

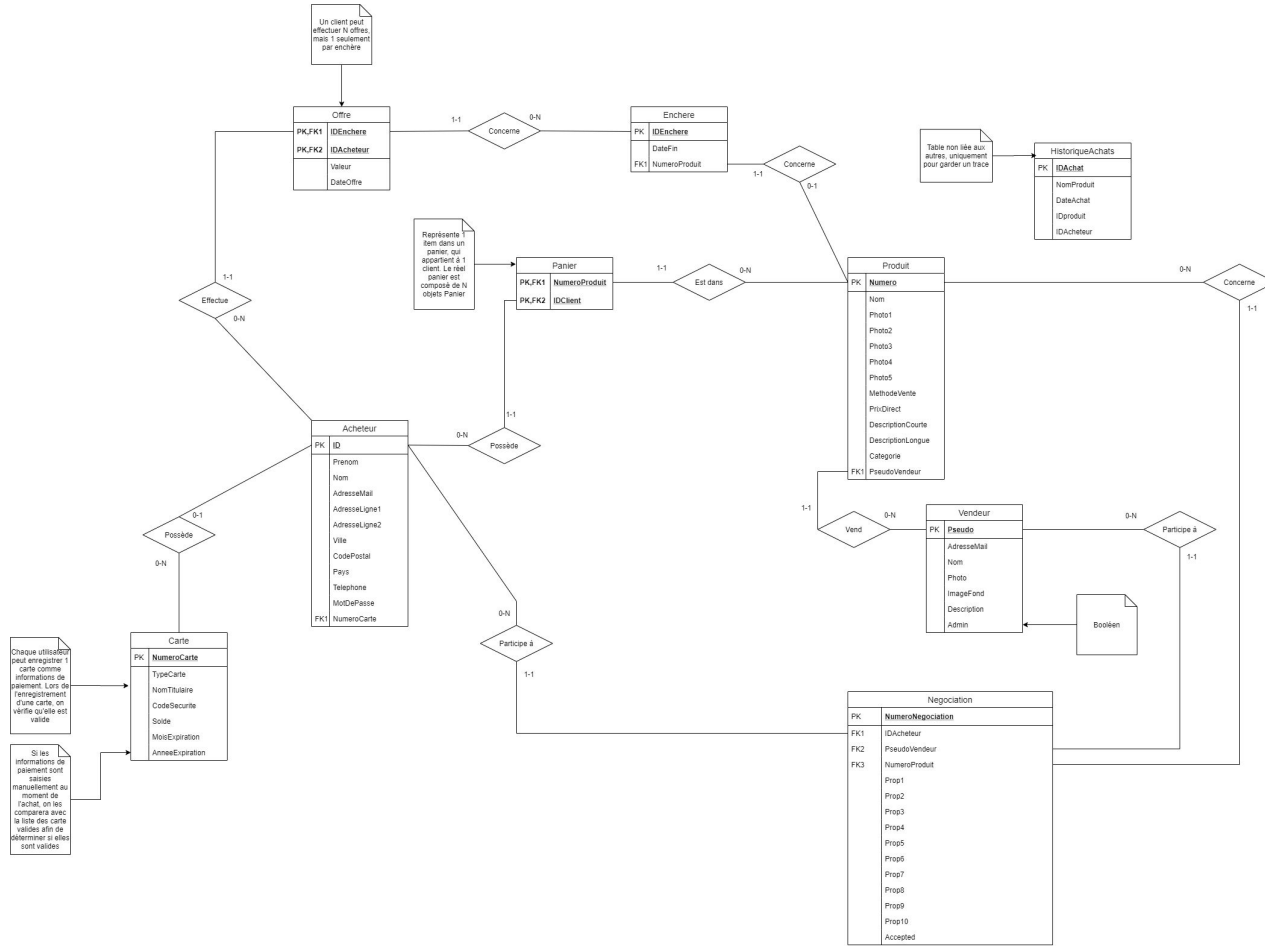
Projet Piscine

Aurèle Duparc – Sylvain Louche – Simon Jolly

Sommaire

- Conception du back
- Design du front
- Spécifications fonctionnelles des pages web
- Versioning GIT
- Bilans individuels
- Bilan collectif
- Bibliographie

Conception du back: Entiter relation

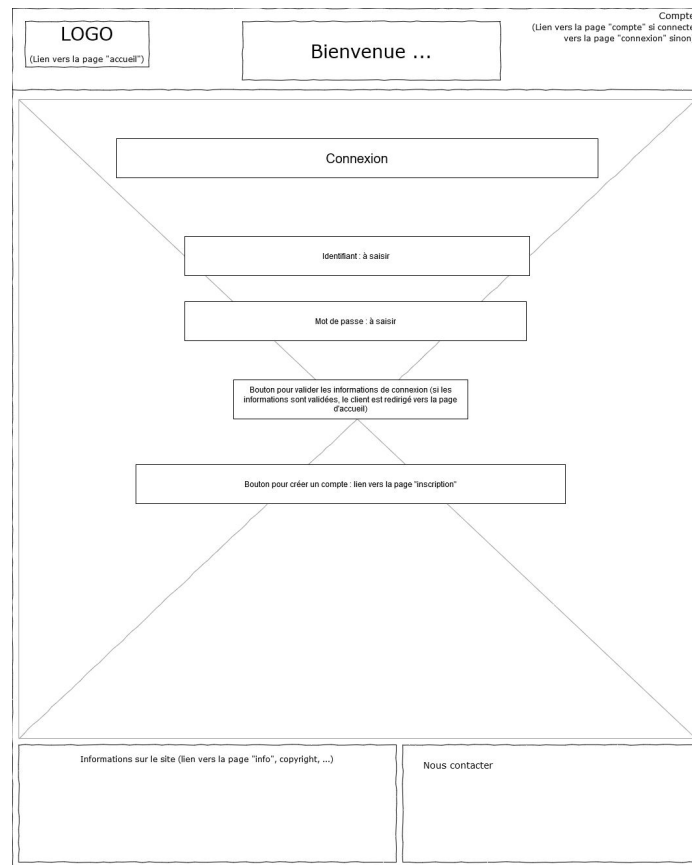
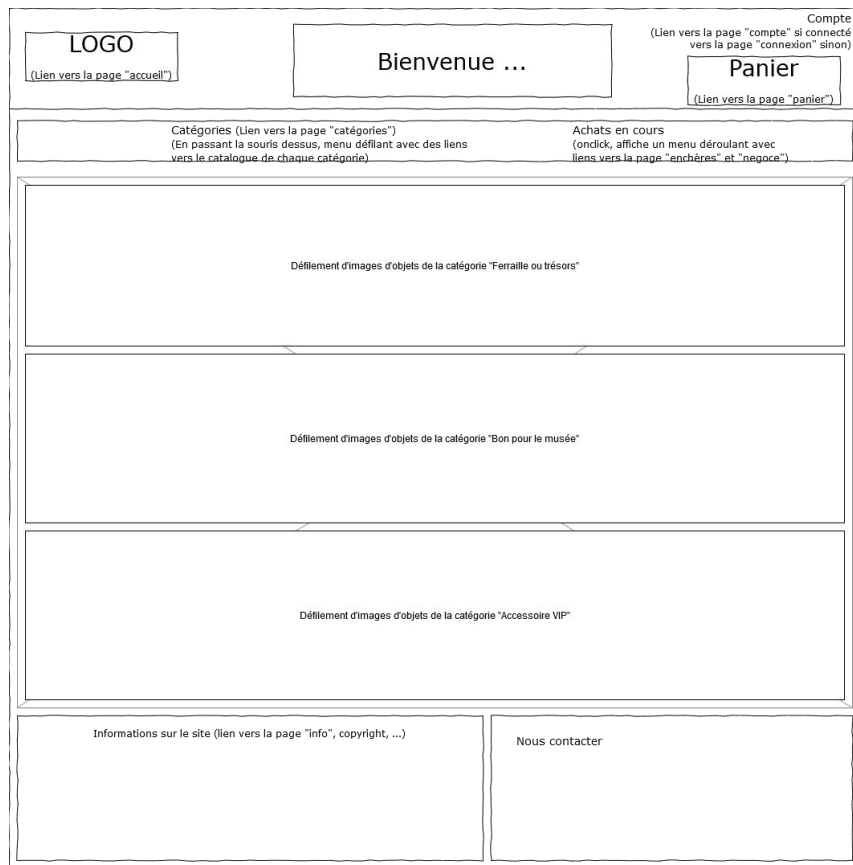


Conception du back : modèle relationnel

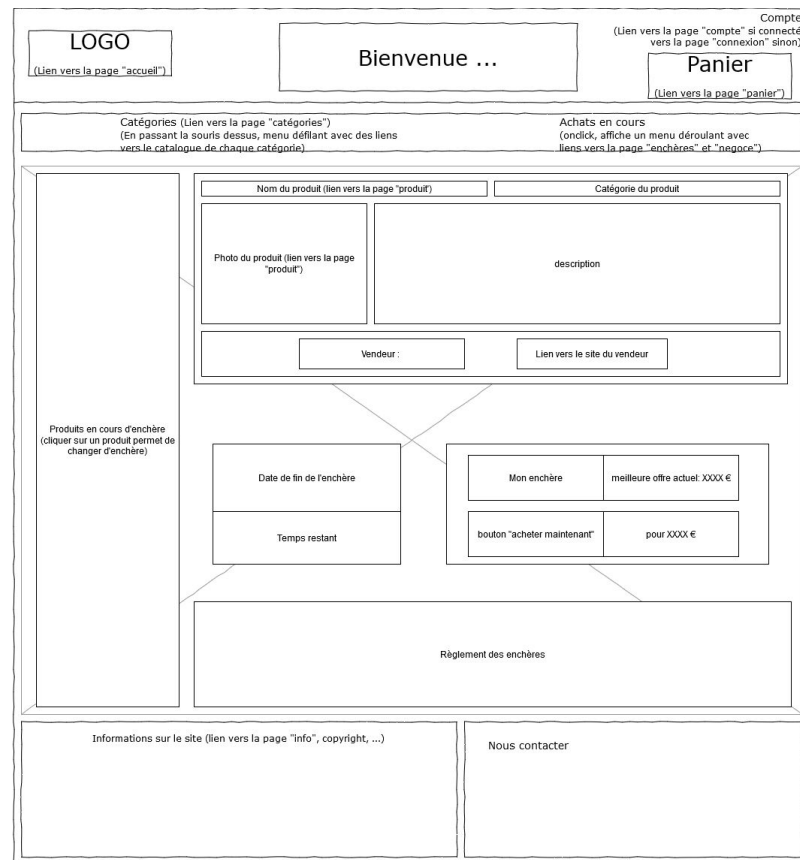
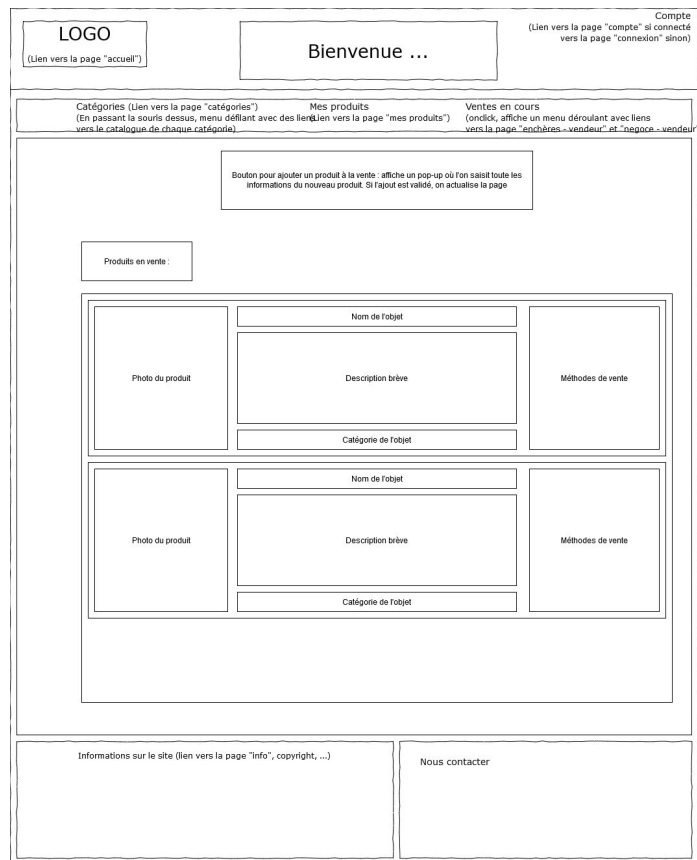
Les tables/relations :

- acheteur(ID, Prenom,...)
- Vendeur(Pseudo, AdresseMail,...)
- Produit(Numero, #ID, #Pseudo, Nom...)
- Enchère(IDEnchere, #ID, #Numero, DateFin)
- Négociation(NumeroNegociation, #IDAcheteur, #ID, #Numero, Prop1, ...)
- Carte(NumeroCarte, #ID, TypeCarte...)
- HistoriqueAchats(IDAchat, #ID, #Numero, DateAchat)

Design du front



Design du front



Design du front

LOGO
(Lien vers la page "accueil")

Bienvenue ...

Compte
(Lien vers la page "compte" si connecté
vers la page "connexion" sinon)
Admin
(Lien vers la page "liste vendeur")

Catégories (Lien vers la page "catégories")
(En passant la souris dessus, menu défilant avec des liens
vers le catalogue de chaque catégorie)

Mes produits
(Lien vers la page "mes produits")

Ventes en cours
(ondick, affiche un menu déroulant avec liens
vers la page "enchères - vendeur" et "negoce - vendeur")

Barre de recherche pour filtrer les vendeur par pseudos

Bouton pour ajouter un vendeur à la base de données : affiche un pop-up où on rentre les infos du nouveau vendeur

Infos sur le vendeur

Bouton pour supprimer le vendeur : affiche un pop-up pour demander vérification

Informations sur le site (lien vers la page "info", copyright, ...)

Nous contacter

LOGO

Bienvenue ...

Compte
Panier

Catégories

Achats

Finalisation du paiement

Informations de livraison (adresse, nom prénom, ...) : remplies depuis la base de données

Prix des produits

Coût de la livraison

Total

Date estimée de livraison

Bouton pour proposer au client de remplir les informations de paiement avec celles qu'il a renseignées sur son profil

Informations de paiement (numéro de carte, date de fin de validité, ...) : à saisir

Bouton payer : vérifie les informations et informe l'utilisateur si le paiement a été effectué. Le ramène à l'écran d'accueil

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.
Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequi nesciunt.

Nous contacter

Spécifications fonctionnelles des pages web

Notre site a pour but de permettre la rencontre entre des vendeurs et des acheteurs. Par conséquent, les utilisateurs doivent pouvoir de se connecter en tant qu'administrateur, vendeur et client. de plus le site doit s'adapter à l'identité de chacun. On utilise pour cela les variables de session du php

```
$typeConnecté=(isset($_SESSION['typeConnecté']))?(int) $_SESSION['typeConnecté']:1;
//visiteur : 1
//client : 2
//vendeur : 3
$idConnecté=(isset($_SESSION['idConnecté']))?(int) $_SESSION['idConnecté']:0;
//id si client connecté
$pseudoConnecté=(isset($_SESSION['pseudoConnecté']))?$_SESSION['pseudoConnecté']:'';
//pseudo si vendeur connecté
```

Ainsi, les en têtes des pages ainsi que les liens s'adaptent en fonction de l'utilisateur connecté.

```
<?php
echo '<a href="'.(($typeConnecté == 3)?"profil_vendeur_privé.php":(($typeConnecté == 2)?"profil_client.php":"page_de_connexion.php")).'>';
?>
```

Ce sont les administrateurs qui ajoutent les vendeurs et leur donnent la possibilité de mettre des objets en vente. Pour cela les administrateurs remplissent un formulaire, codé en html, puis appuient sur le bouton de type 'submit'. Ce bouton valide la condition du code php ci-dessous. Ce code récupère les informations du formulaire html et après quelques étapes de vérification des données, les envoie dans la base de données à l'aide du mot clé 'INSERT'.

```
if (isset($_POST["boutonAjout"])) {
    if($_POST["boutonAjout"] && $db_found) {
        $ajoutNom = isset($_POST["ajoutNom"]) ? $_POST["ajoutNom"] : "";
        $ajoutPseudo = isset($_POST["ajoutPseudo"]) ? $_POST["ajoutPseudo"] : "";
        $ajoutMail = isset($_POST["ajoutMail"]) ? $_POST["ajoutMail"] : "";
        $ajoutCheckAdmin = isset($_POST["ajoutCheckAdmin"]) ? $_POST["ajoutCheckAdmin"] : "";
        if ($ajoutNom==" " || $ajoutPseudo==" " || $ajoutMail==" ") {
            $erreurAjout = "Tous les champs doivent être remplis";
        } else {
            $sqlTestAjout = "SELECT * FROM vendeur where Pseudo like '%" . $ajoutPseudo . "%'";
            $resultTestAjout = mysqli_query($db_handle, $sqlTestAjout) or die (mysqli_error($db_handle));
            if (mysqli_num_rows($resultTestAjout) != 0) {
                $erreurAjout = "Le vendeur existe déjà";
            } else {
                $sqlAjout = "INSERT into vendeur (pseudo,AdresseMail,Nom,Admin) values ('" . $ajoutPseudo . "','" . $ajoutMail . "','" . $ajoutNom . "','" . $ajoutCheckAdmin . "')";
                $resultatAjout = mysqli_query($db_handle, $sqlAjout);
                $erreurAjout = "Ajouté";
            }
        }
    }
}
```


Spécifications fonctionnelles des pages web

Une fois qu'un nouveau produit est ajouté sur le site, celui-ci apparaît pour le client. La page des produits va en effet aller chercher les informations stockées dans le tableau des produits de la base de données en utilisant la fonction php 'mysqli_query'.

Le programme va ensuite regarder chaque ligne du tableau. Pour chaque ligne, le programme php va, par la méthode 'echo', envoyer les informations collectées ainsi que le code html définissant la mise en page de ces informations.

```
$sql = "select * from produit";
$result = mysqli_query($db_handle, $sql);
if (mysqli_num_rows($result) == 0) {
    echo "Aucun produit enregistré";
} else {
    while ($data= mysqli_fetch_assoc($result)) {
        echo '
        <div class="objet">
            <table>
                <tr>
                    <td>
                        
                    </td>
                    <td>
                        <h4>'.$data["Nom"].'</h4>
                        <div class="description_objet">
                            <p>'.$data["descriptionLongue"].'</p>
                            <p>'.$data["pseudoVendeur"].'</p>
                        </div>
                    </td>
                </tr>
                <tr>
                    <td>
                        <h5>Achetez : </h5>
                        <h4>'.$data["PrixDirect"].'</h4>
                        <p>'.$data["pseudoVendeur"].'</p>
                        <p>'.$data["pseudoVendeur"].'</p>
                        <a href="produit_client.php?numeroProduit='.$data["Numero"].'"> aller a la page du produit </a>
                    </td>
                </tr>
            </table>
        </div>
        ';
```

Lorsqu'un vendeur poste un nouveau produit, il peut choisir entre l'organisation d'une vente aux enchères ou celle d'une négociation.

Lorsqu'un client entame une négociation, il doit d'abord proposer une contre offre au prix initial du produit. Il peut proposer jusqu'à cinq contre offres. Lorsqu'il propose une contre offre, le vendeur peut soit accepter (le booléen "accepted" va alors prendre la valeur 1 qui implique l'achat du produit par le client et la fin de la négociation) ou alors proposer également une n-ième offre (lui aussi limité à 5 propositions). Le même schéma se reproduit ainsi de suite avec le client puis le vendeur. Les négociations ne peuvent s'effectuer qu'entre un unique client et un unique vendeur (vérifié par une clé étrangère reçu des tables 'acheteur' et 'vendeur').

```
<div class="col">
    <div class="row">
        <div class="col">
            <form action="negoc_client.php?numeroProduit=<?php echo $numProduit ?>" method="post">
                <div class="col">
                    <input type="text" class="form-control" id="nouveloffre" name="nouveloffre" placeholder="Saisir votre nouvelle offre">
                </div>
                <input type="submit" name="boutonoffre" id="boutonoffre" class="btn btn-default" value="Nouvelle offre">
            </form>
        </div>
    </div>
</div>
```

Mes offres :

Offre 1 : 500.00
Offre 2 : 1562.00
Offre 3 : 1562.00
Offre 4 : 150.00
Offre 5 :

Offres du vendeurs :

Offre 1 : 10000.00
Offre 2 : 20000.00
Offre 3 : 30000.00
Offre 4 :
Offre 5 :

Accepter l'offre

Saisir votre nouvelle offre

Spécifications fonctionnelles des pages web

Lorsqu'une enchère est terminée et que le client ayant fait la meilleure offre se rend sur la page d'enchère du produit, il est informé qu'il a gagné l'enchère.

Lorsqu'il clique sur le bouton "Paiement", il est redirigé vers la page de paiement, où on l'informe du prix à payer. Ce prix est calculé en récupérant les informations des 2 meilleurs enchérisseurs (comparés en fonction de la somme enchérie puis de la date de l'enchère).

```
//Requête pour trouver le participant ayant fait la meilleure offre
$sqlGagnantEnchere = "SELECT A.IDAcheteur as idGagnant,A.Valeur as montantGagnant,B.DateFin as dateFinEnchere from offre A join enchere B on A.IDEnchere = B
.IDEnchere where B.IDEnchere = $produitPaye ORDER by A.Valeur desc,A.DateOffre asc limit 1";
```

```
$sqlSecond = "SELECT A.Valeur as montantSecond from offre A join enchere B on A.IDEnchere = B.IDEnchere where B.IDEnchere = $produitPaye ORDER by A.Valeur desc,A.DateOffre asc limit 1 offset 1";
```

Le site calcule la somme à payer (selon les règles du sujet) et attend les informations de paiement de l'utilisateur.

Lorsque l'utilisateur entre ses informations de livraison et de paiement et valide, le site vérifie les coordonnées et le solde de la carte avant de supprimer les produits de la liste (et les autres tables s'accordent en cascade) et d'ajouter les infos dans l'historique des achats

Règlement des enchères

La vente aux enchères peut être proposée par le vendeur s'il le souhaite. Celui-ci définit une date de fin d'enchère, jusqu'à laquelle chaque utilisateur a la possibilité de faire une unique offre du montant de son choix. Une fois la date limite atteinte, l'utilisateur qui aura fait l'offre la plus élevée est déclaré gagnant de l'enchère. Dans le cas où plusieurs utilisateurs feraient la même offre, seul le premier d'entre eux à effectuer l'offre en question sera déclaré gagnant. La somme à payer pour le gagnant sera la somme minimale, inférieure ou égale à la valeur de son offre, requise pour surpasser toutes les autres offres.

☐ Je certifie avoir pris connaissance des conditions d'utilisations.

Vous avez gagné l'enchère ! Cliquez sur le bouton pour procéder au paiement.
Attention : le produit est toujours disponible à la vente directe !

Paiement

Finalisation du paiement

Informations de livraison	
Nom : <input type="text" value="Doe"/>	Prenom : <input type="text" value="John"/>
Adresse ligne 1 : <input type="text" value="Adresse1"/>	
Adresse ligne 2 : <input type="text"/>	
Ville : <input type="text" value="Paris"/>	Code postal : <input type="text" value="75015"/>
Pays : <input type="text" value="France"/>	Numéro de téléphone : <input type="text" value="+33611111111"/>

Prix des produits : 2000000.00 €

Coût de la livraison : 45.5 €

Total : 2000045.5 €

Informations de paiement	
<input type="button" value="Insérer les informations de paiement enregistrées."/>	
Type de carte : <input checked="" type="radio"/> Visa <input type="radio"/> Mastercard <input type="radio"/> American Express	
Numéro de la carte : <input type="text"/>	Titulaire : <input type="text"/>
Date d'expiration : <input type="text" value="MM"/> / <input type="text" value="AA"/>	Code de sécurité : <input type="text"/>

Payer

Versioning GIT

March 20, 2020 – April 20, 2020

Period: 1 month ▾

Overview

0 Active Pull Requests

0 Active Issues



0

Merged Pull Requests



0

Proposed Pull Requests



0

Closed Issues



0

New Issues

Excluding merges, **3 authors** have pushed **161 commits** to master and **161 commits** to all branches. On master, **0 files** have changed and there have been **0 additions** and **0 deletions**.



Apr 12, 2020 – Apr 20, 2020

Contributions: Commits ▾

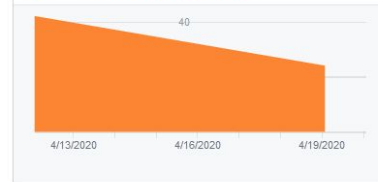
Contributions to master, excluding merge commits



SimonJ92

#1

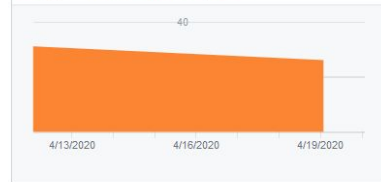
66 commits 17,692 ++ 7,319 --



Pere-Siffleur

#2

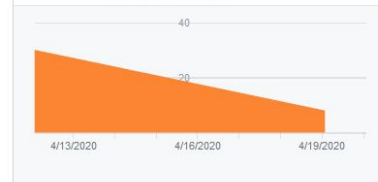
57 commits 6,941 ++ 2,368 --



Elerua

#3

38 commits 4,723 ++ 443 --



Bilan individuel

Aurèle Duparc: Cette semaine piscine a été pour moi bien plus complexe et plus intéressante que celle de notre ING2. Après avoir défini notre design générale, l'apprentissage plus avancé de l'HTML (CSS, JavaScript, Bootstrap) fut long mais sans trop de difficultés. Cependant à l'arrivée du développement back-end j'ai ressenti d'énormes difficultés et incompréhensions en PHP/MySQL. Sans l'aide de Simon et Sylvain il est **certain** que je n'aurais jamais pu rendre un projet aussi complexe que celui-ci. Ce projet piscine m'a permis de tester mes capacités et il est maintenant clair que certaines parties de programmation en web dynamique me sont encore particulièrement flous. Cette fin de projet piscine me permet de focaliser mes futurs efforts en direction de la programmation back-end.

Bilan individuel

Sylvain Louche: du point de vue technique, ce projet a été l'occasion de mettre en oeuvre et d'approfondir nos connaissances sur chacun des quatre langages web que nous avons étudié, html, css, javaScript et php. La communication entre tous ces langages est un des principaux points que ce projet m'a permis d'appréhender et de travailler.

Bilan individuel

Simon Jolly: Ce projet en format piscine de 3e année a été pour moi plus simple que le celui de l'année précédente, bien que présentant un grand nombre de problème pour lesquels il a fallu trouver une solution avant de commencer l'implémentation. Cependant, j'ai trouvé ce projet bien plus long que celui de l'année dernière, et nous avons peut-être été trop ambitieux lorsque nous avons choisi de ne pas se concentrer sur certaines fonctionnalités en premier lieu, mais de tout designer puis tout implémenter sans se soucier de la potentielle difficulté ou de la longueur de la tâche. Je suis tout de même satisfait à la fois du résultat final de notre travail ainsi que des compétences et des connaissances que j'ai pu acquérir tout au long de cette semaine, particulièrement en ce qui concerne la partie back-end, qui m'a nettement plus intéressé.

Bilan collectif

Les deux premiers jours de ce projet, nous avons eu de longues discussions durant lesquelles nous nous sommes mis d'accord sur les outils que nous allions utiliser. Pour la phase de conception du design et du model entier association, nous avons utilisé draw.io qui s'est révélé être un bon outil. De plus comme il nous était impossible de nous rencontrer, nous avons également réfléchi à la manière dont nous allions travailler dans ce contexte difficile. Par exemple, plutôt que d'attribuer à chacun un langage et de nous répartir entre front et back, nous nous sommes repartis le travail par pages. Nous avons chacun choisi les pages sur lesquelles nous allions travailler. Cela nous a permis non seulement de toucher à tous les aspects du cours mais également d'être moins dépendant du travail des autres. Cela a été possible uniquement parce que nous avons clairement défini le design de chaque page ainsi que la base de données à utiliser.

Au cours de ce projet, nous avons acquis, en plus des compétences techniques, des outils et de bonnes pratiques à réemployer lors de projets futurs.

Bibliographie

- Les cours web dynamique de l'ECE
- <https://openclassrooms.com>
- <https://www.commentcamarche.net>
- <https://www.w3schools.com>
- <https://sql.sh/cours>
- <https://www.php.net/manual/fr>