

2021/2022 – ISEN 3 – Projet de Fin d'Année
RAPPORT D'ACTIVITE
 (Département CSM)

Drive

Groupe 10

NOM Prénom étudiant 1	Loïc Blondeau	CSI3 <input checked="" type="checkbox"/>	CIR3 <input type="checkbox"/>	CNB3 <input type="checkbox"/>
NOM Prénom étudiant 2	Louis Boubert	CSI3 <input checked="" type="checkbox"/>	CIR3 <input type="checkbox"/>	CNB3 <input type="checkbox"/>
NOM Prénom étudiant 3	Martin Capelle	CSI3 <input checked="" type="checkbox"/>	CIR3 <input type="checkbox"/>	CNB3 <input type="checkbox"/>
NOM Prénom étudiant 4	Ilies Benslama	CSI3 <input checked="" type="checkbox"/>	CIR3 <input type="checkbox"/>	CNB3 <input type="checkbox"/>
Date de la soutenance	Mercredi 29 juin :			

Table des matières

1.0 - Introduction.....	3
2.0 - Etat de l'art et problématique.....	3
3.0 - Solutions envisagées	3
3.1) Présentation de la solution	3
3.1.1 : Exposition de la proposition	3
3.1.2 : Périmètre fonctionnel	3
3.1.3 : Choix techniques (matériels et logiciels).....	5
3.1.4 : Pilotage du projet	5
3.2) Analyse fonctionnelle.....	6
2.3.2.1 : Détail des fonctionnalités.....	6
2.3.2.2 : Maquette.....	7
3.3) Analyse technique	8
3.3.1 : Architecture technique (interfaçage entre les composants, flux...).....	8
3.3.2 : modélisation de la data	8
3.0 – Résultats obtenus	9
3.1) Difficultés rencontrées et solutions	9
3.2) Respect des délais	9
3.3) Respect du périmètre fonctionnel	9
4.0 – Conclusion	9
4.1) Montée en compétence de l'équipe	9
4.2) Axes d'améliorations	9
4.3) Et si c'était à refaire ?.....	10

1.0 - Introduction

En fin de troisième année, les étudiants ISEN participent à 6 semaines de projet choisi parmi plusieurs matières.

Nous avons choisi de travailler en informatique sur le projet « Drive LBR », qui est une demande concrète du client « Les briques rouges ».

2.0 - Etat de l'art et problématique

A l'occasion du festival « les briques rouges », événement organisé par des membres de Junia, nombre de photos et vidéos seront pris par des photographes et cadreurs à des fins de souvenirs et de teasing de communication.

Notre objectif ici est de créer un espace de stockage privé pour notre client capable de prendre en charge l'ensemble de ces photos et vidéos, toutes éditions confondues. Ce drive devra répondre aux demandes explicitées dans le cahier des charges.

3.0 - Solutions envisagées

3.1) Présentation de la solution

3.1.1 : Exposition de la proposition

Pour répondre à cette demande, nous nous sommes donc orientés vers la conception d'un site dans son intégralité

3.1.2 : Périmètre fonctionnel

Nous allons détailler ici dans l'ordre, toutes les fonctionnalités prises en charge sur notre drive.

Pour commencer, parlons du **compte utilisateur**.

Chaque compte est défini par : nom, prénom, adresse électronique, identifiant, mot de passe règlementé, une description du profil et un des quatre rôles permis sur le site, à savoir : invité, lecture, écriture ou admin.

La création d'un compte avec une information manquante et strictement impossible. Enfin, les possibilités d'interactions avec le site dépendent du rôle attribué.

Abordons les **fonctionnalités autorisées en fonction de chaque rôle**.

Les comptes type :

1. Lecture :

- Ce sont globalement de simple visiteur
- Accès à l'ensemble des photos et vidéos
- Effectuer des recherches par tag
- Possibilité de télécharger l'ensemble des fichiers

2. Ecriture :

- Même fonctionnalité que les comptes types lecture
- Ajouter, modifier, supprimer des vidéos
- Ajouter, modifier, supprimer des Tags sur les vidéos
- Créer, modifier, supprimer des tags sur la plateforme
- Créer, modifier, supprimer une catégorie de tag

3. Invité :

- Ajouter, modifier, supprimer des vidéos qui lui appartiennent
- Ajouter, modifier, supprimer des Tags sur les vidéos qui lui appartiennent
- Créer des tags sur la plateforme

Remarque sur ce point : Les invités ont uniquement l'accès à leurs propres photos et vidéos. Les administrateurs ne peuvent pas encore donner les droits d'accès à certains tags pour ce type de compte. En effet, lors de l'élaboration du diagramme de Gantt, nous n'avons pas catégorisé cette fonctionnalité parmi les prioritaires. Ce faisant, nous avons manqué d'un peu de temps pour ajouter cette fonction.

4. Admin :

- Rôle de lecture/écriture
- Créer/modifier /supprimer compte
- Voir l'espace de stockage total utilisé

5. Tous les rôles :

- Modifier son mot de passe
- Réinitialiser son mot de passe en cas d'oubli

Parlons maintenant du **système de classification des fichiers**.

Les fichiers sont triés selon un système de tag et de catégorie de tag. Nous avons la possibilité de créer des tags et des catégories. A noter : quand on crée un tag, on doit obligatoirement l'attribuer à une catégorie.

Lorsqu'on importe une photo ou une vidéo, cette dernière se verra automatiquement attribuer le tag : « sans tag ». Ce tag appartient à la catégorie « autre » et, ce tag comme cette catégorie ne seront pas supprimables.

Enfin, voyons **toutes les actions demandées** dans le cahier des charges.

Il est possible **d'ajouter un ou plusieurs fichiers** en même temps tant que l'extension correspond bien à des photos et vidéos. L'upload supporte les fichiers de plus de 10Gb.

L'utilisateur peut **supprimer des fichiers** un à un qui seront ensuite placés dans une corbeille. La suppression automatique des fichiers de la corbeille au bout de 30 jours n'est pas intégrée.

Tout ce qui touche aux **créations, suppressions, modifications des tags ou catégories de tag** est fonctionnel, demande de confirmation incluse.

Le **clic gauche** permet de visualiser une image sans avoir à la télécharger, quant au clic droit, il permet d'accéder aux informations d'un fichier, d'y ajouter ou supprimer des tags, de supprimer le fichier ou de télécharger le fichier.

Lorsqu'un compte est supprimé, ses informations sont conservées mais la connexion au compte n'est plus possible.

Enfin, nous avons intégré **un journal de bord** répondant à « qui a fait quoi et quand ? » accessible uniquement par les administrateurs.

Petit point sur la **sécurité du site** : Les mots de passes sont hachés en SHA256, il n'est pas possible de subir une injection SQL, toutes interactions utilisateurs/serveurs est traitée permettant de sécuriser à la fois l'utilisateur (mot de passe) ainsi que le site en lui-même (failles XSS) enfin, toutes les extensions de fichiers ne sont pas autorisées interdisant l'importation de fichiers malveillants (.php, .exe etc...)

3.1.3 : Choix techniques (matériels et logiciels)

Nous avons utilisé PhpMyAdmin, PHP, SQL, HTML5, CSS3, JavaScript et jQuery. Nous avons utilisé « atom » comme éditeur de texte, Wamp comme serveur virtuel (afin de simuler un serveur sur notre ordinateur) et enfin gitHub pour avoir un drive partagé du projet.

3.1.4 : Pilotage du projet

3.1.4.1 : Estimation des charges

Actions	Membre de l'équipe	Temps estimé
Maquette Figma	Loïc, Martin, Ilies	1,5 semaine
Cours PHP	Martin, Ilies	1 semaine
Configurer gitHub	Loïc	2h00
Etablir le MCD	Louis, Ilies	2 jours
Création de la base de données	Louis	1,5 jours
Importation de fichiers	Loïc, Louis	1 semaine
Créer, modifier, supprimer compte	Martin, Ilies	2 jours
Page de connexion et hachage mot de passe	Loïc, Louis	2 jours
Interface clic droit à l'aide du plugin contextmenu	Loïc	1 jour
Agrandir le media au clic gauche	Loïc	½ journée
Envoie de mail à la création de compte	Martin	1 jour
Créer, modifier, supprimer une catégorie de tag	Louis, Loïc	1,5 jours
Créer, modifier, supprimer un tag et le déplacer vers une catégorie	Louis, Loïc	1,5 jours
Journal de bord	Martin, Ilies	1,5 jour
Fonction « trier par »	Loïc	1,5 jour
Gestion des rôles	Martin, Ilies, Louis	1,5 jour
Barre de recherche	Martin	1,5 jour
Recherche par tag	Loïc	½ journée
Gestion de sécurité	Martin	2 jours
Stockage	Louis	½ journée
Site responsive pour mobile	Loïc	½ journée
Corbeille	Loïc	½ journée
Gestionnaire de compte	Martin, Ilies	2 jours
Rapport de projet	Ilies	1,5 jours

3.1.4.2 : Communication

Pour communiquer nous avons utilisé un groupe Snapchat, des réunions Discord et chaque matin nous commençons par un débriefing de la journée précédente à 9h00 suivi de la planification des tâches journalières.

3.2) Analyse fonctionnelle

2.3.2.1 : Détail des fonctionnalités

La fonctionnalité de **création d'un compte** est uniquement autorisée pour les administrateurs. Ces 2 lignes de codes permettent de faire appel à la fonction « `autorisation_admin` » du fichier « `functions` ». Nous utilisons cette fonction de manière récurrente, d'où l'intérêt de la créer une fois puis de l'appeler.

```
12
13 // Autorisation admin
14 include_once('functions.php');
15 autorisation_admin();
16 ?>
17

function autorisation_admin()
{
    if($_SESSION['loggedUser']['Role']!="Admin")
    {
        header("Location: accueil.php");
        exit();
    }
}
```

```
<form action="account_manager_submit_user.php" method="POST"> <!-- Methode get nul, transect info par url, sert à envoyer un formulaire -->
<div class="mb-3">
    <label for="Prenom" class="form-label">Prenom</label>
    <input type="text" class="form-control" id="Prenom" name="Prenom">
</div>
<div class="mb-3">
    <label for="Nom" class="form-label">Nom</label>
    <input type="text" class="form-control" id="Nom" name="Nom">
</div>
<div class="mb-3">
    <label for="Email" class="form-label">Email</label>
    <input type="email" class="form-control" id="Email" name="Email">
</div>
<div class="mb-3">
    <label for="MDP" class="form-label">Mot de passe</label>
    <input type="password" class="form-control" id="MDP" name="MDP">
</div>
<div class="mb-3">
    <label for="Description" class="form-label">Description</label>
    <textarea class="form-control" placeholder="Écrivez la description ici" id="Description" name="Description"></textarea>
</div>
</form>
```

Une fois l'autorisation donnée, on utilise la méthode « `POST` » pour créer un formulaire à remplir par l'utilisateur où il entrera toutes les données nécessaires à la création d'un compte. Nous aurions pu choisir la méthode « `GET` », mais les informations auraient été visibles via l'URL.

```
// Ecrute de la requête
$query = "UPDATE profil SET MDP = :MDP, email = :email, Nom = :Nom, Prenom = :Prenom, Description = :Description, Role = :Role";

// Préparation
$stmt = $pdo->prepare($query);

// Exécution : l'utilisateur est maintenant en base de données
$stmt->execute([
    'Id_Profil' => $id_Profil,
    'MDP' => $MDP_sha256,
    'email' => $email,
    'Nom' => $Nom,
    'Prenom' => $Prenom,
    'Description' => $Description,
    'Role' => $Role,
]);
```

Pour **modifier un compte** nous faisons une simple requête SQL type « `UPDATE` » qui écrase l'ancienne donnée

La **suppression de compte** est quant à elle une variante de la **modification de compte**, il suffit de remplacer le mot de passe actuel par « `NULL` »

En ce qui concerne la **suppression d'un fichier** : Cela se fait en deux étapes. Première étape, on place le fichier dans la corbeille via une requête SQL. Deuxième étape, grâce à la requête « `DELETE` », on supprime définitivement le fichier de la base de données.

```
<?php //delete
if(isset($_SESSION['random_OK'], $_POST['randomdeleteOK']) && $_POST['randomdeleteOK'] == $_SESSION['random_OK']){
    include("connect.php");
    $req = $pdo->prepare("UPDATE fichier SET Corbeille = 1 WHERE bin=?");
    $req->execute(array($_POST['delete']));
}
unset($_POST['randomdeleteOK']);
?>
```

```
<?php //delete
if(isset($_SESSION['random_OK'], $_POST['randomdeleteOK']) && $_POST['randomdeleteOK'] == $_SESSION['random_OK']){
    include("connect.php");
    $req = $pdo->prepare("DELETE FROM fichier WHERE bin=?");
    $req->execute(array($_POST['delete']));
    unlink($_POST['delete']);
}
unset($_POST['randomdeleteOK']);
?>
```

Parlons maintenant de l'importation de fichier :

```
<?php // Importer des photos
if(isset($_SESSION['random_ok'], $_POST['randomFormOK']) && $_POST['randomFormOK'] == $_SESSION['random_ok']) { // Protection contre l'actualiser la page"
    $total_count = count($_FILES['media']['name']);
    for($i=0; $i < $total_count; $i++) {
        if(isset($_FILES['media'][$i]['error'][$i]) == 0){
            if($_FILES['media'][$i]['size'][$i] <= 1000000000) { //Max 1000
                include("connect.php");

                $allowedExtensions = ['ogg', 'm4v', 'mpg', 'webm', 'ogv', 'mov', 'asx', 'mpeg', 'mp4', 'm4v', 'avi', 'jpg', 'jpeg', 'gif', 'png', 'tiff', 'pjp', 'jflif', 'bmp',
                $extension = strtolower(pathinfo($_FILES['media'][$i]['name'][$i])['extension']);
                if(in_array($extension, $allowedExtensions)){ //on vérifie que l'extension est un média
                    $req=$PDO->query("SHOW TABLE STATUS FROM lbr LIKE 'fichier'");
                    $res = $req->fetch();
                    $id_fichier = $res['Auto_increment'];
                    $req = $PDO->prepare("insert into caractériser(id_fichier,id_tag) values(?,0)");
                    $req->execute(array($id_fichier));
                    mkdir("upload/".$id_fichier, 0700);
                    $nomfichier = basename($_FILES['media'][$i]['name'][$i]);
                    $chemin = "upload/".$id_fichier."/". $nomfichier;
                    move_uploaded_file($_FILES['media'][$i]['tmp_name'][$i], $chemin);
                    $date = date('Y-m-d H:i:s',filemtime($chemin));
                    $req=$PDO->prepare("insert into fichier(type,titre,auteur_id,taille,date_de_publication,bin) values(?,?,?,?,?,?)");
                    $req->execute(array($_FILES['media'][$i]['type'][$i],$_FILES['media'][$i]['name'][$i],$_SESSION['loggeduser'][$i]['id_profil'],$_FILES['media'][$i]['size'][$i],$date,$chemin));
                }
                else{
                    echo "<script>alert('Erreur, mauvaise extension: ".$extension."');</script>";
                }
            }
        }
    }
}
else{
```

On utilise une boucle « for » qui importe les « i » fichiers détectés sur la ligne juste au dessus un par un. Dans cette boucle, on vérifie notamment la taille du fichier ainsi que son extension qui sont règlementées. Concrètement, dans la base de données se trouvera un lien vers le fichier qui sera stocker directement sur le serveur.

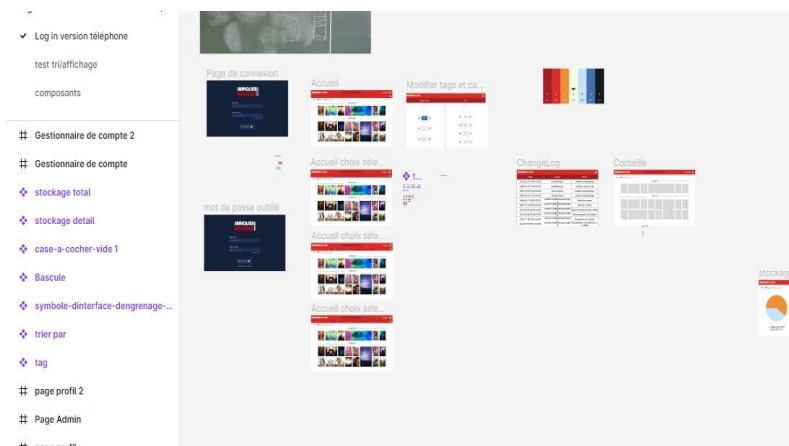
Dernière fonction remarquable : l'envoi de mail automatique

La fonction mail renvoie un booléen « True » si le mail est envoyé, « False » sinon.

Si le mail est envoyé, on informe les administrateurs via l'historique de modification. Dans le cas contraire, un message d'alerte est affiché.

```
account_Manager_header.php 102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
```

2.3.2.2 : Maquette

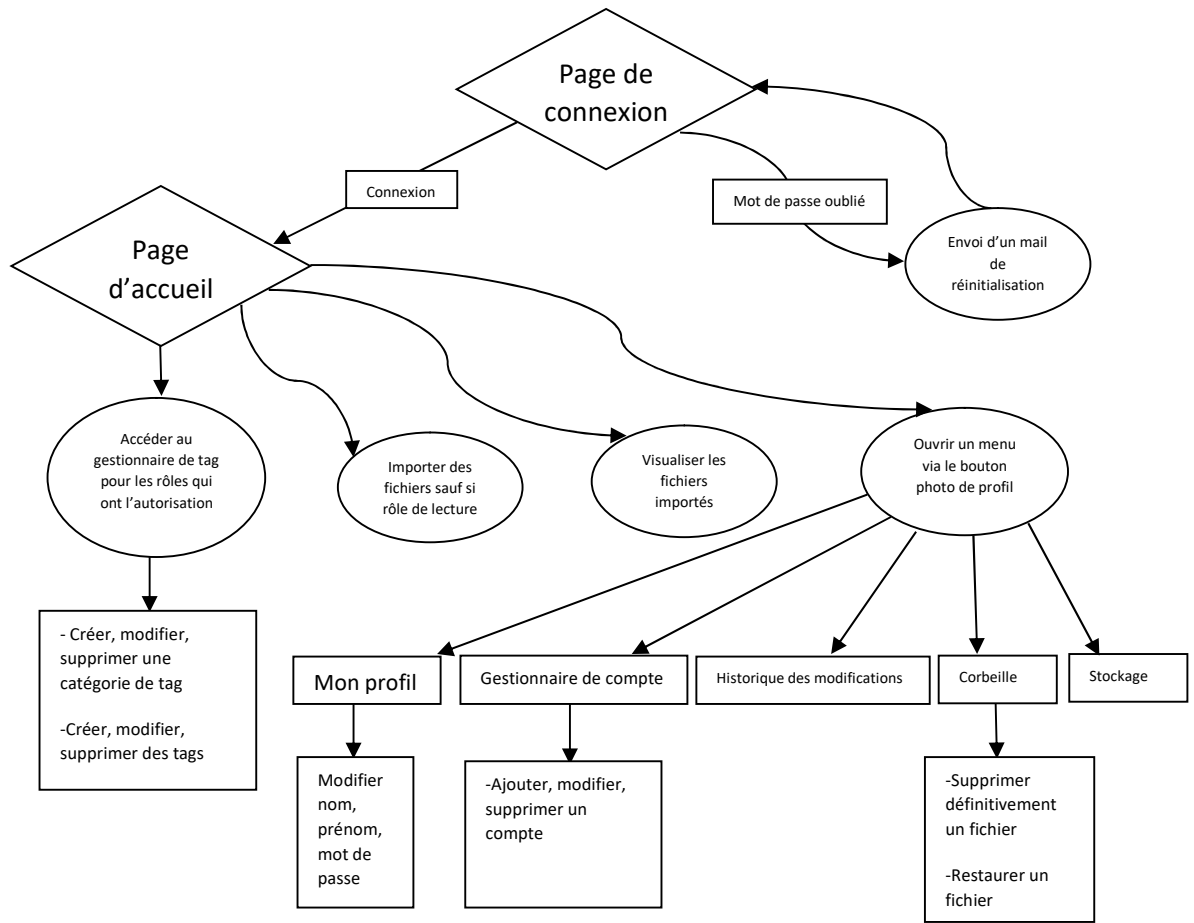


Lien vers la présentation figma :

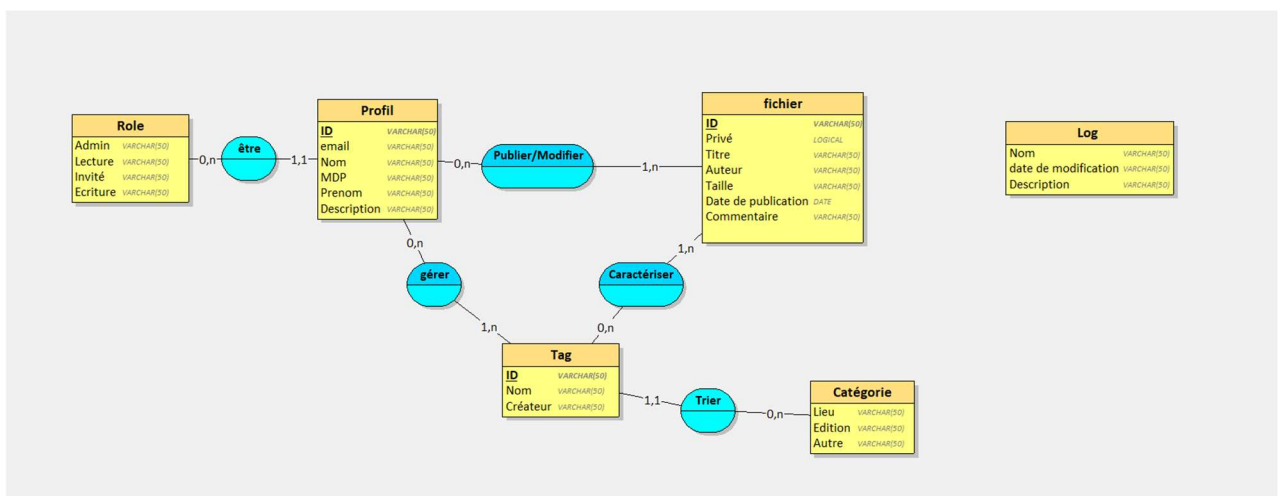
<https://www.figma.com/proto/FcGGPpbyLA7IAzF6kzArIx/Maquette-drive-LBR---BLONDEAU-CAPELLE-BENSLAMA-BOUBE-RT?node-id=156%3A56&scaling=contain&page-id=0%3A1&starting-point-node-id=156%3A>

3.3) Analyse technique

3.3.1 : Architecture technique (interfaçage entre les composants, flux...)



3.3.2 : modélisation de la data



3.0 – Résultats obtenus

3.1) Difficultés rencontrées et solutions

La toute première difficulté rencontrée était d'apprendre un nouveau langage, le PHP, depuis le début et un nouveau logiciel, Figma. Nous avons donc suivi un cours en ligne sur Open Classroom afin de nous familiariser à ce langage et des vidéos YouTube pour comprendre Figma.

Ensuite, nous souhaitions stocker tous les fichiers dans la base de données. Problème, les fichiers sont lourds et la ralentisse. De plus stocker des images étaient déjà difficile alors on ne pensait même pas au stockage de vidéo avec cette méthode.

On a donc décider de déplacer les fichiers sur le serveur et d'enregistrer le chemin vers le fichier sur la base de données à la place du blob.

Autre problème, nous avons essayé de réaliser les envoies de mail avant de recevoir le PDF du professeur. La méthode que nous souhaitions suivre était de donner les droits d'accès à Wamp sur Gmail. C'était sans compter le récent changement de Gmail qui empêche les applications tierces d'accéder à ce service. En revanche après lecture du PDF, il ne nous a pas fallu longtemps avant de réussir cette tâche.

3.2) Respect des délais

Le projet sera rendu fonctionnel en temps et en heure mais nous trouvons encore des axes d'améliorations. Globalement nous nous étions défini des délais sur Asana, une application de gestion de tâche, mais en raison de certains problèmes rencontrés pendant le projet, nous avons pris du retard impliquant le manque de certaines fonctions demandées dans le cahier des charges.

3.3) Respect du périmètre fonctionnel

A ce jour, les fonctions manquantes par rapport au cahier des charges sont : La suppression automatique des fichiers dans la corbeille au bout de 30 jours, la possibilité pour les admins d'autoriser la visualisation de certains tags pour les comptes invités, la suppression et le téléchargement de plusieurs fichiers à la fois ainsi que la suppression et l'attribution d'un même tag sur plusieurs fichiers de façon simultané.

4.0 – Conclusion

4.1) Montée en compétence de l'équipe

Individuellement, Loïc qui était le chef de projet a notamment appris à gérer une équipe, augmenter son nombre de compétence technique et approfondi ses connaissances de base. Louis a appris à créer une base de données de toute pièce, coder en PHP et faire des requête SQL. En ce qui concerne Martin, coder en PHP, faire des requête SQL, configurer un serveur local pour les mails et les envoyer sont des compétences acquises. Enfin, Ilies a appris à coder en PHP, approfondi ses bases de CSS, faire des requête SQL et manier l'application Figma.

Collectivement, nous nous sommes rendu compte qu'avancer dans un groupe avec une bonne cohésion permet de travailler plus vite et dans un environnement sain et agréable. Nous nous sommes aussi rendu compte que malgré tout, l'organisation peut vite dégénérer si on ne fait pas attention et qu'il est vraiment important de bien définir les tâches à accomplir en début de projet.

4.2) Axes d'améliorations

Nous pensons que nous aurions pu prendre plus de temps pour améliorer le design du site. Nous aurions dû également faire tester le site à plusieurs personnes « témoins » pour avoir des retours sur l'ergonomie, l'utilisation et la prise en main du site.

4.3) Et si c'était à refaire ?

Si nous avions à refaire un choix entre les 4 projets, nous referions le drive LBR car ce projet mêle gestion de compte, de média et d'affichage. C'est un projet complet, polyvalent et surtout concret ce qui nous a extrêmement motivé. De plus, travailler dans cette équipe était une vraie partie de plaisir, l'ambiance sérieuse mais surtout joyeuse était propice au développement d'un bon projet.