

# Cahier de conception

## Service de vente aux enchères



***Réalisé par :***

***MICHEL Jérémy  
MENAT Emmanuel  
TAURAN Xavier  
ALCAINA Mathieu***

***Date de rédaction :***

***30/04/2015***



# Table des matières

Introduction : Cahier des Charges.....	4
1. Interactions Homme-Machine.....	5
1.1. Charte graphique.....	5
1.2. Architecture des fichiers.....	10
1.3. Transitions d'une page à l'autre.....	11
2. Implémentation de la Base de Données.....	15
2.1. Diagramme entité – relation.....	15
2.2. Description des tables.....	16
2.3. Script de génération des tables.....	16
3. Services PHP.....	18
Fonctionnalité : inscription.....	18
Fonctionnalité : connexion.....	18
Fonctionnalité : déconnexion.....	18
Fonctionnalité : recherche.....	19
Fonctionnalité : tri.....	19
Fonctionnalité : enchérir.....	19
Fonctionnalité : affichage des informations des objets.....	19
Fonctionnalité : mise en vente.....	19
Fonctionnalité : fin de l'enchère.....	20
Fonctionnalité : récupérer informations compte.....	20
Fonctionnalité : supprimer enchères.....	20
Fonctionnalité : notifications évolutions des enchères.....	20
Fonctionnalité : achat immédiat.....	20
Fonctionnalité : liste de souhaits.....	20
Fonctionnalité : créditer compte.....	21
Fonctionnalité : supprimer utilisateurs.....	21
4. Répartition des tâches.....	22
4.1. Jérémy Michel.....	22
4.2. Xavier Tauran.....	22
4.3. Emmanuel Menat.....	22
4.4. Mathieu Alcaina.....	22

# **Introduction : Cahier des Charges**

Ce projet a pour but la réalisation d'un site Web de vente aux enchères. La réalisation de ce site nécessite la coordination de plusieurs mécanismes, à savoir le PHP, le HTML5, le CSS et le PostgreSQL. Plusieurs types de profils sont disponibles : un utilisateur non authentifié dispose de droits limités, un utilisateur authentifié a accès à la majorité des services à l'exception de certains auxquels a accès le profil administrateur (qui sont liés à l'administration du site).

Le site dispose de nombreuses fonctionnalités, qui sont mises en place à l'aide du PHP qui, en plus de traiter les données du client, permet également la communication avec la Base de Données PostgreSQL.

Ce dossier de conception est organisé en trois parties majeures : la première partie va traiter de l'Interface Homme-Machine (IHM), la seconde la Base de Données et la troisième les services PHP.

La première partie traitée concerne ainsi les IHM.

## **Mises à jour**

- Modification de la structure des pages du site WEB.
- Modifications au niveau des champs de la base de données (la structure des tables reste inchangée)

# 1. Interactions Homme-Machine

## 1.1. Charte graphique

Accueil connecté – enchères en cours

The image is a wireframe of a web browser window. The browser's address bar shows the URL "http://www.x3m.com". The page title is "Accueil". The browser's navigation buttons (back, forward, stop, home) are visible on the left. The page content includes a navigation menu with links: "Accueil", "Enchères en cours", "Vendre objet", "Acheter crédits", and "Mon compte". On the right side of the menu, it says "Crédits possédés: 75675 crédits" and "Déconnexion(toto)". The main content area features a large rectangular box with the text "Welcome on x3m.com". Below this, there is a search bar with a "ComboBox" dropdown menu, a "search" button, and a "Submit" button. At the bottom of the main content area, the text "Liste des enchères en cours :" is displayed. The browser's status bar at the bottom right shows a small icon.

## Accueil non connecté

← → × 🏠

http://www.x3m.com

🔍

Accueil

Inscription

Login

Password

← Connection

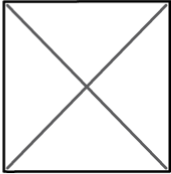
Welcome on x3m.com

ComboBox ▾

🔍 search

← Submit

Liste des enchères en cours :



Montre

Prix actuel : 500

Fin enchère : 29/05/15 00:50:46

Prix proposé

← Envoyer enchères

← Plus d'informations

## Acheter crédits

← → × 🏠

http://www.x3m.com

🔍

Accueil

Enchères en cours

Vendre objet

Acheter crédits

Mon compte

Crédits possédés: 75675 crédits

Déconnexion(toto)

Ajoutes des crédit à votre compte

Votre nombre actuel de crédits : 75675

Entrez la quantité de crédits que vous souhaitez ajouter

Enter amount

← Submit

←

→

✕

🏠

http://www.x3m.com

Rechercher

Accueil

Inscription

Connexion

Informations personnelles

Pays :

Enter country

Civilité :

☐ Homme ☐ Femme

Prénom :

Enter given name

Nom :

Enter family name

Date de naissance :

jj/mm/aaaa

Adresse :

Enter adress

Ville :

Enter city

Code postal :

Enter postal code

Téléphone :

Enter phone number

Informations Générales

Email :

Type your email

Confirmez votre Email :

Type your Email again

Identifiant :

Type your ID

Mot de passe :

Type your password

Confirmez votre mot de passe

Type your password again

Submit

## Vendre objet

Vendre objet

http://www.x3m.com

Q

AccueilEnchères en coursVendre objetAcheter créditsMon compteCrédits possédés: 75675 créditsDéconnexion(toto)

### Mettez un de vos objets en vente !

Nom de l'objet :  Quantité :

URL vers l'image de l'objet :

Catégorie :

Prix initial :

Prix d'achat immédiat :

Description de l'objet (500 caractères)

Durée de l'enchère :  
Heures :  Minutes :  Secondes :

## Mon compte

Accueil

http://www.x3m.com

Q

AccueilEnchères en coursVendre objetAcheter créditsMon compteCrédits possédés: 75675 créditsDéconnexion(toto)

### Mon compte

BIENVENUE A VOTRE COMPTE. ICI VOUS POUVEZ GERER L'ENSEMBLE DE VOS INFORMATIONS PERSONNELLES ET DES ENCHERES.

Menu du compte

Historique personnel

Mon profil

Contacts

#### Historique personnel :

Gérer vos objets mis en ventes, vos enchères en cours...

Objet(s) en ventes : 4 objets en vente

Enchère(s) remportée(s) : Aucune enchère remportée

Enchère(s) perdue(s) : 2 enchères perdues

#### Mon profil :

Gérez les préférences de votre compte et les paramètres des e-mails.

Informations générales

Adresse e-mail : emmanuel.menat@gmail.com

Mot de passe : \*\*\*\*\*

Informations personnelles

Nom : Emmanuel



## Mon compte - historique

← → × 🏠

http://www.x3m.com

🔍

AccueilEnchères en coursVendre objetAcheter créditsMon compteCrédits possédés: 75675 créditsDéconnexion(toto)

Mon compte

BIENVENUE A LA RUBRIQUE HISTORIQUE. ICI VOUS POUVEZ VISUALISER ET GERER L'ENSEMBLE DE VOS VENTES.

Menu du compteHistorique personnelMon profilContacts

Nom Objet	Prix	Date de mise en vente	Statut

## Mon compte – informations générales

← → × 🏠

http://www.x3m.com

🔍

AccueilEnchères en coursVendre objetAcheter créditsMon compteCrédits possédés: 75675 créditsDéconnexion(toto)

Mes Informations Générales

BIENVENUE A LA RUBRIQUE MES INFORMATIONS GENERALES. ICI VOUS POUVEZ MODIFIER VOS INFORMATIONS GENERALES.

Menu du compteHistorique personnelMon profilContacts

Informations générales

Email :

Confirmez votre Email :

Identifiant :

Mot de passe :

Confirmez votre mot de passe

← Enregistrer

## Mon compte – informations personnelles

Acceuil   Enchères en cours   Vendre objet   Acheter crédits   Mon compte   Crédits possédés: 75675 crédits   Déconnexion(toto)

### Mes Informations Personnelles

BIENVENUE A LA RUBRIQUE MES INFORMATIONS PERSONNELLES. ICI VOUS POUVEZ MODIFIER VOS INFORMATIONS PERSONNELLES.

Menu du compte   Historique personnel   **Mon profil**   Contacts

#### Informations personnelles

Pays :

Civilité : ☐ Homme   ☐ Femme

Prénom :

Nom :

Date de naissance :

Adresse :

Ville :

Code postal :

Téléphone :

### 1.2. Architecture des fichiers

Le projet comporte plusieurs types de fichiers, dont le nom et l'extension sont définies ici.

→ D'abord, les fichiers HTML : ils correspondent aux pages Web statiques affichées par le navigateur, et où on ajoutera du code Javascript. Ces pages seront principalement des pages statiques, car elles ne peuvent pas intégrer de PHP (à moins de configurer le serveur, mais ça n'est pas le cas dans notre projet)

→ Ensuite, les fichiers PHP : ils correspondent soit aux pages Web dynamiques, soit aux fichiers de traitement PHP. On aura ainsi énormément de fichiers PHP, car de nombreuses modifications doivent s'afficher. De plus, chaque fonctionnalité du site fait l'objet d'un fichier PHP, pour plus de visibilité. Une liste de ces services est trouvable [en troisième partie de ce document](#).

→ Les fichiers images (la bannière et les images des objets en vente) seront **tous** au format JPG ou JPEG.

### 1.3. Transitions d'une page à l'autre

Voici, sous forme algorithmique les différentes transitions possibles entre nos différentes pages. Nous détaillerons également quelles requêtes sont émises (ou du moins, le type de ces requêtes). Ces interactions seront traitées page par page.

#### Page Accueil – Utilisateur non connecté

**SI** Clic sur l'image d'un objet

**ALORS**

Affichage de la page description objet

Requête SQL SELECT pour récupérer les informations sur l'objet

**SI** Clic sur le bouton « enchérir »

**ALORS**

Affichage de la page de connexion

Service PHP ne détecte aucune session

**SI** Clic sur onglet « Enchere en cours »

**ALORS**

Affichage de la page de connexion

Service PHP ne détecte aucune session

**SI** Clic sur onglet « Vendre objet »

**ALORS**

Affichage de la page de connexion

Service PHP ne détecte aucune session

**SI** Saisie de « mot clé » dans la zone de recherche

**ALORS**

Affichage des objets contenant « mot clé »

Requête SQL de type SELECT sur les objets en vente

**SI** Clic sur menu déroulant « Catégorie »

**ALORS**

Affichage du menu déroulant

**ET SI**

Clic sur une des catégories

**ALORS**

Affichage des objets de cette catégorie

Requête SQL de type SELECT

**SI** Clic sur bouton « inscription »

**ALORS** Ouverture de la page inscription

**SI** Clic sur connexion ET (champ login vide **OU** champ mot de passe **vide**)  
**ALORS** Affichage de la page Connexion  
**SINON SI** Clic sur « connexion » **ET** (champ login **OK ET** champ mot de passe **OK**)  
**ALORS**  
    Lancement du service de connexion  
    Requête SQL de type SELECT

**SI** Clic sur bouton inscription  
**ALORS** Ouverture de la page d'inscription

### **Page Accueil – Utilisateur connecté**

**SI** clic sur image d'un objet  
**ALORS**  
    Affichage de la page description objet  
    Requête SQL de type SELECT

**SI** Clic sur le bouton « enchérir »  
**ALORS**  
    Affichage de la page « enchérir »  
    Détection de la session

**SI** Clic sur onglet « Enchere en cours »  
**ALORS**  
    Affichage de la page « Enchere en cours »  
    Détection de la session

**SI** Clic sur l'onglet « vendre objet »  
**ALORS**  
    Affichage de la page « vente d'objets »  
    Détection de la session

**SI** Clic sur onglet Mon compte  
**ALORS**  
    Affichage de la page Mon compte  
    Détection de la session

**SI** Clic sur l'onglet Acheter des crédits  
**ALORS**  
    Affichage de la page Acheter Crédits  
    Détection de la session

**SI** Saisie de « mot clé » dans la zone de recherche  
**ALORS**  
    Affichage des objets contenant « mot clé »  
    Requête SQL de type SELECT

**SI** Clic sur menu déroulant « Catégorie »

**ALORS**

Affichage du menu déroulant

**ET SI**

Clic sur une des catégories

**ALORS**

Affichage des objets de cette catégorie

Requête SQL de type SELECT

**SI** Clic sur « Déconnexion »

**ALORS**

Affichage de la page d'accueil en tant qu'utilisateur non connecté

Destruction de session

### **Page Inscription – Utilisateur non connecté OU connecté**

**SI** Clic sur le bouton « Inscription »

**ALORS**

Traitement du formulaire par le service PHP approprié

Service de connexion

Requête SQL de type INSERT INTO

**SI** (connexion échoue)

**ALORS**

Rechargement de la page Inscription

**SINON**

Début session PHP

Ouverture de la page d'accueil (statut : utilisateur connecté)

### **Page Enchere en cours – Utilisateur connecté seulement**

**SI** Clic sur Encherir

**ALORS** appel de 'encherir.php'

### **Page Vendre objet – Utilisateur connecté seulement**

**SI** Clic sur le bouton « Vendre »

**ALORS**

Traitement du formulaire par le service PHP approprié

Requête SQL de type INSERT INTO

**SI** (vente échoue)

**ALORS**

Rechargement de la page Vendre

**SINON**

Ouverture de la page d'accueil (statut : utilisateur connecté)

## **Page Mon Compte – Utilisateur connecté seulement**

**SI** Clic sur onglet « Mes ventes »

**ALORS**

Affichage de la fenêtre « Mes ventes »  
Requête SQL de type SELECT

**SI** Clic sur onglet « Mes encheres en cours »

**ALORS**

Affichage de la fenêtre « Mes encheres en cours ».  
Requêtes SQL de type SELECT

**SI** Clic sur onglet « Mes encheres remportés »

**ALORS**

Affichage de la fenêtre « Mes enchères remportées »  
Requêtes SQL de type SELECT

**SI** Clic sur onglet « Mes informations »

**ALORS**

Affichage de la page « Mes informations »  
Requêtes SQL de type SELECT

**SI** Clic sur onglets « Achats de crédits »

**ALORS**

Affichage de la fenêtre « Achat de crédits »

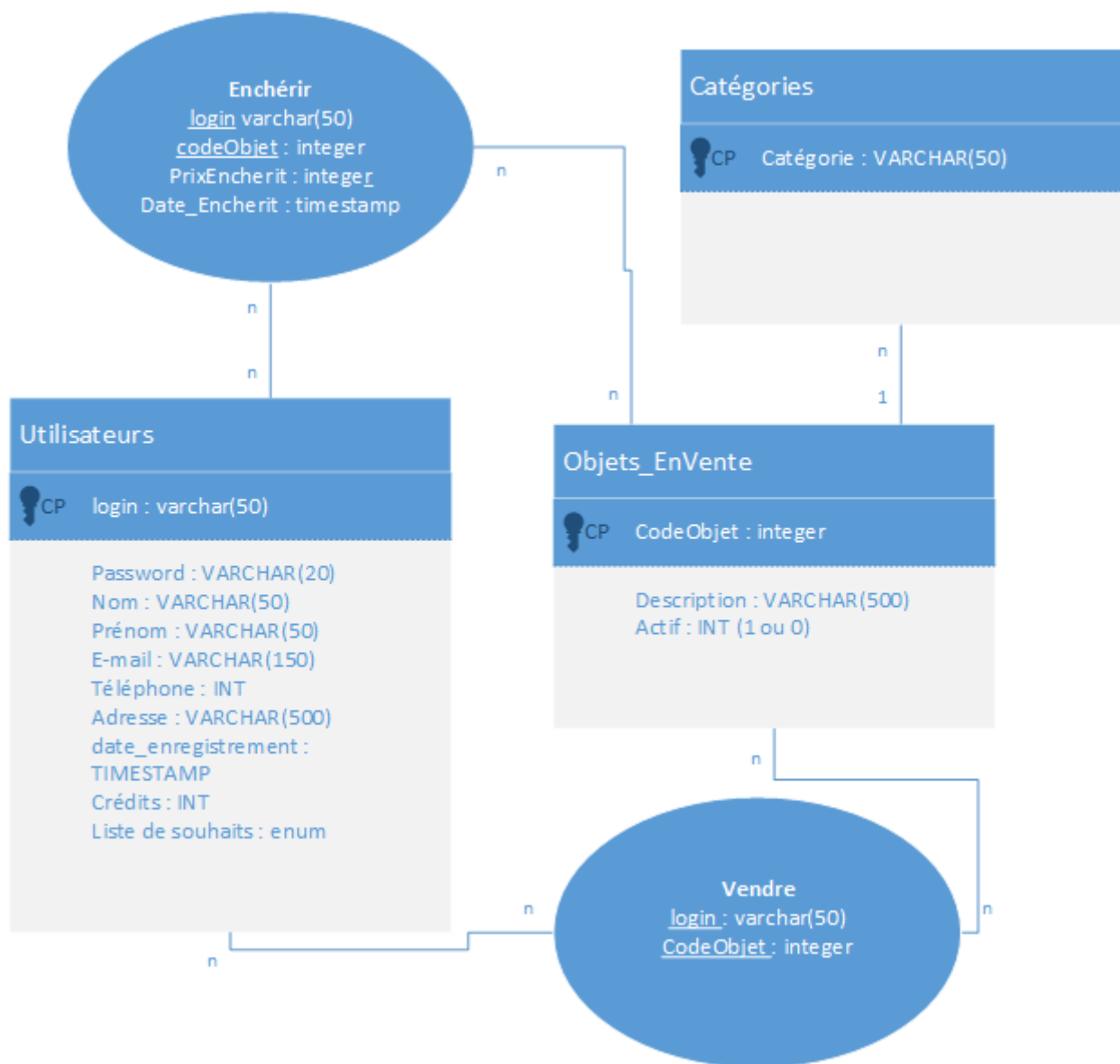
**SI** Clic sur onglet « Contacts »

**ALORS** Affichage de la fenêtre contacts (contient les informations de contacts du site – partie statique)

## 2. Implémentation de la Base de Données

Une Base de Données permet de stocker des informations et de les récupérer facilement à l'aide de requêtes. Cependant, une fois construite et en utilisation il peut être difficile d'en modifier la structure (ou du moins, cela peut demander beaucoup de travail), il est donc important d'en expliciter la structure dès le départ afin de ne pas perdre de temps.

### 2.1. Diagramme entité – relation



## 2.2. Description des tables

La base de données doit permettre la récupération et le stockage de nombreuses informations. La fonction et description avancée des différentes tables de la Base de Données.

### Table Utilisateurs

Cette table regroupe les informations sur les utilisateurs, y compris leur ID et mot de passe (crypté à l'aide de md5). Toutes les informations concernant les utilisateurs sont stockées dans cette table.

### Table Objets\_EnVente

Cette table regroupe les informations sur les objets actuellement vendus.

### Table Objets\_Terminés

Cette table regroupe sur les objets qui ne sont plus vendus. Un objet de la table Objets\_EnVente sera placé dans la table ObjetsVendus lorsqu'il sera vendu.

La table Utilisateurs et la table Objets\_EnVente sont reliés par deux liaisons « Vend » et « Encherit ». Objets\_EnVente et Objets\_Terminés sont reliés par une liaison « Devient ». Ces liaisons permettent aux tables de communiquer entre elles.

## 2.3. Script de génération des tables

### Table Users

```
CREATE TABLE Users (  
login varchar(50),  
password varchar(50),  
nom varchar(50),  
prenom varchar(50),  
email varchar(100),  
telephone varchar(15),  
adresse varchar(300),  
civilite varchar(1),  
pays varchar(20),  
date_enregistrement date,  
crédits integer,  
birthDate date,  
CONSTRAINT pk_users PRIMARY KEY (login)  
);
```



### **Table Catégories**

```
CREATE TABLE Categories (  
    libelleCategorie varchar(50),  
    CONSTRAINT pk_categories PRIMARY KEY (libelleCategorie)  
);
```

### **Table Objets**

```
CREATE TABLE Objets (  
    codeObjet integer,  
    nomObjet varchar(50),  
    quantiteObjet integer,  
    categorie varchar(50),  
    url varchar(200),  
    description varchar(500),  
    prixInitial integer,  
    prixAchatImmediat integer,  
    dateDebutEnchere timestamp,  
    dateFinEnchere timestamp,  
    statut integer,  
    CONSTRAINT pk_objets PRIMARY KEY (codeObjet)  
);
```

### **Table Vendre**

```
CREATE TABLE Vendre(  
    login varchar(50),  
    codeObjet integer,  
    CONSTRAINT pk_Vendre PRIMARY KEY (login, codeObjet),  
    CONSTRAINT fk1_Vendre FOREIGN KEY (login) REFERENCES Users (login),  
    CONSTRAINT fk2_Vendre FOREIGN KEY (codeObjet) REFERENCES Objets (CodeObjet)  
);
```

### **Table Encherir**

```
CREATE TABLE Encherir(  
    login varchar(50),  
    codeObjet integer,  
    prixEncherit integer,  
    dateEncherit timestamp,  
    CONSTRAINT fk1_ench FOREIGN KEY (login) REFERENCES Users (login),  
    CONSTRAINT fk2_ench FOREIGN KEY (codeObjet) REFERENCES Objets (codeObjet)  
);
```

## 3. Services PHP

Cette partie va étudier les différentes fonctionnalités. Il est important de noter que chacune de ces fonctionnalités fera l'objet d'un fichier .php différent.

### **Fonctionnalité : inscription**

Cette fonctionnalité consiste à créer un compte afin que l'utilisateur puisse s'y connecter. En pratique, lorsque l'utilisateur clique sur le bouton d'inscription sur le site Web, une page disposant de formulaires de saisie s'affiche, permettant à l'utilisateur de rentrer les informations nécessaires à la création du compte. Lorsque la totalité des informations sont renseignées, l'utilisateur clique sur « Valider ». Cette action va déclencher un fichier .php qui va traiter les données.

Plusieurs possibilités alors :

- les données sont incorrectes (champ(s) vide(s) ou mal renseigné(s)). La page du formulaire est alors rechargée avec une indication des informations incorrectes, permettant à l'utilisateur de les corriger. Aucune information n'est envoyée à la BDD.

- les données sont correctes. Les informations de chaque champ sont récupérées et stockées dans des variables. Une connexion à la base de données est alors réalisée (fonction `pg_connect()`), et une requête SQL de type INSERT INTO (`pg_query()`) ajoute une nouvelle ligne à la table Utilisateurs. Tous les champs sont stockés en clair, à l'exception du champ mot de passe qui est crypté à l'aide de la fonction `md5()` de PHP. Enfin, le PHP appelle la fonction `pg_close()` qui termine la connexion à la base.

Si l'opération s'est déroulée avec succès, un message apparaît sur la page HTML indiquant que l'opération s'est déroulée avec succès, et invitant l'utilisateur à se connecter en utilisant ses informations de compte entrées plus tôt.

Enfin, un mail est envoyé pour confirmer la création du compte.

### **Fonctionnalité : connexion**

Lorsque l'utilisateur entre son identifiant, son mot de passe, et valide, un fichier .php traite les données entrées. `pg_connect()` est invoqué, et `pg_query` appelle une requête de type SELECT afin de savoir si le combo identifiant/mot de passe rentré existe. Si la réponse est positive (la requête retourne une ligne) alors l'utilisateur existe, et on connecte l'utilisateur : on démarre une session à l'aide de PHP (`session_start()`).

Dans le cas où la requête SELECT échoue (ne renvoie rien), un message est ajouté en dessous du formulaire pour indiquer que la tentative de connexion a échoué.

### **Fonctionnalité : déconnexion**

Cette opération consiste à s'assurer que les informations de connexion de l'utilisateur ne sont pas conservées : elle permet ainsi essentiellement de détruire la session php créée. On appelle donc la fonction `session_destroy()`. Il reste cependant à détruire le cookie de session, qui a permis de propager l'identifiant de la session. La fonction `setcookie()` permet de parvenir à nos fins.

## **Fonctionnalité : recherche**

Une fois encore, cette fonctionnalité fait appel à notre Base de Données. Un formulaire de recherche est disponible sur la page d'accueil du site, qui permet de rechercher une enchère par son nom. Lorsque l'utilisateur valide sa recherche, un fichier .php s'occupe du traitement de la donnée. Une connexion s'ouvre avec la base (pg\_connect()), et une requête (pg\_query) de type SELECT est effectuée sur la table des Objets\_EnVente. Si une enchère possède la recherche dans son nom, elle est récupérée avec ses informations. La connexion à la base est coupée (pg\_close()) et l'utilisateur verra sur son écran les enchères en cours correspondant à sa recherche.

## **Fonctionnalité : tri**

L'interface possède, à côté du formulaire de recherche, plusieurs options de tri. Lors de la sélection de ces options de tri, la liste d'enchères est mise à jour et triée selon la méthode choisie. Ainsi, si, par exemple, la méthode de tri sélectionnée est le prix (enchère actuelle), le fichier .php récupérera la liste des enchères depuis la BDD (méthode SELECT) en les triant selon leur enchère actuelle, puis enverra le résultat au client.

## **Fonctionnalité : enchérir**

Lorsqu'un utilisateur souhaite déposer une enchère supérieure à celle précédente (bouton « Enchérir »), il fait appel à un fichier .php. Ce fichier va permettre de récupérer le montant rentré par l'utilisateur. Si la valeur entrée dans la case est correcte (pas de caractères, non vide, format correct), une requête de type SELECT est alors exécutée afin de récupérer le montant actuel de l'enchère. Deux scénarios peuvent alors survenir :

- Si le montant entré par l'utilisateur est supérieur à l'enchère actuelle, une requête UPDATE est exécutée et remplace deux champs au niveau de la table Objets\_EnVente: la valeur de l'enchère et l'enchérisseur actuel. La table utilisateurs est également mise à jour, car il faut réduire la quantité de crédits de l'utilisateur. Un message est envoyé au client lui indiquant qu'il est désormais l'enchérisseur actuel.
- Si le montant entré par l'utilisateur est inférieur à l'enchère actuelle, un message est envoyé au client lui indiquant que le prix entré n'est pas suffisant. Ses crédits ne sont pas déduits.

## **Fonctionnalité : affichage des informations des objets**

Lorsqu'un utilisateur clique sur un objet, il a accès à une nouvelle page qui affiche des informations générales sur l'objet : sa description. Ces informations sont récupérées avec un fichier PHP depuis la Base de Données.

## **Fonctionnalité : mise en vente**

Cette fonctionnalité est appelée lorsqu'un utilisateur souhaite mettre un objet en vente. Un formulaire est rempli par l'utilisateur, qu'il doit ensuite valider (bouton « Submit »). Lorsque la saisie est validée, le PHP vérifie que tous les champs obligatoires sont correctement remplis. Si cette vérification est correcte, une requête SQL de type INSERT INTO est envoyée à la Base de Données, et rajoute une ligne à la table objets en vente.

Bien sûr, si un des champs est incorrect, le formulaire est rechargé avec une indication de la position de ou des erreurs.

### **Fonctionnalité : fin de l'enchère**

Lorsqu'une enchère se termine (délai terminé par exemple), plusieurs opérations doivent être réalisées : le débit de l'acheteur, le crédit du vendeur, l'enchère devient inactive ...

### **Fonctionnalité : récupérer informations compte**

Lorsqu'un utilisateur souhaite consulter son compte ou consulter le compte de quelqu'un d'autre, ces informations sont directement tirées de la Base de Données. Cela signifie qu'une requête récupère directement ces données via un SELECT sur la table Utilisateurs.

### **Fonctionnalité : supprimer enchères**

Un vendeur pourrait souhaiter supprimer une enchère, ainsi il en a la possibilité via un bouton. Le PHP va d'abord faire une requête SELECT pour regarder si quelqu'un a enchéri sur l'objet : si c'est le cas, l'enchère ne pourra pas être supprimée sauf si l'action est réalisée par l'administrateur. Si la suppression est possible, une requête DELETE est envoyée et supprime la ligne de la table des objets en vente

### **Fonctionnalité : notifications évolutions des enchères**

**Cette fonctionnalité ne fait pas l'objet d'un fichier .php** : il s'agit de l'envoi d'un message lorsque notre offre se fait surenchérir.

### **Fonctionnalité : achat immédiat**

Le vendeur peut ou non mettre en place un prix « achat immédiat ». Cela signifie que si l'utilisateur enchéri un montant égal au prix d'achat immédiat, il remporte immédiatement l'enchère. Au niveau SQL, la ligne de la table Objets\_EnVente est mise à jour avec son attribut 'actif' qui devient 0. Le vendeur est crédité du montant (UPDATE de sa ligne) et l'acheteur est débité du montant (UPDATE de sa ligne).

### **Fonctionnalité : liste de souhaits**

Un utilisateur peut mettre un objet dans sa liste de souhaits (action via un bouton). Cette action génère au niveau PHP une requête UPDATE sur l'objet utilisateur afin de mettre à jour sa ligne et ajouter le nom de l'objet qu'il souhaite acheter.

Si l'utilisateur souhaite consulter cette liste, une requête SELECT est envoyée afin de récupérer le nom des objets, et fait ensuite appel à la fonctionnalité de recherche afin d'afficher la liste des objets en vente possédant ce nom.

### **Fonctionnalité : créditer compte**

Un utilisateur peut créditer son compte s'il le souhaite, à l'aide d'une monnaie virtuelle. Cette action génère une requête SQL UPDATE sur la table utilisateur.

### **Fonctionnalité : supprimer utilisateurs**

Cette fonctionnalité est réservée à un compte administrateur. Une requête SQL de type DELETE est envoyée à la Base de Données afin de supprimer la ligne de l'utilisateur concerné.

## 4. Répartition des tâches

### 4.1. JérémY Michel

- Réalisation des scripts Javascript (fonctions check, check2 et ToFade dans signcontent.html, et activation du checkTimestamps dans content)
- Réalisation de la partie PHP du site (partie dynamique)

### 4.2. Xavier Tauran

- Réalisation de toutes les pages liées au compte, et du CSS lié
- Réalisation de la page Acheter crédits
- Mise en place des différentes éléments (navbar, barre de recherche/tri)

### 4.3. Emmanuel Menat

- Réalisation de la Base de Données
- Réalisation des formulaires d'inscription, de vente

### 4.4. Mathieu Alcaina

- Réalisation des requêtes SQL
- Réalisation des pages HTML d'accueil (+enchères en cours)