

Electrouvetout

1 - Présentation

A - Contexte

Electrouvetout est un site e-commerce de produits électronique tel que les claviers, souris, caméras, écrans et tout ce qui s'apparente à de l'électronique. Le but d'Electrouvetout est que vous trouvez tout ce dont vous avez besoin en matière d'électronique.

B - Mission

Electrouvetout est un site e-commerce que nous avons créé en travail d'équipe. Le but final est d'avoir un site e-commerce professionnel et à la fois facile à comprendre pour les clients, les vendeurs et l'administrateur.

Il y a la possibilité d'avoir un compte client. Ce dernier peut donner son avis sur un article. Il peut aussi supprimer son avis. Il peut commander un article, avoir un accès à ses anciennes commandes avec facture à l'appui générée en XML et modifier ses informations personnelles ainsi que la possibilité de supprimer son compte afin de pouvoir effacer son empreinte numérique.

Pour chaque compte client, on doit avoir son nom, prénom, mail, mot de passe, adresse, ville et code postal.

Le client peut aussi avoir la possibilité que s'il oublie son mot de passe, il peut faire une demande afin de le récupérer.

Pour les avis, on doit seulement voir le prénom du client et non son nom. On doit voir la date et le commentaire.

Un compte client peut aussi devenir un compte vendeur. Ce dernier doit comporter un nom vendeur, un numéro de SIRET, un mail, un mot de passe, une adresse, une ville et un code postal.

Le compte vendeur a accès à un tableau de bord propre à chaque vendeur. Dans ce tableau de bord, il peut seulement ajouter un nouvel article. Il est seulement un intermédiaire entre le client et Electrouvetout. Il "loue" du stockage dans l'entrepôt de Electrouvetout et vend ses articles rattachés au vendeur.

Pour chaque article, il doit y avoir un type, une image, un prix, une quantité en stock, la capacité du stock pour cet article, et une description. Le vendeur ne peut pas fixer la capacité en stock, seul l'administrateur peut le faire. Le vendeur peut néanmoins définir la quantité qu'il souhaite vendre.

Pour le stockage des articles, on doit avoir sa quantité en stock, l'identifiant de l'article et la capacité de stockage possible pour cet article.

Pour chaque type on doit avoir le nom du type ainsi que le nombre d'articles par type. On veut aussi un préfixe qui sera généré automatiquement et qui soit différent. Par exemple, pour le type processeur et les 2 préfixes ne peuvent pas être tous les 2 "PR-", l'un sera "PR-" l'autre "PO-".

Le compte administrateur a accès à tous et peut tout gérer. Il peut supprimer n'importe quel article, n'importe quel avis. Il peut ajouter un type d'article à vendre, par exemple un type carte graphique pour des articles de type carte graphique comme une RTX 4080 qui est de type carte graphique. Il peut voir les commandes qui sont en cours payées. Une fois la commande payée, il peut l'expédier.

Il peut aussi avoir le détail de chacune des commandes, comme le numéro de commande, l'identifiant article, le prix unitaire de chacun des articles et la quantité achetée. Chaque commande est rattachée à un identifiant client et un mail. Pour chaque commande on doit voir son prix total, la date et le statut de la commande (En cours, Payer, Expédier).

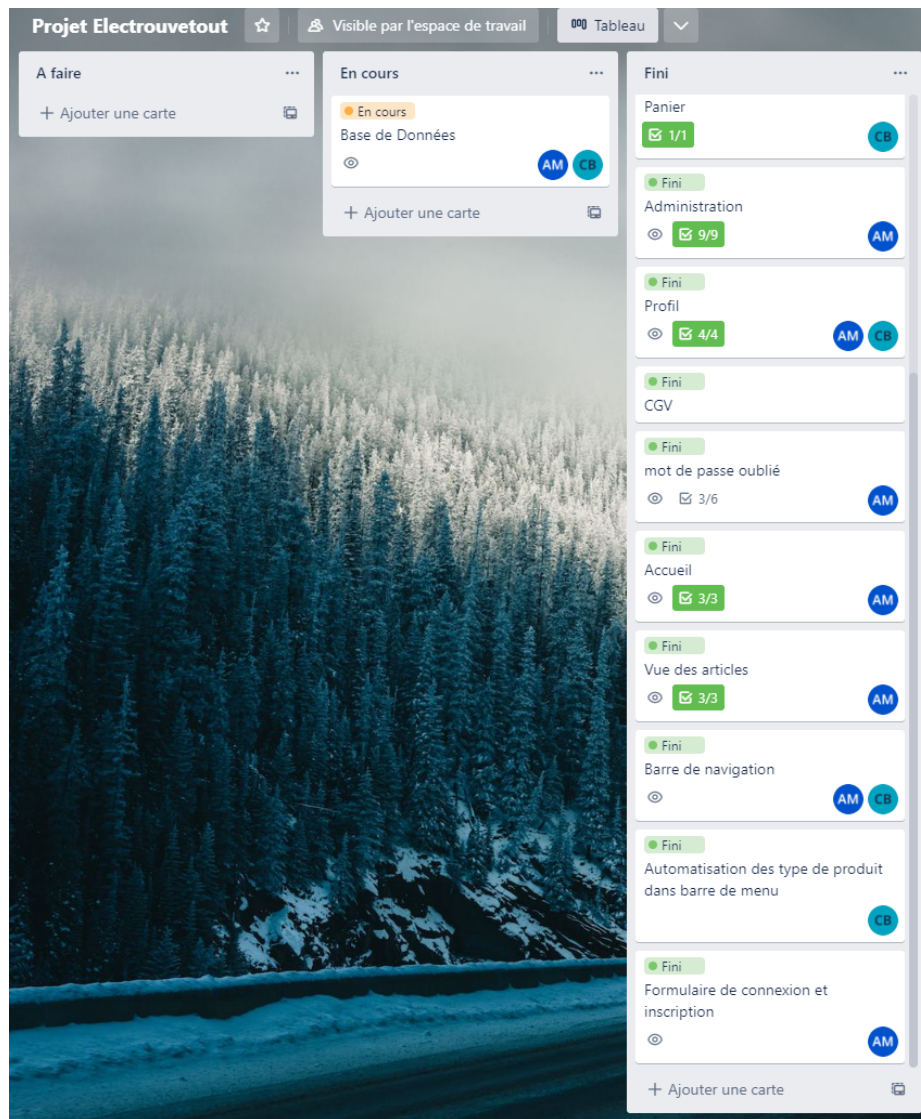
L'administrateur a accès à des logs qui répertorient toutes les actions réalisées par les clients. Les logs renvoient le mail de la personne qui a réalisé l'action, l'adresse IP, l'action réalisée, la date et le niveau de danger pour le site. Par exemple, un utilisateur se connectant à son compte affichera "client@client1.fr s'est connecté avec succès le 19-03-2023 à 14:05:38 avec l'adresse IP 178.65.36.1 de niveau de dangerosité basse". Il peut publier ou non un article, il peut les modifier ou supprimer.

Il peut voir qui vend les articles. Les informations qu'il peut voir sur le vendeur sont l'identifiant vendeur, l'identifiant utilisateur car un vendeur est obligatoirement un client, son nom vendeur, son numéro de SIRET, son mail, l'adresse du siège social du vendeur (s'il en possède un) ou alors l'adresse client qui est du coup son adresse vendeur. L'administrateur peut supprimer un vendeur si ce dernier ne respecte pas les conditions d'utilisation ainsi que les conditions générales de ventes. Il peut voir les types d'articles qu'il y a et le nombre d'articles par type. L'administrateur peut accéder aux informations des utilisateurs dans la limite du respect des données personnelles, tel que son identifiant client, son mail et la date de création.

2 - Réalisation

Avant de commencer, nous avons mis en place un Trello et créé un projet sur bitbucket afin de se partager les tâches. Lien du bitbucket :

- <https://bitbucket.org/alexandremotte/electrouvetout/src/master/>



| | | | | |
|--|----------|---------|--------------------------------------|------------|
| | 17 Xe-La | b5a9034 | connexion à la base + structure site | 2023-01-25 |
| | 17 Xe-La | e9e34da | ajout du bootstrap et de jquery | 2023-01-25 |
| | 17 Xe-La | 32f54f9 | ajout du script de la bdd | 2023-01-25 |
| | 17 Xe-La | 0031357 | test2 | 2023-01-25 |
| | Benoit | 5ef712d | test | 2023-01-25 |
| | 17 Xe-La | ecf9405 | ajout index | 2023-01-25 |

[Prev](#) [Next](#)

Une base de données à aussi été créée. En voici le MCD de cette dernière. (cf. annexe 1).

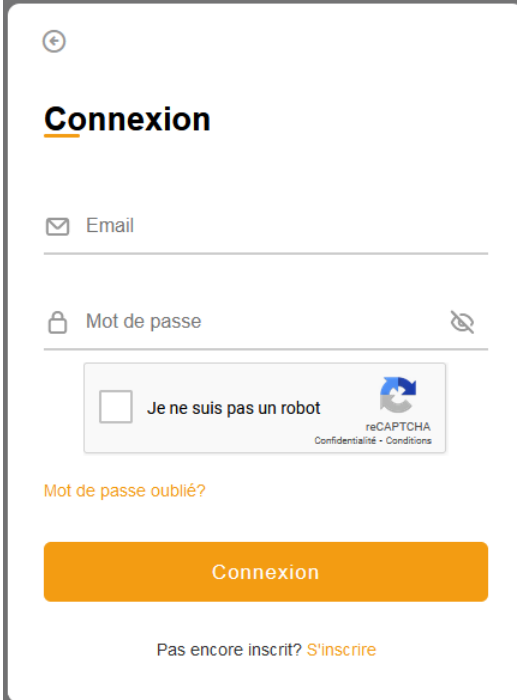
Cette documentation a été mise à jour au fur et à mesure de la création du site puis finalisée complètement. La suite de cette documentation est la version finalisée.

La partie A à H a été faite par MOTTE Alexandre et la partie I à P a été faite par CLAVERIE Benoit.

Pour le bon déroulé du test du site il est important de désactiver le proxy.

A - Formulaire de connexion et inscription

Pour commencer j'ai donc commencé par réaliser un formulaire de connexion et d'inscription. J'ai décidé de mettre les 2 formulaires sur la même page. Si on veut se connecter le formulaire d'inscription sera caché et si on veut s'inscrire, le formulaire de connexion sera caché. J'ajoute le captcha, mais il y a une erreur lors de l'intégration du captcha. J'ai trouvé la solution plus tard en mettant à jour le fichier JavaScript du captcha.



The image shows a login form with the following elements:

- Header: **Connexion**
- Input field: Email (with an envelope icon)
- Input field: Mot de passe (with a lock icon and a visibility toggle icon)
- reCAPTCHA: A checkbox labeled "Je ne suis pas un robot" and the reCAPTCHA logo with "Confidentialité - Conditions" below it.
- Link: [Mot de passe oublié?](#)
- Button: **Connexion** (orange)
- Footer: Pas encore inscrit? [S'inscrire](#)

Pour la partie inscription, je l'ai divisé en 2 parties. La première en demandant d'abord le nom, prénom, mail et mot de passe du client. Une fois ceci validé, le client est redirigé vers un 2^{ème} formulaire pour y mettre son adresse, sa ville et son code postal.

⊖


Inscription

👤 Nom

👤 Prénom

✉ Email

🔒 Mot de passe

🔒 Confirmer mot de passe 

J'accepte toutes les conditions

[S'inscrire](#)

Déjà membre? [Se connecter](#)

Une fois ce formulaire créé je l'ai relié à la base de données. Pour le formulaire de connexion, j'ai protégé les champs d'entrée en limitant le nombre de caractères et en spécifiant quel type de caractère je voulais. Pour cela j'ai utilisé les variables minlength et maxlength ainsi que pattern. J'ai aussi spécifié les champs obligatoires. (cf annexe 2)

Dans le fichier qui permet de vérifier si les identifiants sont bons, j'ai aussi réalisé une protection face aux injections SQL et XSS en utilisant htmlentities. Pour la connexion j'ai décidé mettre certaines variables qu'on aura besoin dans des sessions (\$_SESSION) tel que l'email, le mot de passe ainsi que l'identifiant utilisateur. Si l'utilisateur est un administrateur alors la session "admin" s'active. (cf. annexe 3)

J'ai ensuite ajouté la possibilité de récupérer son mot de passe. J'ai donc créé à nouveau un formulaire. Dans ce formulaire, on y rentre son mail. Si le mail existe dans la base de données alors le mail est envoyé avec un nouveau mot de passe.

L'utilisateur se connecte avec ce nouveau mot de passe. Il lui est ensuite spécifié qu'il doit le changer pour plus de sécurité.

B - Mise en place de la barre de navigation

J'ai donc mis en place une barre de navigation responsive et compréhensible par l'utilisateur. J'ai utilisé une barre de navigation Bootstrap, que j'ai ensuite personnalisée afin qu'elle soit adaptée au site.

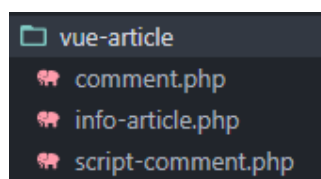


J'ai donc effectué ceci et l'autre personne avec qui je travaillais a fait la liaison de la barre de recherche à la base de données, pour que cela puisse être géré avec de l'ajax. Il a aussi géré la bulle du panier pour que celle-ci monte ou descend selon le nombre d'articles que l'utilisateur possède dans son panier.

C - Passage du site entier en MVC (Model Vue Controller)

On a donc décidé de passer le site en MVC pour de l'optimisation. Cela va nous permettre de pouvoir bosser sur le backend et à la fois le frontend sur la même page sans qu'il y est de conflits. Cela va nous permettre de séparer les données des méthodes qui les utilisent.

Donc pour le MVC de notre site on décide de créer un dossier vue où dedans on y répertorie des sous-dossiers avec le nom des pages et le module de chacune des pages. Par exemple pour la page "vue-article" (là où l'on voit les informations sur un article lorsqu'on clique dessus), il y a une section sur les informations de l'article, sur les commentaires et une autre section pour le script ajax des commentaires.



Dans la page vue-article, nous avons donc d'afficher ceci :

```
pages > vue-article.php > html > body
1  <?php
2
3  session_start();
4
5  require("../form/database.php");
6  require("../controller/requestSQL.php");
7
8  $num = $_GET["article"];
9
10 $detailsArticle = "";
11
12 foreach ($Articles as $article => $id) {
13
14     if ($id["id"] == $num) {
15         $detailsArticle = $id;
16     }
17 }
18
19 if ($detailsArticle == "") {
20     header("Location: ../index.php");
21 }
22
23 $selNomVendeur = $connexion->prepare("SELECT vendeur.nomEntreprise FROM vendeur, articles WHERE ? = vendeur.id");
24 $selNomVendeur->execute(array($detailsArticle["idVendeur"]));
25 $tabNomVendeur = $selNomVendeur->fetchAll(PDO::FETCH_ASSOC);
26
27 ?>
28
29 <!DOCTYPE html>
30 <html lang="fr">
31 <head>
32     <meta charset="UTF-8">
33     <meta http-equiv="X-UA-Compatible" content="IE=edge">
34     <meta name="viewport" content="width=device-width, initial-scale=1.0">
35     <title><?php echo $detailsArticle["nomArticle"]; ?></title>
36
37     <?php include "../vues/pages/link-css.php"; ?>
38
39 </head>
40 <body>
41
42     <?php
43     include "../vues/pages/nav.php";
44     include "../vues/pages/vue-article/info-article.php";
45     include "../vues/pages/vue-article/comment.php";
46     include "../vues/pages/footer.php";
47     include "../vues/pages/script-js.php";
48     include "../vues/pages/vue-article/script-comment.php";
49     ?>
50
51 </body>
52 </html>
```

C'est plus un MVC pour séparer les différentes sections de la page qu'un réel MVC. La seule chose qui nous a obligé à utiliser les requêtes en haut de page comme au-dessus est tout simplement que certaines pages ne fonctionnaient pas du tout avec le modèle MVC. Suite à un délai assez court on a préféré faire comme ceci. Cela étant dit nous utilisons quand même des fichiers php spécialement pour des requêtes, comme c'est le cas pour la partie administrateur où j'ai ajouté toutes les requêtes pour les valeurs des tableaux séparément dans un ce fichier (cf. annexe 4).

D - Mise en place de la page d'accueil

J'ai donc mis en place la page d'accueil. Cette dernière est importante car elle donne directement l'image que l'on veut renvoyer du site. J'ai donc effectué plusieurs fois des changements sur cette page d'accueil. Sur la page d'accueil, j'ai voulu expliquer ce qu'était Electrouvetout. On m'a fait comprendre que ce n'était pas dans une page d'accueil que nous mettions ce genre de messages. Je me suis donc rapatrié sur le fait de présenter les types les plus courants et communs dans des cards (cf. annexe 5).

Une fois que j'ai fait ces cards (cartes) j'ai mis une seconde section où dans cette dernière j'y affiche les articles les plus récents (cf. annexe 6).

E - Administration

Une fois la mise en place de la base du site, il faut bien une personne qui puisse ajouter et/ou supprimer des commentaires, des articles, voir les commandes passées etc. J'ai donc mis en place un dashboard pour l'administrateur. Comme on nous l'a demandé dans le cahier des charges, il faut que l'administrateur puisse avoir et faire tout ce qui est possible.

J'ai donc commencé par faire la page d'accueil du Dashboard. Dans cette page d'accueil j'y mets des sections où l'on aperçoit les 5 dernières actions qui se sont passées sur le site, les 5 derniers articles et un affichage qui permet de savoir s'il y a un problème sur le site (cf. annexe 7).

J'ai ensuite créé la page des logs. Sur cette page j'y affiche un tableau avec les 20 derniers logs.

| Email | Action | Date | IP | Niveau |
|---------------------|---|---------------------|-----------|--------|
| elec-admin@admin.fr | L'utilisateur s'est authentifié avec succès. | 2023-04-11 10:45:05 | 127.0.0.1 | low |
| elec-admin@admin.fr | L'administrateur a publié un article (PR-19412). | 2023-03-31 15:31:31 | 127.0.0.1 | low |
| elec-admin@admin.fr | L'administrateur a retiré la publication d'un article (PR-19412). | 2023-03-31 15:30:56 | 127.0.0.1 | low |
| elec-admin@admin.fr | L'utilisateur s'est authentifié avec succès. | 2023-03-31 15:27:30 | 127.0.0.1 | low |
| elec-admin@admin.fr | L'utilisateur n'a pas rentré les bonnes informations. | 2023-03-31 15:26:48 | 127.0.0.1 | medium |
| elec-admin@admin.fr | L'utilisateur a échoué la vérification du captcha. | 2023-03-28 14:49:29 | 127.0.0.1 | medium |
| elec-admin@admin.fr | L'utilisateur a échoué la vérification du captcha. | 2023-03-28 14:49:16 | 127.0.0.1 | medium |
| elec-admin@admin.fr | L'administrateur a expédié la commande (C-99429287-39394). | 2023-03-19 11:52:23 | 127.0.0.1 | low |
| elec-admin@admin.fr | L'administrateur a expédié la commande (C-32001052-96426). | 2023-03-19 11:52:17 | 127.0.0.1 | low |
| elec-admin@admin.fr | L'administrateur a expédié la commande (C-39597001-96426). | 2023-03-19 11:51:54 | 127.0.0.1 | low |
| elec-admin@admin.fr | L'administrateur a expédié la commande (C-54588932-96426). | 2023-03-19 11:51:45 | 127.0.0.1 | low |
| elec-admin@admin.fr | L'administrateur a expédié la commande (C-53663409-96426). | 2023-03-19 11:51:30 | 127.0.0.1 | low |
| elec-admin@admin.fr | L'utilisateur s'est authentifié avec succès. | 2023-03-19 11:49:03 | 127.0.0.1 | low |
| foo-bar@example.com | L'utilisateur n'a pas rentré les bonnes informations. | 2023-03-09 11:29:34 | 127.0.0.1 | medium |
| foo-bar@example.com | L'utilisateur s'est inscrit avec succès. | 2023-03-09 11:29:34 | 127.0.0.1 | low |
| elec-admin@admin.fr | L'utilisateur n'a pas réalisé le captcha. | 2023-03-06 10:02:11 | 127.0.0.1 | medium |
| client1@cli.fr | L'utilisateur s'est authentifié avec succès. | 2023-03-02 15:48:55 | 127.0.0.1 | low |
| client1@cli.fr | L'utilisateur s'est authentifié avec succès. | 2023-02-24 14:06:29 | 127.0.0.1 | low |
| client@client.com | L'utilisateur n'a pas rentré les bonnes informations. | 2023-02-24 14:06:01 | 127.0.0.1 | medium |
| elec-admin@admin.fr | L'utilisateur s'est authentifié avec succès. | 2023-02-24 13:56:03 | 127.0.0.1 | low |

Affichage de 0 à 20 logs sur 522 enregistrés.

»»

Pour mettre en place ce système, j'ai donc mis un bout de code à chaque connexion ou tentative de connexion d'un client, ajout et/ou suppression d'articles, de commandes et de commentaires. Voici donc ce que j'ai mis. Cette fonction me permet de récupérer l'adresse IP de l'utilisateur.

```
function getRemoteIp()
{
    if (!ini_get("register_globals")) {
        return $_SERVER["REMOTE_ADDR"];
    } else {
        return @$_REMOTE_ADDR;
    }
    return "";
}

$ip = getRemoteIp();
```

Par exemple pour la connexion je récupère le login de l'utilisateur, donc son mail, qui tente de se connecter. Si la connexion est réussie alors j'exécute ce bout de code.

```
$remarksLog = "L'utilisateur s'est authentifié avec succès.";
$level = "low";

















































$ins = $connexion->prepare("INSERT INTO logs (idUser, action, date, ip, level) VALUES (?, ?, ?, ?, ?)");
$ins->execute(array($email, $remarksLog, $date, $ip, $level));
```

La variable "\$level" permet de spécifier le niveau de dangerosité de l'action réalisée. La variable "\$remarksLogs" permet de mettre l'action qui s'est réalisée. J'insère ensuite ceci dans la base de données dans la table "logs".

Si la connexion a échoué par exemple si l'utilisateur n'a pas rempli le captcha alors je fais la même chose mais je modifie le niveau de dangerosité et l'action réalisée.

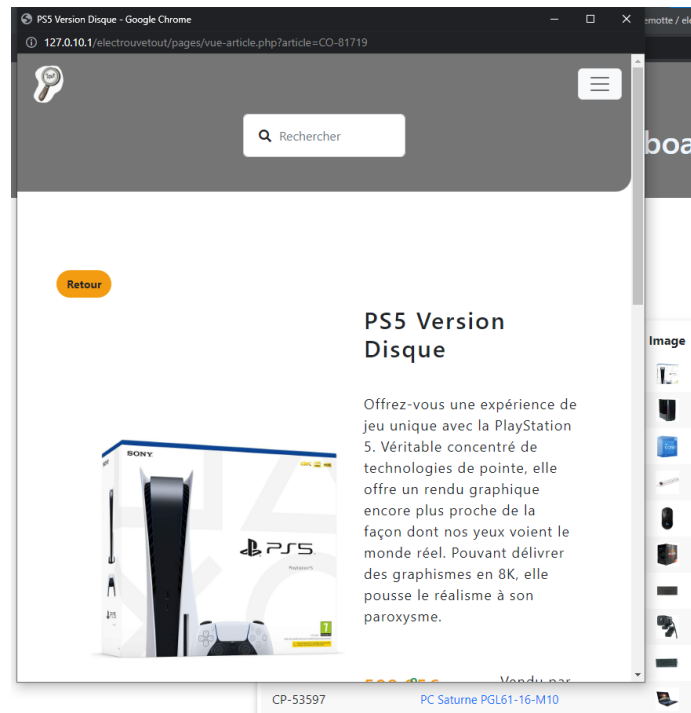
Je m'occupe ensuite de la mise en place de la page de gestion d'articles. Pareil que pour les logs, j'en affiche 20. S'il y en a plus, il y a un système de pagination.

[Ajouter un article](#)

| Identifiant | Nom | Image | Prix | Quantité | Type | Date | Vendeur | Publié | Modifier | Supprimer |
|-------------|--|---|----------|----------|------------|------------|----------------|---|---|---|
| CO-81719 | PS5 Version Disque |  | 599.95€ | 8 | Console | 2023-02-22 | Electrouvetout | Oui  |  |  |
| CP-21848 | PC Zenifier-SSD |  | 549.98€ | 25 | Ordinateur | 2023-02-22 | Electrouvetout | Oui  |  |  |
| PR-33145 | Intel Core i5-12400F (2.5 GHz / 4.4 GHz) |  | 218.95€ | 67 | Processeur | 2023-02-16 | EiEclair | Oui  |  |  |
| PL-18549 | Multiprise 5 prises avec interrupteur |  | 12.95€ | 535 | Prise | 2023-02-09 | Electrouvetout | Oui  |  |  |
| SR-63801 | Logitech G Pro Wireless |  | 109.96€ | 43 | Souris | 2023-02-09 | EiEclair | Oui  |  |  |
| PR-19412 | AMD Ryzen 5 5600X Wraith Stealth (3.7 GHz / 4.6 GHz) |  | 212.95€ | 53 | Processeur | 2023-02-09 | EiEclair | Oui  |  |  |
| CL-37887 | Targus Clavier Bluetooth multi-plateforme KB55 |  | 25.95€ | 8 | Clavier | 2023-02-08 | Electrouvetout | Oui  |  |  |
| CM-20145 | Logitech StreamCam (HD) |  | 149.95€ | 53 | Caméra | 2023-02-07 | Electrouvetout | Oui  |  |  |
| CL-66927 | Clavier compact USB (AZERTY, Français) |  | 12.95€ | 22 | Clavier | 2023-02-07 | EiEclair | Oui  |  |  |
| CP-53597 | PC Saturne PGL61-16-M10 |  | 1279.94€ | 9 | Ordinateur | 2023-02-07 | Electrouvetout | Oui  |  |  |
| EC-90207 | ASUS 27 |  | 399.95€ | 32 | Ecran | 2023-02-07 | EiEclair | Oui  |  |  |
| SR-15504 | Logitech G G502 Lightspeed |  | 119.95€ | 45 | Souris | 2023-02-07 | Electrouvetout | Oui  |  |  |

Affichage de 0 à 20 articles sur 12 enregistrés.

Dans ce tableau, il est possible de voir, l'identifiant de l'article, son nom, l'image de l'article, le prix, la quantité en stock, le type, la date d'ajout de l'article, le vendeur, s'il est publié, le modifier et le supprimer. Lorsqu'on clique sur le nom de l'article, une fenêtre pop-up de la page de l'article afin de voir comment cela rends sur la page.



L'administrateur peut publier ou non l'article, ce n'est pas le vendeur qui est maître de la publication de son article. L'administrateur peut aussi modifier l'article (cf. annexe 8) et le supprimer. L'administrateur peut ainsi ajouter un article et le publier.

Type

clavier ▾

Nom de l'article

Choisir une photo (png, jpeg et jpg)

Choisir un fichier Aucun fichier choisi

Prix

Quantité en stock

Capacité en stock

Description

Créer un article Annuler

Juste en dessous du tableau d'article, j'ai décidé d'y mettre la possibilité de voir les types d'articles ainsi que d'en ajouter.


Ajouter un type

| Identifiant | Nom | Nombre d'articles de ce type | Préfixe | Supprimer |
|-------------|------------|------------------------------|---------|-----------|
| 1 | clavier | 2 (Voir) | CL- | |
| 2 | souris | 2 (Voir) | SR- | |
| 3 | caméra | 1 (Voir) | CM- | |
| 4 | ecran | 1 (Voir) | EC- | |
| 5 | prise | 1 (Voir) | PL- | |
| 6 | ordinateur | 2 (Voir) | CP- | |
| 7 | processeur | 2 (Voir) | PR- | |
| 8 | console | 1 (Voir) | CO- | |

Affichage de 0 à 20 types sur 8 enregistrés.

On peut donc voir pour chaque type, son nom, le nombre d'articles par type ainsi que son préfixe. On peut aussi supprimer le type. La suppression n'est pas possible si des articles sont existants avec ce type. Pour le nombre d'articles par type, on peut voir les articles correspondant à ce dernier. Par exemple pour le type clavier, on peut

voir qu'il y a 2 articles de ce type. Si l'on clique pour pouvoir les voir, nous voyons ce tableau :

| Identifiant | Nom | Image | Prix | Quantité | Modifier | Supprimer |
|-------------|--|---|-------|----------|---|---|
| CL-37887 | Targus Clavier Bluetooth multi-plateforme KB55 |  | 25.95 | 8 |  |  |
| CL-66927 | Clavier compact USB (AZERTY, Français) |  | 12.95 | 22 |  |  |

Affichage de 0 à 20 types sur 2 enregistrés.

Lorsque l'administrateur ajoute un nouveau type, un script s'effectue qui récupère les 2 premières lettres du type, les mets en majuscule et les définit comme préfixe dans la base de données.

```
$nom = $_POST["nom"];
$name = $_POST["nom"];



$selectPrefix = $connexion->prepare("SELECT prefixe FROM types");
$selectPrefix->execute();
$tabPrefix = $selectPrefix->fetchAll(PDO::FETCH_ASSOC);

$nouv = "";

foreach ($tabPrefix as $key) {
    if (substr($key["prefixe"], 0, 2) == strtoupper(substr($nom, 0, 2))) {
        $nouv = $nom[0].$nom[2];
    } else {
        $nouv = strtoupper(substr($nom, 0, 2));
    }
}
```

L'administrateur peut aussi avoir les informations de n'importe quel client de Electrouvetout ainsi que n'importe quel vendeur. Les informations clients que possède l'administrateur sont dans la limite de la protection des données à caractère personnelles. Cela veut dire que l'administrateur peut seulement voir l'identifiant client, le mail et la date de création du compte.

Pour le vendeur, l'administrateur peut voir son identifiant vendeur, l'identifiant client auquel il est rattaché (un client peut être vendeur), le nom du vendeur, le numéro de SIRET, l'email, l'adresse, la ville et le code postal du vendeur. L'administrateur a la possibilité de supprimer un vendeur si ce dernier ne respecte pas les consignes d'utilisation et ne respecte pas les conditions générales de ventes.

| Identifiant vendeur | Identifiant utilisateur | Nom vendeur | SIRET | Email | Adresse | Ville | CP | Supprimer |
|---------------------|----------------------------|----------------|----------------|-----------------------|--------------------------|---------|-------|---|
| VD-3446169687 | USER-39394 | Electrouvetout | 54654896953362 | elec-admin@admin.fr | 8 Rue des Gargousses | Tarbes | 65000 |  |
| VD-8464727869 | USER-37559 | EIEclair | 87689568318112 | alexaigle65@gmail.com | 5 rue Georges Ledormeure | Pontacq | 64120 |  |

Affichage de 0 à 20 vendeur sur 12 enregistrés.

| Identifiant | Email | Date de création |
|-------------|-----------------------|---------------------|
| USER-37559 | alexaigle65@gmail.com | 2023-01-25 15:20:06 |
| USER-39394 | elec-admin@admin.fr | 2023-01-25 15:20:24 |
| USER-41957 | foo-bar@example.com | 2023-03-09 11:29:34 |
| USER-77790 | ogur@gmail.com | 2023-02-19 21:37:05 |
| USER-94631 | benoit@clv.fr | 2023-01-25 15:20:33 |
| USER-96426 | client1@cli.fr | 2023-01-27 15:20:38 |

Affichage de 0 à 20 utilisateurs sur 6 enregistrés.

L'administrateur est le seul à pouvoir expédier les commandes. Le vendeur est seulement l'intermédiaire entre Electrouvetout et le client. L'administrateur a accès à un tableau avec le numéro de commande. Si l'administrateur clique sur le numéro de commande alors il peut savoir ce que comporte la commande (cf. annexe 9). Dans les détails des commandes on peut voir les articles achetés avec une image de l'article, l'identifiant de l'article et si on clique dessus cela nous renvoie sur la page des détails de l'article, le prix unitaire de l'article et la quantité achetée. Pour les informations sur les commandes, l'administrateur voit ensuite l'identifiant client, l'email client, le prix total, la date de la commande, le statut de la commande. Si la commande est "En cours" l'administrateur ne peut pas expédier la commande mais il peut la visualiser. Si la commande est en statut "Payer" l'administrateur a un bouton qui apparaît et il peut expédier la commande. La commande est ensuite en statut "Expédier".












| Numéro de commande | Identifiant client | Email client | Prix total | Date | Statut de la commande | Expédier |
|--------------------|--------------------|-----------------------|------------|---------------------|-----------------------|----------|
| C-32003252-39394 | USER-39394 | elec-admin@admin.fr | 38.85€ | 2023-03-06 10:16:11 | En cours | |
| C-34019949-96426 | USER-96426 | client1@cli.fr | 25.95€ | 2023-03-02 15:52:55 | En cours | |
| C-99429287-39394 | USER-39394 | elec-admin@admin.fr | 149.95€ | 2023-02-24 13:57:16 | Expédié | |
| C-32001052-96426 | USER-96426 | client1@cli.fr | 599.95€ | 2023-02-24 11:13:02 | Expédié | |
| C-39597001-96426 | USER-96426 | client1@cli.fr | 149.95€ | 2023-02-24 10:51:32 | Expédié | |
| C-54588932-96426 | USER-96426 | client1@cli.fr | 7108.45€ | 2023-02-24 10:21:28 | Expédié | |
| C-53254250-37559 | USER-37559 | alexaigne65@gmail.com | 599.95€ | 2023-02-22 16:09:13 | Expédié | |
| C-74213859-39394 | USER-39394 | elec-admin@admin.fr | 549.80€ | 2023-02-21 09:30:27 | Expédié | |
| C-61102144-77790 | USER-77790 | ogur@gmail.com | 9943.45€ | 2023-02-19 21:38:12 | Expédié | |
| C-53663409-96426 | USER-96426 | client1@cli.fr | 103.80€ | 2023-02-17 16:15:43 | Expédié | |
| C-16699794-96426 | USER-96426 | client1@cli.fr | 1807.64€ | 2023-02-17 10:53:29 | Expédié | |
| C-68959360-96426 | USER-96426 | client1@cli.fr | 93283.30€ | 2023-02-16 16:33:32 | Expédié | |
| C-88521737-37559 | USER-37559 | alexaigne65@gmail.com | 109.96€ | 2023-02-16 15:00:24 | Expédié | |
| C-18095212-39394 | USER-39394 | elec-admin@admin.fr | 25.90€ | 2023-02-14 09:49:15 | Expédié | |

Affichage de 0 à 20 commandes sur 14 en cours ou payées.

F - Vendeur

Pour la partie vendeur, j'ai effectué un tableau de bord ressemblant à celui de l'administrateur. Mais beaucoup de choses sont différentes. Premièrement le vendeur ne voit pas les logs, il ne voit pas les commandes qui sont effectuées, il ne peut pas ajouter de types d'articles et ne peut pas gérer la capacité en stockage d'un article. Il peut seulement ajouter un article, modifier ses informations et voir les types d'articles.

[Ajouter un article](#)

| Identifiant | Nom | Image | Prix | Quantité | Type | Date | Publié | Modifier | Supprimer |
|-------------|--|---|----------|----------|------------|------------|--------|---|---|
| CO-81719 | PS5 Version Disque |  | 599.95€ | 8 | Console | 2023-02-22 | Oui |  |  |
| CP-21848 | PC Zenifier-SSD |  | 549.98€ | 25 | Ordinateur | 2023-02-22 | Oui |  |  |
| PL-18549 | Multiprise 5 prises avec interrupteur |  | 12.95€ | 535 | Prise | 2023-02-09 | Oui |  |  |
| CL-37887 | Targus Clavier Bluetooth multi-plateforme KB55 |  | 25.95€ | 8 | Clavier | 2023-02-08 | Oui |  |  |
| CP-53597 | PC Saturne PGL61-16-M10 |  | 1279.94€ | 9 | Ordinateur | 2023-02-07 | Oui |  |  |
| SR-15504 | Logitech G G502 Lightspeed |  | 119.95€ | 45 | Souris | 2023-02-07 | Oui |  |  |
| CM-20145 | Logitech StreamCam (HD) |  | 149.95€ | 53 | Caméra | 2023-02-07 | Oui |  |  |

Affichage de 0 à 20 articles sur 7 enregistrés.

Le vendeur ne peut pas publier les articles qui vend. Par exemple, s'il ajoute une poêle de type console, l'administrateur peut donc vérifier si le vendeur ne vend pas n'importe quoi. L'administrateur peut ensuite supprimer l'article et informer le vendeur que ce n'est pas un article correspondant aux valeurs du site.

G - Conditions Générales de Ventes

J'ai donc mis en place un système juridique en m'aidant de ressources sur Internet pour mettre en place des Conditions Générales de Ventes (CGV).

Les présentes conditions générales de vente s'appliquent à toutes les ventes conclues sur le site Internet www.electrouvetout.fr, sous réserve des conditions particulières indiquées dans la présentation des produits.

Editeur du site

SARL Electrouvetout Numéro de SIRET :
12334455600034

Responsables éditorial : Alexandre MOTTE et Benoît
CLAVERIE

8 Rue des Gargousses, 65000 TARBES

Téléphone : 09 72 21 25 07

Email : contact@electrouvetout.fr

Site Web : www.electrouvetout.fr

Hébergement

Hébergeur : SARL Electrouvetout

8 Rue des Gargousses, 65000 TARBES

Site Web : www.electrouvetout.fr

Développement

SARL Electrouvetout

Adresse : 8 Rue des Gargousses, 65000 TARBES

Site Web : www.electrouvetout.fr

Ceci est obligatoire pour tout site marchand. Ces conditions sont divisées en plusieurs parties, les voici :

- Caractéristiques essentielles des produits et services vendus
- Prix
- Délai de disponibilité des produits
- Commande
- Livraison
- Modalités de paiement
- Droit de rétractation
- Garantie et droit de retour du produit (vice caché ou défautuosité)
- Conditions et délais de remboursement
- Réclamations du consommateur
- Propriété intellectuelle
- Responsabilité
- Données à caractère personnel
- Juridiction compétente et droit applicable

H - Réalisations de tests

J'ai donc effectué des tests d'intégrations sur les parties que j'ai faites. Le test d'intégration que j'ai effectué est avec Selenium. Je les ai effectués sur l'ajout d'un article à partir du tableau de bord de l'administrateur.

J'ai eu une erreur lorsque le test a dû être effectué car Selenium génère des faux chemins pour les images et donc ne peut pas ajouter d'image mais sur le reste du test je n'ai eu aucune erreur.

| Log | Reference |
|--|---|
| 2. setWindowSize on 1936x1176 | OK |
| 3. click on css=.rounded-circle | OK |
| 4. click on linkText=Profil | OK |
| 5. click on css=.aces-dash img | OK |
| 6. click on linkText=Articles | OK |
| 7. click on linkText=Ajouter un article | OK |
| 8. click on name=choix | OK |
| 9. select on name=choix with value label=souris | OK |
| 10. click on id=nom | OK |
| 11. type on id=nom with value souris | OK |
| 12. click on name=image | OK |
| 13. type on name=image with value C:\fakepath\bug-informatique.png | Failed: {"code":-32000,"message":"Not allowed"} |

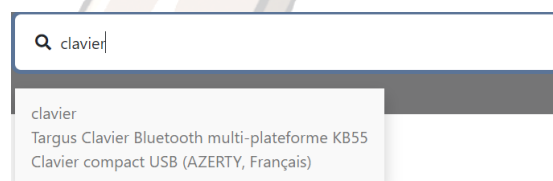
'Ajout article Electrouvetout Admin' ended with 1 error(s)

I - Barre de recherche côté backend

La barre de recherche permet d'accéder à la page d'un article spécifique ou bien à la page d'une catégorie d'article.

Par exemple :

Si l'on écrit « clavier » dans la barre de recherche.



Il y a trois résultats. Le premier amène sur la page de tous les articles de type clavier et les deux autres amènent sur leurs pages respectives de clavier avec leurs description, prix...

Pour que la barre de recherche retourne le résultat attendu « clavier » et non autre chose. Des requêtes SQL sont exécutées grâce à l'ajax dans le dossier : vues/pages/nav.php.

Le code Ajax :

```
<script >
$(document).ready(function(){
    $('#search').keyup(function(){
        var mot = $(this).val();
        if (mot != ""){
            $.ajax({
                type: 'GET',
                url: '../controller/barreRecherche.php',
                data: 'mot2=' + encodeURIComponent(mot),
                success: function(data){
                    if(data != ""){
                        $('#suggestions').addClass("dropdownperso-content");
                        if ($('#search').keydown()){
                            $('.suggest').remove();
                        }
                        console.log(data);
                        $('#suggestions').append(data);
                        $('#search').keyup(function(event){
                            if (event.which == 13){
                                $('.suggest').get(0).click();
                            }
                        })
                    } else {
                        document.getElementById('suggestions').innerHTML = "<div class='suggest'>Aucun résultat</div>";
                    }
                }
            })
        }
        if(mot.length == 0) {
            $('.suggest').remove();
            $('#suggestions').removeClass("dropdownperso-content");
        }
    });
});
</script>
```

Script PHP exécuté :

```
<?php
include("../form/database.php");

if (isset($_GET["mot"])) {
    $choix = $_GET["mot"];
}
if (isset($_GET["mot2"])) {
    $choix = $_GET["mot2"];
}

$res = $connexion-> prepare("SELECT nom FROM types WHERE nom like '%" . $choix . "%' ");
$res->execute();
$Nom = $res->fetchAll(PDO::FETCH_ASSOC);

foreach ($Nom as $key) {
    if (isset($_GET["mot"])) {
        echo "<a href='pages/articles.php?type=" . $key["nom"] . "' class='dropdown-item suggest'>". $key["nom"] . "</a>";
    }
    if (isset($_GET["mot2"])) {
        echo "<a href='../pages/articles.php?type=" . $key["nom"] . "' class='dropdown-item suggest'>". $key["nom"] . "</a>";
    }
}

$res = $connexion-> prepare("SELECT id,nomArticle FROM articles WHERE nomArticle like '%" . $choix . "%' ");
$res->execute();
$Nom2 = $res->fetchAll(PDO::FETCH_ASSOC);

foreach ($Nom2 as $key) {
    if (isset($_GET["mot"])) {
        echo "<a href='pages/vue-article.php?article=" . $key["id"] . "' class='dropdown-item suggest'>". $key["nomArticle"] . "</a>";
    }
    if (isset($_GET["mot2"])) {
        echo "<a href='../pages/vue-article.php?article=" . $key["id"] . "' class='dropdown-item suggest'>". $key["nomArticle"] . "</a>";
    }
}
if (empty($Nom) && empty($Nom2)) {
    echo "<div class='suggest'>Aucun résultat</div>";
}
?>
```

Le code Ajax permet dans ce cas-là qu'à chaque appui de touche sur le clavier d'exécuter le script PHP barreRecherche.php dans le dossier : controller\.

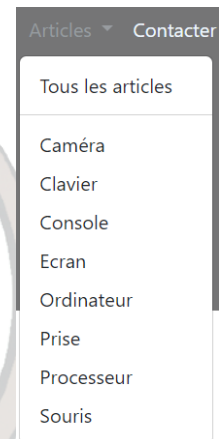
Cela va permettre d'obtenir tous les résultats possibles grâce aux 2 requêtes SQL qui vont fouiller les tables articles et types tout en réactualisant la liste des résultats sous la barre de recherche.

Tout en envisageant le cas qu'il n'y ait aucun résultat.

J - Barre de menu

Présence d'une liste déroulante lorsque l'on clique sur "Articles" :

L'administrateur du site peut rajouter des types d'articles, s'ensuit cette liste déroulante qui exécute une requête SQL pour obtenir tous les types d'articles afin de tous les afficher dans la liste déroulante. (Emplacement du code : vues/pages/nav.php Ligne 22)



Requête SQL utilisée : (Emplacement du code : controller/requestSQL.php Ligne 54)

```
//Création du tableau des types d'article par ordre alphabétique.  
$tabTypes = $connexion->prepare("SELECT * from types order by nom");  
$tabTypes->execute();  
$Types = $tabTypes->fetchAll(PDO::FETCH_ASSOC);  
//
```

Présence d'une bulle au niveau du panier pour savoir combien d'articles sont dans le panier.

Elle s'adapte en fonction du nombre d'articles qui se trouve dans le panier du client. Si le client est connecté, le PHP exécute une requête SQL pour connaître le nombre total d'articles que le client a ajoutés à son panier. (Emplacement du code : vues/pages/nav.php Ligne 45)

```
<?php
if($_SESSION["online"]==true){
    $tabCom = $connexion->prepare("SELECT sum(qte) as total from commandes, detailCommandes where commandes.
    idCommande=detailCommandes.idCommande AND idUser=? AND etatCommande=?");
    $tabCom->execute(array($_SESSION["idUser"],"En cours"));
    $tabCommande = $tabCom->fetchAll(PDO::FETCH_ASSOC);
    if ($tabCommande[0]["total"]==null){
        echo "0";
    } else {
        echo $tabCommande[0]["total"];
    }
} else {
    echo "0";
}
?>
```

K - Mise en place et gestion du panier :

Tout d'abord pour accéder au panier ou ajouter un article au panier il faut être connecté.

Il faut savoir que lorsque l'on ajoute un article au panier du script en Ajax est exécuté (Emplacement du code : vues/pages/vue-article/script-comment.php Ligne 39).

Ce dernier déclenche le script gestionPanier.php (Emplacement du code : controller/gestionPanier.php)

Ce script va gérer plusieurs choses :

- Créer une commande si l'utilisateur n'a pas de commande en cours. Si c'est le cas, il va créer un nouvel identifiant de commande unique pour ensuite l'ajouter à la base de données. Sinon si l'utilisateur a déjà une commande en cours, le code va récupérer et stocker son identifiant de commande.
- Ajouter l'article dans la table detailCommande qui regroupe tous les articles de toutes les commandes effectuées par les utilisateurs en fonction de leur identifiant commande et de leur identifiant utilisateur. Si l'article a déjà été ajouté à la table detailCommande de la commande en cours, il va alors augmenter la quantité de l'article sélectionné de 1.
- Augmenter automatiquement le prix total de la commande en cours du client en fonction du prix de l'article sélectionné.
- Diminuer le stock total de l'article que l'on ajoute à votre panier dans la table de stockage.

Depuis la page de visuel du panier l'utilisateur peut effectuer plusieurs choses.



Il peut directement modifier la quantité d'article identique qu'il veut commander avec cette case :

La quantité a pour minimum 1 et pour maximum le stock total de l'article. Cela s'effectue grâce à de l'Ajax (emplacement du code : pages/panier.php Ligne 137) qui appelle du PHP (emplacement du code : controller/adaptQTE-PrixTot.php). Ce code PHP va modifier le prix total de la commande, la quantité de l'article correspondant à son stock restant.

Quantité

1

Il peut retirer un article de son panier :

Supprimer



Cela correspond à la suppression de l'article voulu dans la table detailCommande qui correspond à l'identifiant commande en cours de l'utilisateur en exécutant un script PHP (emplacement du code : controller/supprimer-article.php). Il va aussi réactualiser le prix total de la commande.

Il peut vider son panier :

Vider le panier.

Cela correspond à la suppression de la commande en cours de l'utilisateur dans la table commande faite par un script PHP (emplacement du code : controller/supprCommande.php) qui est déclenché par de l'Ajax (emplacement du code : pages/panier.php Ligne 157).

Ces deux évènements de suppression utilisent tous deux une procédure stockée dans la base de données qui s'appelle gestionStock. Elle permet de rétablir le stock dans la table stockage des articles étant dans le panier du client lorsqu'il retire un article de sa commande ou bien supprime sa commande.

Pour le paiement des commandes nous avons décidé d'utiliser une solution proposée par PayPal.

Elle permet d'effectuer des paiements par carte bancaire et par PayPal. Pour tester les paiements par PayPal nous avons utilisé le portail développeur appelé SandBox de PayPal qui nous offre la liberté de payer les commandes facilement avec de l'argent fictif afin de vérifier le fonctionnement de l'application.

Lors du paiement de la commande de l'ajax s'exécute (emplacement du code : pages/panier.php Ligne 184) pour déclencher un script PHP (emplacement du code : controller/majEtat.php). Il permet de mettre à jour l'état de la commande en « Payer ». Une commande doit passer par 3 états dans l'ordre suivant :

- En cours : le client ajoute des articles au panier. Le client n'a pas encore payé.
- Payer : le client a payé sa commande.
- Expédier : la commande du client a été expédiée et il la recevra dans quelques jours.

L - Les triggers

Dans notre base de données il y a la présence de triggers qui vont permettre de gérer les suppressions automatiquement lors de la suppression d'un article, d'une commande et d'un utilisateur.

| | | | | | |
|----------------------------------|-------------|--------|----------|-----------|---------------|
| supprArticlesCom | articles | Éditer | Exporter | Supprimer | BEFORE DELETE |
| supprArticlesDeLaCommande | commandes | Éditer | Exporter | Supprimer | BEFORE DELETE |
| supprQte | articles | Éditer | Exporter | Supprimer | AFTER DELETE |
| supprUtilisateur | utilisateur | Éditer | Exporter | Supprimer | BEFORE DELETE |
| supprUtilisateurCmd | utilisateur | Éditer | Exporter | Supprimer | BEFORE DELETE |
| supprStockArticle | articles | Éditer | Exporter | Supprimer | BEFORE DELETE |

Ce sont tous des triggers qui vont s'exécuter soit avant ou soit après une suppression. Lors de la suppression les triggers vont supprimer ou modifier une ou plusieurs lignes dans une autre table.

Pour la suppression d'un article, 3 triggers vont s'exécuter :

- supprArticlesCom, il va supprimer tous les commentaires qui correspondent à l'article supprimé dans la table commentaire.
- supprQte, il va soustraire un à la quantité totale du type d'article auquel appartient l'article supprimé dans la table types.
- supprStockArticle, il va supprimer la ligne de stock qui correspond à l'article supprimé dans la table de stockage.

Pour la suppression d'une commande, 1 trigger va s'exécuter :

- supprArticlesDeLaCommande, il va supprimer le détail de la commande d'un client qui correspondent au numéro de commande supprimé dans la table détailCommandes.

Pour la suppression d'un utilisateur, 2 triggers vont s'exécuter :

- supprUtilisateur, si l'utilisateur à un statut de vendeur alors ses informations en temps que vendeur vont être supprimé dans la table vendeur.
- supprUtilisateurCmd, il supprime toutes les commandes faites par l'utilisateur dont le compte est supprimé dans la table commandes.

M - Gestion des commandes passé et actuelle

Les commandes passées :

| Vos dernières commandes. | | | |
|----------------------------------|------------|---------------------|---|
| Numéro de commande | Prix Total | Etat de la commande | Télécharger facture |
| C-16699794-96426 | 1807.64€ | Expédié | Facture Commande : C-16699794-96426 |
| C-32001052-96426 | 599.95€ | Expédié | Facture Commande : C-32001052-96426 |
| C-39597001-96426 | 149.95€ | Expédié | Facture Commande : C-39597001-96426 |
| C-53663409-96426 | 103.80€ | Expédié | Facture Commande : C-53663409-96426 |
| C-54588932-96426 | 7108.45€ | Expédié | Facture Commande : C-54588932-96426 |
| C-68959360-96426 | 93283.30€ | Expédié | Facture Commande : C-68959360-96426 |

Cela affiche toutes les commandes faites avec possibilité d'avoir un récapitulatif de la commande voulue en cliquant sur son numéro de commande.

La commande en cours :

 Vos commandes **en cours**.

| Numéro de commande | Prix Total |
|----------------------------------|------------|
| C-34019949-96426 | 233.55€ |

Afficher le numéro de la commande et son prix total. Si l'on clique sur le numéro de commande cela nous renvoie au panier.

N - Factures en XML

Stocker les factures en XML permet de stocker les informations en brut et de les enregistrer à chaque commande passée. Cela évite tout problème de modification ou de suppression à cause de la base de données.

Toutes les factures XML respectent la structure du fichier : xml/facture.dtd

```
<!ELEMENT commandes (commande*)>
<!ELEMENT commande (nom, prenom, mail, adresse-livraison, prixTotal, date, detailCommandes)>
<!ELEMENT nom (#PCDATA)>
<!ELEMENT prenom (#PCDATA)>
<!ELEMENT mail (#PCDATA)>
<!ELEMENT adresse-livraison (adresse, ville, codepostal)>
<!ELEMENT adresse (#PCDATA)>
<!ELEMENT ville (#PCDATA)>
<!ELEMENT codepostal (#PCDATA)>
<!ELEMENT prixTotal (#PCDATA)>
<!ELEMENT date (#PCDATA)>
<!ELEMENT detailCommandes (article*)>
<!ELEMENT article (nom, prix, qte, vendeur)>
<!ELEMENT nom (#PCDATA)>
<!ELEMENT prix (#PCDATA)>
<!ELEMENT qte (#PCDATA)>
<!ELEMENT vendeur (#PCDATA)>
```

Il y a deux types de factures :

1) Les factures clients :

Lors du paiement fait par le client, un code ajax est exécuté (pages\panier.php):

```
$.ajax({
  type: 'GET',
  url: '../xml/factureClient.php',
  data: 'idCom='+ encodeURIComponent($('#numCom').text()),
  success: function(data){
    document.location = "panier.php?payer="+encodeURIComponent(true);
  }
});
```

Il permet de déclencher un script PHP (xml/factureClient.php) qui utilise l'objet DomDocument. Cela permet de créer un fichier en XML qui se nomme : Facture ElectrouveTout « numéro de la commande payé ».xml et qui se stocke dans le dossier : xml/factureClients/ . Toutes les factures des clients se stockent dans ce dossier.

2) La facture pour l'administrateur

Cela repose sur le même principe que les factures clients. A chaque paiement d'un client un code ajax est exécuté (pages\panier.php) :

```
$.ajax({
  type: 'GET',
  url: '../xml/facture.php',
  data: 'idCom='+ encodeURIComponent($('#numCom').text()),
  success: function(data){
    document.location = "panier.php?payer="+encodeURIComponent(true);
  }
});
```

Ce code va déclencher un script PHP (xml/facture.php) qui va vérifier si le fichier : Factures Commandes Admin.xml existe dans le répertoire xml\ . Si le fichier existe il va l'actualiser en ajoutant une nouvelle commande ou si il n'existe pas créer un nouveau fichier : Factures Commandes Admin.xml dans le dossier xml/ .

O- Les commentaires

Si un utilisateur est connecté sur ElectrouveTout il peut poster un commentaire sous un article.

Tous les commentaires s'affichent sous l'article auquel ils appartiennent grâce à de l'Ajax (Emplacement du code : vues/pages/vue-article/script-comment.php. Ligne 12) qui exécute un fichier PHP (Emplacement du code : controller/affichageCommentaire.php) qui retourne les commentaires de tous les utilisateurs.

P - Affichage de tous les articles

La page articles.php qui se trouve dans le dossier pages/ permet d'afficher tous les articles de la boutique ElectrouveTout. Cela est possible grâce à une requête SQL qui regroupe tous les articles. La requête tient compte des différents tris possibles pour l'affichage des articles en fonction des choix du client en termes de date d'ajout des articles, de leurs prix et de leur type.

Trier par nouveauté :

- Les plus récents
- Les plus anciens

Trier par prix :

- Prix croissant
 - Prix décroissant
- Choix du Type

caméra ▼

Valider

La requête utilisée : (Emplacement du code : controller/requestSQL.php Ligne 36).

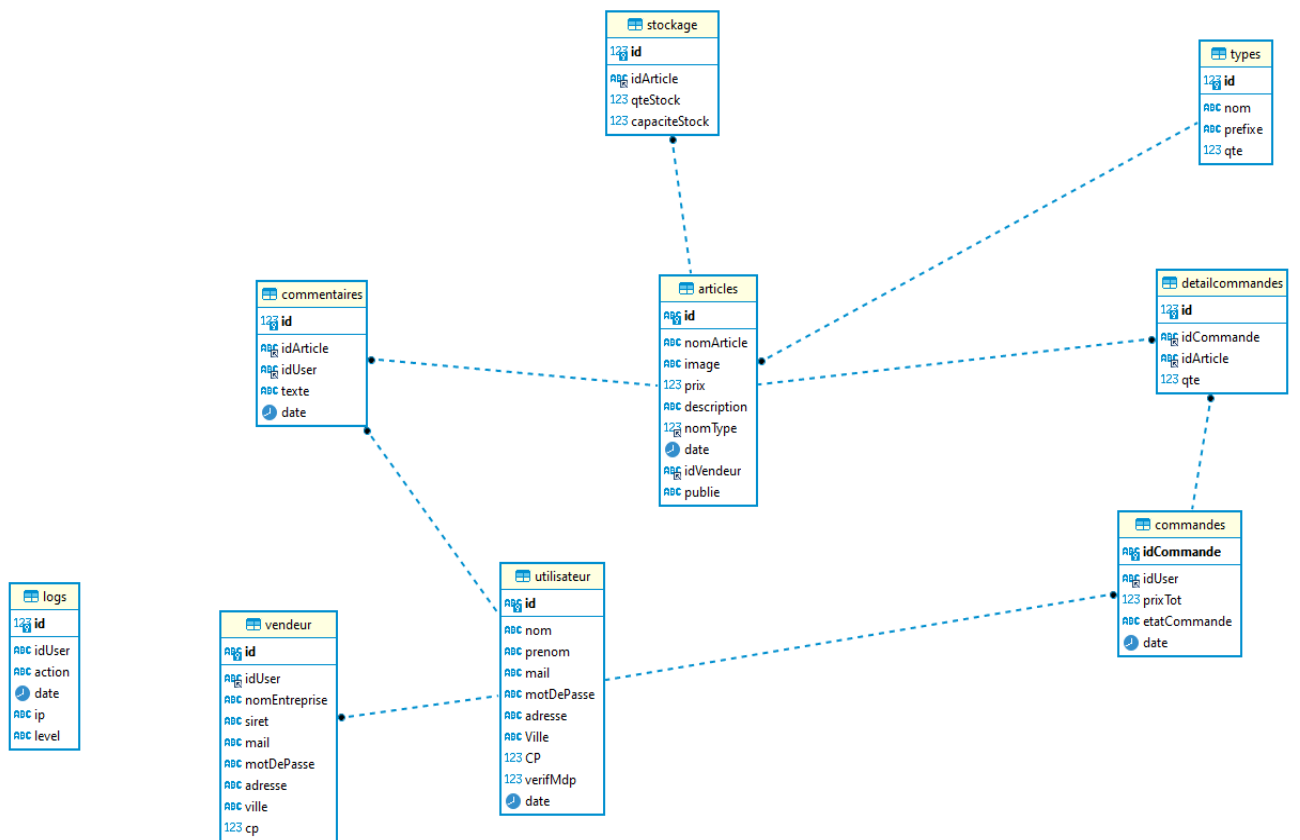
```
$tabArticles = "SELECT articles.*,stockage.qtestock from articles,stockage WHERE articles.id=stockage.idArticle and articles.publie = ?";
if (isset($type)){
    $tabArticles = "SELECT articles.id, articles.nomArticle, articles.image, articles.prix, articles.description, articles.nomType,
    articles.date FROM articles,types WHERE articles.nomType=types.id AND types.nom = ? AND articles.publie = ?" ;
}
if ($utilise=="") {
    $tabArticles= $connexion->prepare($tabArticles);
} else {
    $tabArticles= $connexion->prepare($tabArticles.$possibilite[$utilise]);
}

if (isset($type)) {
    $tabArticles -> execute(array($type, "1"));
} else {
    $tabArticles -> execute(array("1"));
}
$Articles = $tabArticles->fetchAll(PDO::FETCH_ASSOC);
```



3 - Annexes

Annexe 1



Annexe 2

```
<div class="input-field">
  <input type="text" minlength="1" maxlength="25" name="nom" placeholder="Nom" required>
  <i class="uil uil-user"></i>
</div>
<div class="input-field">
  <input type="text" minlength="1" maxlength="20" name="prenom" placeholder="Prénom" required>
  <i class="uil uil-user"></i>
</div>
<div class="input-field">
  <input type="email" minlength="1" maxlength="30" name="email" placeholder="Email" pattern="[a-z0-9._%+-]+@[a-z0-9.-]+\.[a-z]{2,}$" required>
  <i class="uil uil-envelope icon"></i>
</div>
```

Annexe 3

```
session_start();
$_SESSION["email"] = $email;
$_SESSION["mdp"] = $mdp;
$_SESSION["online"] = true;
$_SESSION["idUser"] = $tab["id"];
```

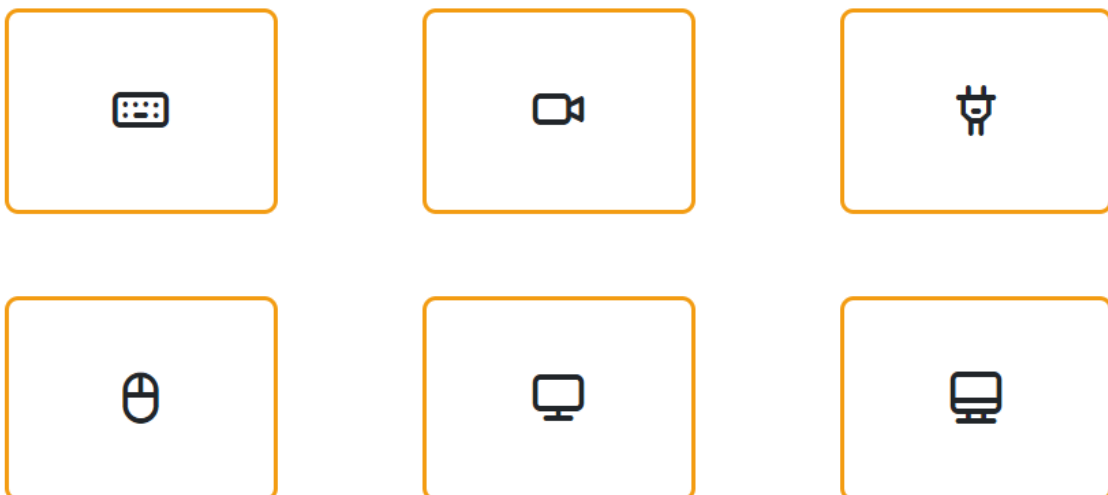
```
if ($email == "elec-admin@admin.fr") {
  $_SESSION["admin"] = true;
}
```

Annexe 4

```
1 <?php
2
3 // REQUETE POUR PAGE D'ACCUEIL
4 $selLogs = $connexion->prepare("SELECT * FROM logs ORDER BY date DESC");
5 $selLogs->execute();
6 $stabLogs = $selLogs->fetchAll(PDO::FETCH_ASSOC);
7
8 $selArticle = $connexion->prepare("SELECT * FROM articles ORDER BY date DESC");
9 $selArticle->execute();
10 $stabArti = $selArticle->fetchAll(PDO::FETCH_ASSOC);
11
12 $selType = $connexion->prepare("SELECT types.nom FROM articles, types WHERE types.id = articles.nomType");
13 $selType->execute();
14 $stabType = $selType->fetchAll(PDO::FETCH_ASSOC);
15
16 $minAccueil = 0;
17 $maxAccueil = 5;
18 $ecartAccueil = 0;
19
20 // REQUETE POUR PAGE DES LOGS
21 $selLogs = $connexion->prepare("SELECT * FROM logs ORDER BY date DESC");
22 $selLogs->execute();
23 $stabLogs = $selLogs->fetchAll(PDO::FETCH_ASSOC);
24
25 $countLogs = $connexion->prepare("SELECT COUNT(*) FROM logs");
26 $countLogs->execute();
27 $nbLogs = $countLogs->fetchColumn();
28
29 $minLogs = 0;
30 $maxLogs = 20;
31 $ecartLogs = 0;
32
33 if (isset($_GET["logs-affiches"])) {
34     $ecartLogs = $_GET["logs-affiches"];
35 }
```

Annexe 5

Nos types d'articles.




Annexe 6

Nos derniers articles.



PS5 Version Disque 599.95€

[Voir l'article](#)



PC Zenifier 559.98€

[Voir l'article](#)



Intel Core i5-12400F (2.5 GHz / 4.4 GHz) 218.95€

[Voir l'article](#)

[Tous les articles](#)

Annexe 7

Dashboard Electrouvetout

Informations générales.

| Email | Action | Date | IP | Niveau |
|--|--|---------------------|-----------|--------|
| elec-admin@admin.fr | L'administrateur a ajouté un article (SR-45189). | 2023-04-17 10:59:04 | 127.0.0.1 | low |
| elec-admin@admin.fr | L'utilisateur s'est authentifié avec succès. | 2023-04-17 10:05:28 | 127.0.0.1 | low |
| elec-admin@admin.fr | L'utilisateur s'est authentifié avec succès. | 2023-04-14 14:39:35 | 127.0.0.1 | low |
| elec-admin@admin.fr | L'utilisateur s'est authentifié avec succès. | 2023-04-11 10:45:05 | 127.0.0.1 | low |
| elec-admin@admin.fr | L'administrateur a publié un article (PR-19412). | 2023-03-31 15:31:31 | 127.0.0.1 | low |

Les 5 derniers logs enregistrés.

| Statut de Electrouvetout | Identifiant | Nom | Prix | Type | Date |
|--------------------------|--------------------------|--|---------|---------|------------|
| Aucun problème détecté. | SR-45189 | souris | 58.00€ | Clavier | 2023-04-17 |
| | CO-81719 | PS5 Version Disque | 599.95€ | Clavier | 2023-02-22 |
| | CP-21848 | PC Zenifier | 559.98€ | Souris | 2023-02-22 |
| | PR-33145 | Intel Core i5-12400F (2.5 GHz / 4.4 GHz) | 218.95€ | Souris | 2023-02-16 |
| | PL-18549 | Multiprise 5 prises avec interrupteur | 12.95€ | Souris | 2023-02-09 |

Les 5 derniers articles ajoutés.

Annexe 8

Identifiant

CP-21848

Nom de l'article

PC Zenifier

Prix (€)

559,98

Quantité en stock

25

Capacité en stock

50

Description

Professionnel ou particulier, profitez de la puissance d'un PC polyvalent et évolutif conçu pour durer. Le PC Zenifier-SSD embarque un processeur AMD Ryzen 5 Pro 4650G et le SSD de 480 Go apporte plus de rapidité au démarrage de

Modifier l'article

Annuler

Annexe 9

Retour

| Numéro de commande | Image | Identifiant article | Prix unitaire | Quantité |
|--------------------|-------|---------------------|---------------|----------|
| C-32003252-39394 | | CL-66927 | 12.95€ | 3 |
| C-32003252-39394 | | CM-20145 | 149.95€ | 1 |

Affichage de 0 à 20 détails de la commande