à écrire un « bon code », c'est à dire un code lisible et organisé. Ainsi celui-ci pourra facilement être maintenu et amélioré par un autre développeur. De plus, un Framework offre des fonctionnalités prêtes à être utilisées, ce qui évite aux développeurs de devoir réinventer « la roue » à chaque nouveau projet. Elles permettent d'économiser des heures de développement.

Ensuite, un Framework améliore la façon de travailler. En effet pour créer un site internet, il faut en général plusieurs développeurs qui ont chacun leur spécialité. Il y a par exemple des codeurs PHP et des webdesigners, qui doivent travailler en même temps. Un Framework facilite ce travail en équipe. D'une part, ils utilisent généralement l'architecture MVC (Modèle - Vue - Contrôleur), qui permet de séparer le code HTML du PHP. Ainsi, le designer et le codeur travaillent sur des fichiers différents. D'autre part, un Framework a une structure et des conventions de code connues. Ainsi, il



est possible de recruter un autre développeur : s'il connaît déjà le Framework en question, il s'intégrera très rapidement au projet.

Enfin, le dernier avantage est la communauté soutenant chaque Framework. C'est elle qui fournit les tutoriaux ou les cours, de l'aide sur les forums, ou encore les mises à jour.

Il existe plein d'autres avantages :

- Une documentation de qualité est régulièrement mise à jour ;
- Un code source maintenu par des développeurs attirés ;
- Un code qui respecte les standards de programmation ;
- Un support à long terme garanti et des mises à jour qui ne cassent pas la compatibilité ;

Donc l'utilisation d'un Framework a beaucoup d'avantages et elle n'a en fait que très peu d'inconvénients. S'il ne fallait en citer qu'un, cela serait le temps nécessaire à l'apprentissage du Framework. En effet, pour maîtriser un Framework, il faut un temps non négligeable et réaliser plusieurs projets pour trouver les meilleures méthodes de conception et programmation.

## 2) Framework Symfony

Symfony est un Framework PHP très populaire, l'un des plus utilisés dans le monde, notamment dans les entreprises. La première version est sortie en 2005 et est aujourd'hui toujours très utilisée. Cela lui apporte un retour d'expérience et une notoriété exceptionnelle. Aujourd'hui, beaucoup d'entreprises dans le domaine du développement Web recrutent des développeurs capables de travailler sous ce Framework. Ces développeurs pourront ainsi se greffer aux projets de l'entreprise très rapidement, car ils en connaîtront déjà les grandes lignes. La deuxième version que nous utilisons pour le projet est arrivée en 2011. De plus, Symfony est open source, donc toute la communauté peut participer à son développement. Ainsi, il est possible de récupérer des bibliothèques réalisant des actions précises pour les intégrer sur son site. Nous avons par exemple pour notre site utilisé FOSUserBundle qui permet de gérer des utilisateurs sur son site : gestion des connexions / déconnexions, des sessions, etc...

Nous allons maintenant voir rapidement comment s'organise l'exécution du code. Symfony2 utilise l'architecture MVC qui signifie, comme cela a été dit plus haut, « Modèle / Vue / Contrôleur ».

C'est un découpage très répandu pour développer les sites internet, car il sépare les couches selon leur logique propre :

- Le **Contrôleur** génère la réponse à la requête HTTP demandée par le visiteur. Il analyse et traite la requête de l'utilisateur. Le contrôleur contient la logique de notre site internet et va se contenter « d'utiliser » les autres composants : les modèles et les vues. Par exemple, pour un site qui publie des articles, le contrôleur va vérifier que tel utilisateur peut publier et si ce dernier veut modifier un article, le contrôleur va récupérer l'article et demander le formulaire d'édition. Puis, il va envoyer ce formulaire à l'utilisateur avec les données dans leurs champs.
- Le **Modèle** gère les données et le contenu. Pour le cas de l'article, lorsque le contrôleur veut récupérer un article, il demande à la couche modèle de récupérer l'article avec un tel id, car seule cette couche connaît l'organisation des données dans la base de données, et elle va générer une requête SQL. D'ailleurs avec Symfony, même le développeur n'est pas obligé de connaître précisément la façon dont les données sont rangées, tout est géré automatiquement par le Framework.
- La **Vue** permet l'affichage des pages. Dans l'exemple de l'article, ce n'est pas le contrôleur qui affiche le formulaire, il ne fait qu'appeler la bonne vue. Si nous avons une vue Formulaire, les balises HTML du formulaire d'édition de l'article y seront et au final le contrôleur ne fera qu'afficher cette vue sans savoir vraiment ce qu'il y a dedans. En pratique, c'est le designer d'un projet qui travaille sur les vues. La séparation des vues et des contrôleurs permet aux designers et développeurs PHP de travailler ensemble sans se « marcher dessus ».

Le dernier point important concernant Symfony est son organisation en bundles. Un bundle est en fait une brique qui contient tous les contrôleurs, modèles, vues et fichiers associés en rapport avec une fonctionnalité du site. Nous avons par exemple pour notre site un bundle pour les utilisateurs, un pour les objets et un bundle principal qui s'occupe du reste. L'intérêt de ce concept est la possibilité d'échanger ses bundles avec d'autres développeurs, on peut également récupérer les bundles des autres, toujours dans le but d'augmenter la productivité.

### 3) Framework Bootstrap

Bootstrap est un Framework CSS, mais pas seulement, puisqu'il embarque également des composants HTML et JavaScript. Il comporte un système de grille simple et efficace pour mettre en

ordre l'aspect visuel d'une page web. Il apporte du style pour les boutons, les formulaires, la navigation, etc. Il permet ainsi de concevoir le design d'un site web rapidement et avec peu de lignes de code ajoutées.

Un des principaux avantages de Bootstrap est qu'il est compatible avec tous les navigateurs, ainsi le design sera le même pour tous. Ensuite, il fait gagner du temps de développement, il propose différents styles et est très personnalisable, il propose une gestion de type grille pour faciliter le positionnement des éléments et enfin il s'adapte aux différents types et tailles d'écrans.

#### IV] Présentation du site

Nous allons maintenant voir plus en détail le site que nous avons réalisé au cours de ce projet. Tout d'abord le site est séparé en deux parties. Une partie publique, où tous les utilisateurs inscrits peuvent aller et une partie privée réservée aux administrateurs.

Les plupart des utilisateurs du site arriveront sur ce dernier depuis une application mobile en général, qui va lire les codes QR placés sur les objets. Ils arriveront alors sur la page contenant la description de l'objet (cf. Figure 1). Ils devront ensuite cliquer sur le bouton "Ajouter à ma liste", afin d'accéder aux fonctionnalités du site. Si l'utilisateur n'est pas connecté, il sera redirigé vers la page de connexion (cf. Figure 2), il pourra s'inscrire si ce n'est pas fait (cf. Figure 3) et une fois connecté, l'objet sera ajouté à sa liste (cf. Figure 4). Il peut retirer l'objet de sa liste si il le souhaite.



Figure 1



Figure 2

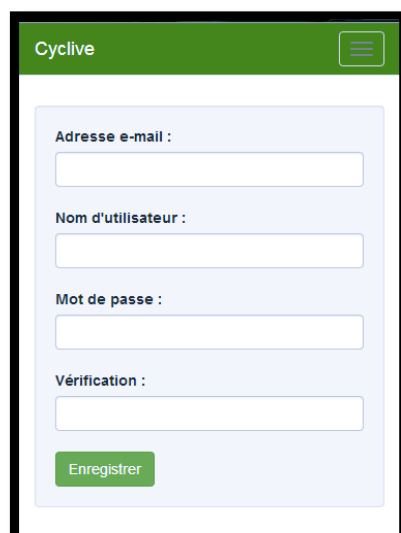


Figure 3

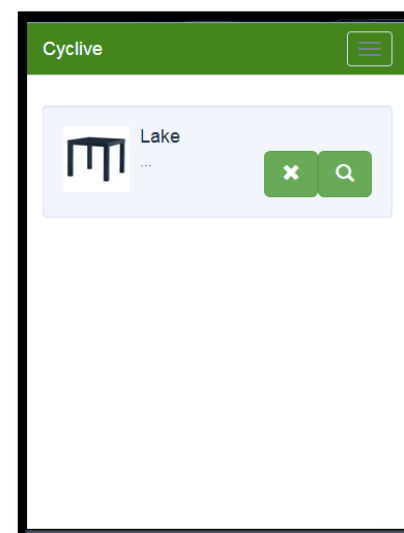


Figure 4

Une fois que l'utilisateur a l'objet dans sa liste, il peut accéder aux fonctionnalités "Réparer", "Vendre" et "Se Débarrasser", seule la fonction "Réparer" est implémentée pour le moment (cf. Figure 5).



Figure 5

Nous avons également une page de type blog pour informer les utilisateurs du site (cf. Figure 6) et une page de contact (cf. Figure 7).

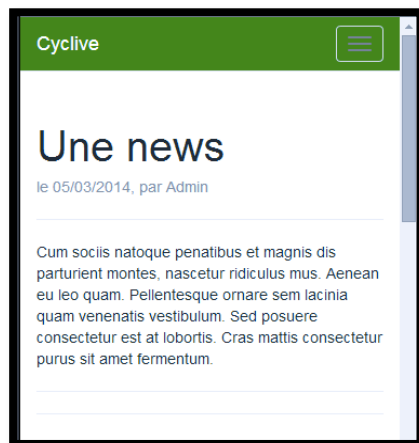


Figure 6

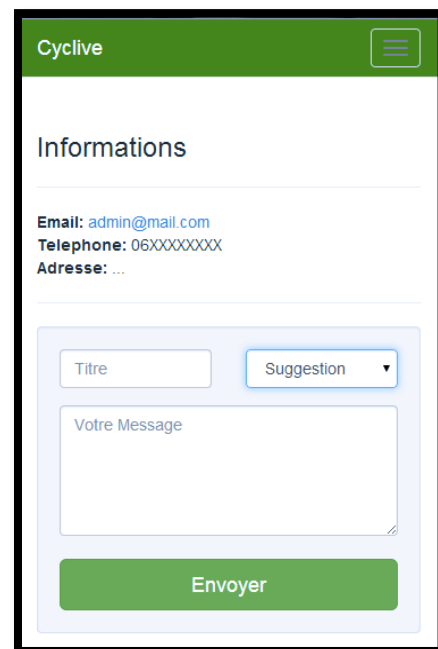


Figure 7

On peut accéder à toutes ces différentes pages depuis le menu qui apparaît lorsque l'on clique sur le bouton en haut à droite de l'écran, ce qui permet d'économiser de l'espace (cf. Figure 8).

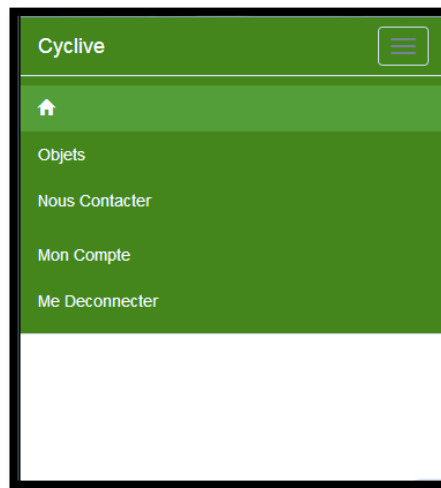


Figure 8

Passons maintenant à la partie administration. Celle-ci est également utilisable sur mobile, même si nous recommandons d'y accéder à partir d'un ordinateur. En effet, l'affichage des tableaux, se fait beaucoup mieux sur une grande surface. Vous pouvez voir ci-dessous deux pages permettant de gérer les messages du blog.

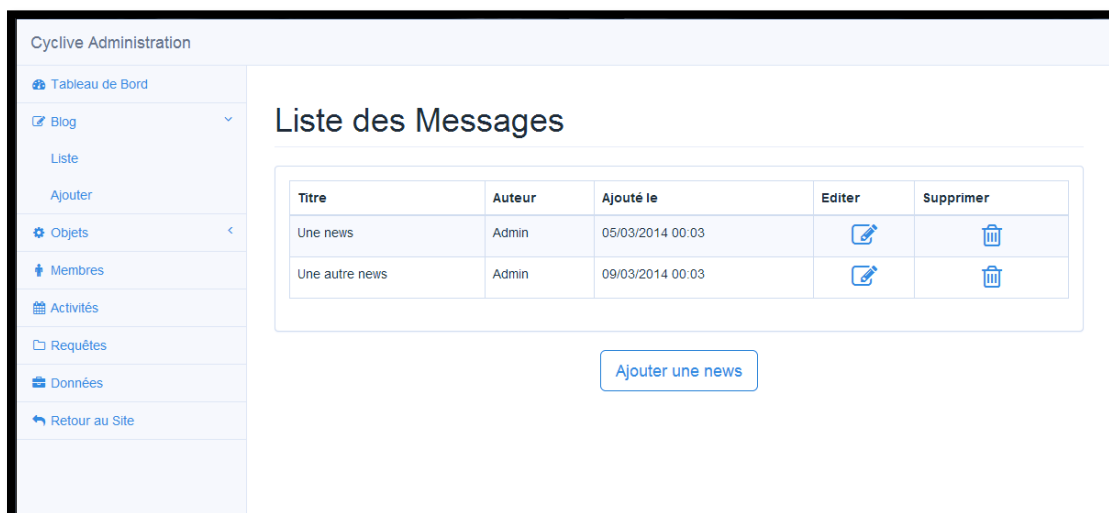
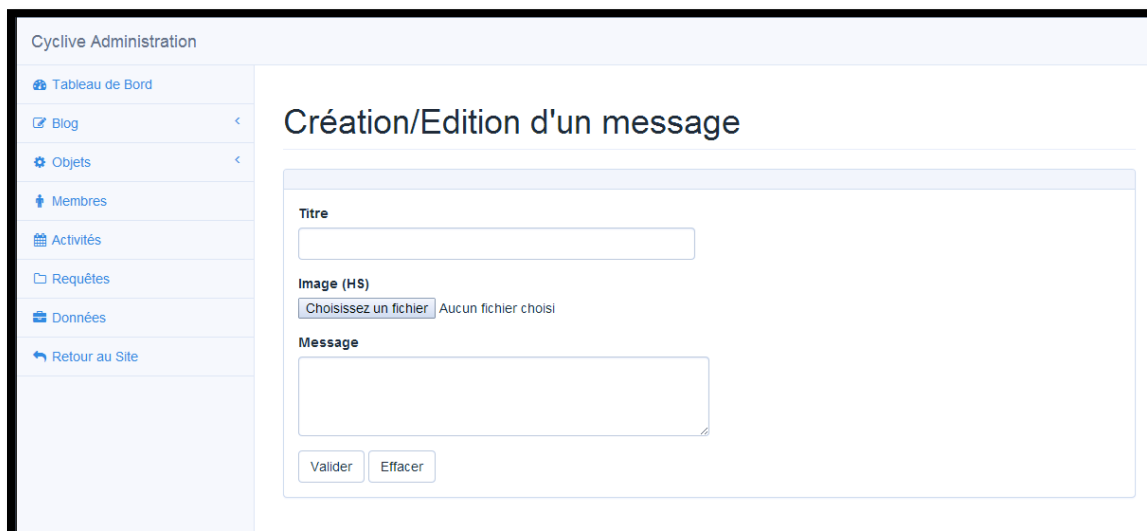


Figure 9 et 10

## **V] Conclusion**

Ce projet fût une bonne expérience car nous avons eu la possibilité de développer des compétences dans le domaine du développement Web comme la prise en main de Framework pour travailler avec du code HTML/CSS et PHP. Par manque de temps, nous n'avons pas pu implémenter d'autres fonctions utiles comme la possibilité de « Vendre », « Acheter » ou encore « Se débarrasser ». De même pour l'aspect visuel, il serait possible de le modifier pour avoir une allure attirante, simple et efficace, critères importants pour attirer les clients. Nous avons eu parfois des difficultés car nous n'avons pas eu un enseignement particulier sur le développement Web et il fallait par conséquent être autodidacte.