

Documentation technique

02:05:2022
Matos
Machado David
Francisco Garcia

1. Table des matières

2.	Tables des versions	3
3.	Introduction	3
4.	Rappel de l'énoncé.....	3
5.	Méthodologie.....	4
6.	Planification	6
6.0.1	Planning prévisionnel	7
6.0.2	Planning Réel	9
7.	Généralités concernant l'implémentation.....	12
7.3.1	Dictionnaire de données.....	17
8.	Plan et rapport de test.....	27
9.	Conclusion	30
9.1	Difficultés rencontrées	30
9.2	Améliorations possibles	30
9.3	Bilan personnel.....	31
9.4	Remerciements	31
10.	Annexe	31

2.Tables des versions

N° de Version	Date	Auteur	Changements apportés
1.0	18.05.2022	Machado David	Version finale du document pour le rendu TPI

3.Introduction

Voici ma documentation technique du projet *Matos*. Ce projet a été réalisé lors du Travail pratique individuel (TPI) en mai 2022. Il a été réalisé afin de pouvoir valider mes compétences en informatique au bout de quatre ans au Centre de Formation Professionnelle Technique du Petit-Lancy à Genève.

Matos est un site internet qui permet aux enseignants d'emprunter du matériel de l'école. En tant que visiteur il pourra seulement voir le matériel disponible. Connecter enseignant sur le site il pourra sélectionner le matériel pour l'emprunter et sera mis en attente que l'admin accepte. L'admin pourra ajouter ou modifier du matériel et accepter les prêts des enseignants mis en attente.

4.Rappel de l'énoncé

4.1 Organisation

Élève	Responsable TPI
David Machado	Francisco Garcia
David.mchdb@eduge.ch	Francisco.garcia@edu.ge.ch
Experts	
Yvan Poulin	Francesco Foti
Yvpoulin@gmail.com	Francesco.foti@devinfo.net

4.2 Livrable

Pour les experts et le formateur par email :

- Planning détaillé du projet
- Rapport de projet contenant le code source au format PDF
- Journal de bord
- Résumé du TPI(1 page en A4)

Pour le formateur uniquement :

- L'accès au repository distant du projet avec les droits de « clone »

- Un readme explicitant l'installation du projet en local
- Un dump de la base de données contenant la structure ainsi qu'un set de données de test

4.3 Matériel et logiciels à disposition

- Un PC standard école, 2 écrans
- Serveur SGBD à choix (EasyPHP, Wamp, Laragon, Adminr, phpMyAdmin, autre)
- Windows 10
- Visual Studio à jour
- Logiciel de création de schémas (Visio, Gliffy, autre)
- Outil de versionnage de code (Git, avec dépôt distant sur GitHub / Bitbucket / GitLab)
- Navigateur web (Mozilla Firefox/Google Chrome/Safari)
- Logiciel de création de maquettes d'interfaces utilisateur (Sketch, Adobe XD, Proto.io, autre)
- Outil bureautique à choix pour les documents (Google Docs, MSOffice, OpenOffice)
- L'étudiant est autorisé à utiliser son matériel personnel au besoin

4.4 Description de l'application

5. Méthodologie

Afin d'être organisé tout au long de mon travail, j'ai décidé d'utiliser la méthode en 6 étapes que nous avons apprises lors de nos 4 années de formation.

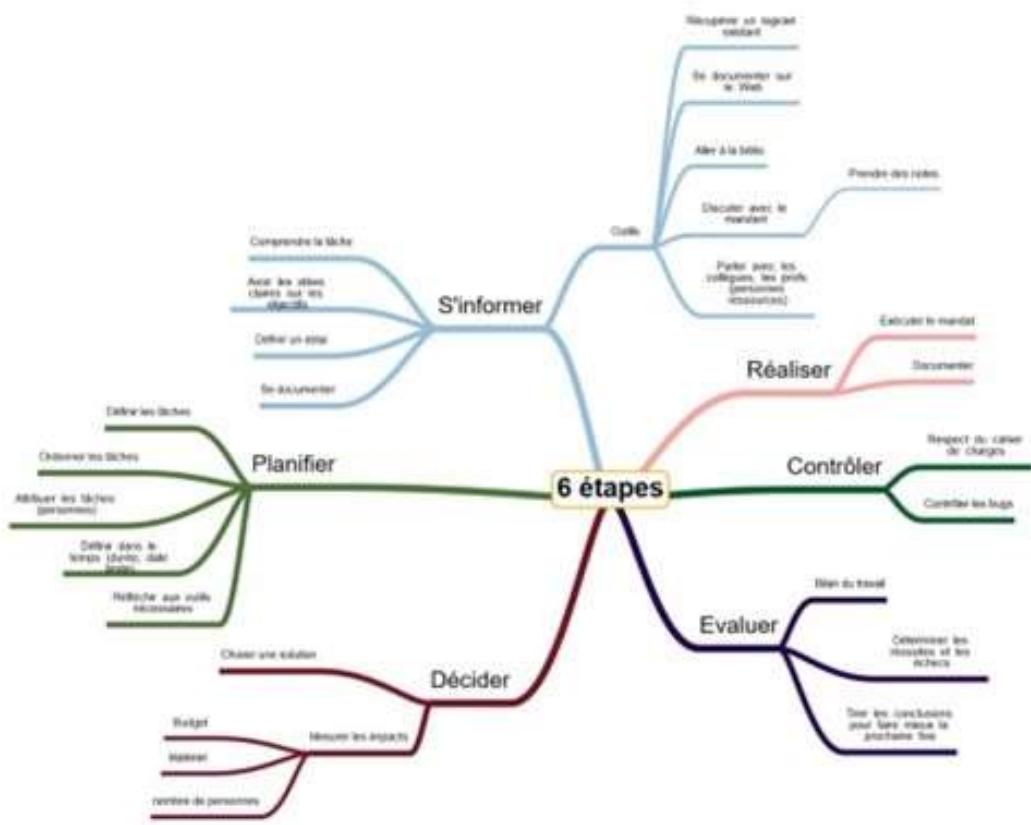


Figure 1 Méthode en 6 étapes

1. S'informer

Pour cette première étape de la méthode en 6 étapes, j'ai commencé par m'informer, c'est-à-dire que j'ai dû lire et analyser mon énoncé afin de le décortiquer pour comprendre chaque tâche qui m'était demandée. Pour se faire, j'ai décidé d'imprimer l'énoncé pour pouvoir avoir un visu constamment et rapide des points.

2. Planifier

J'ai dû établir un planning de travail afin de me faire une idée du temps que les tâches allaient me prendre. J'ai pris les éléments de l'énoncé. **A compléter**

J'ai décidé de mettre un ordre de priorité sur chaque tâche afin d'avoir un ordre de réalisation de tâches, voici le code que j'ai instauré pour les priorités :

- P0 : Bloquant
- P1 : Critique
- P2 : Important
- P3 : Secondaire

Une fois mes user stories créées, je les ai intégrées à un product backlog, qui me permet d'avoir un détail de chaque tâche du projet.

J'ai ensuite mis en place un diagramme de Gantt, qui me permet par la suite de voir les différences entre ce que j'avais prévu et ce que réellement réalisé.

3. Décider

Tout au long de la réalisation de mon travail, j'ai dû prendre des décisions sur ma façon de faire. J'ai parfois dû arrêter de développer pendant un moment afin de choisir entre deux solutions, laquelle pouvait être la meilleure à employer.

4. Réaliser

La quatrième étape est la réalisation. Elle consiste à exécuter les tâches que nous avons planifiées et décider au préalable. C'est en général l'étape la plus longue du projet, il faut se fier un maximum sur le plan établi.

Lors de la réalisation, j'ai tenu un journal de bord qui m'a servi à expliquer le fond de ma pensée de façon quotidienne. C'est aussi durant la réalisation que j'ai effectué le plus gros du travail. Pour ma part, la réalisation s'est bien passée, entre autres, car j'ai bien planifié le déroulement du projet.

5. Contrôler

Les contrôles se font lors de la cinquième étape. À ce moment-là, chaque User Story doit être complet, pour qu'elle soit ensuite vérifiée. Par exemple, nous pouvons vérifier si la description et le résultat de la tâche correspondent bien aux consignes établies. Il s'agit aussi d'effectuer un plan de test sur le projet en lui-même. Il est conseillé de faire effectuer les contrôles et les tests à des personnes tierces, car cela apporte de meilleurs résultats.

Pour ma part, j'ai choisi de faire des tests au milieu du projet. Parce que les fonctionnalités n'étaient pas compliquées au début et je n'ai pas eu d'erreur bloquant . J'avais aussi pensé à le faire uniquement à la fin de la conception de mon projet, mais je n'ai pas opté pour cette façon de faire, parce que j'allais oublier la plupart des tests.

6. Évaluer

La sixième et dernière étape est l'évaluation du travail. Lors de l'évaluation, nous passons en revue le déroulement du projet et nous repérons les aspects qui ont bien fonctionné et ceux qui ont moins bien fonctionné. De plus, pour ne pas avoir à refaire les mêmes erreurs, nous cherchons à connaître les points qui pourraient être améliorés pour le prochain projet.

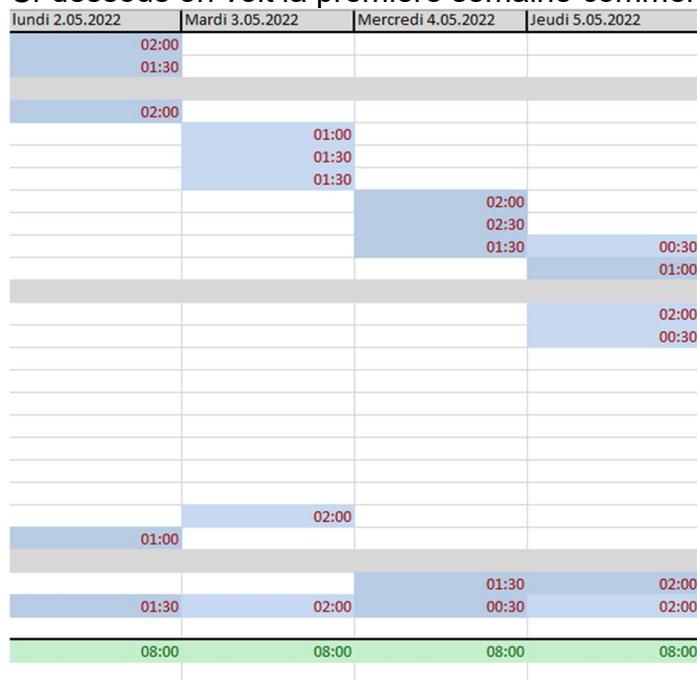
6. Planification

6.0.1 Planning prévisionnel

Voici ci-dessous toutes les tâches de mon planning prévisionnel et avec le temps estimer de chaque tâche. J'ai séparé le développement en deux parti une partie vue seulement visuelle et une deuxième partie seulement la partie back-end.

Tâches à réaliser	Temps
Lire et compréhension de l'énoncer	02:00
Planification	10:00
HTML/CSS (vue)	
Maquette	02:00
vue de login	01:00
vue accueil	01:30
vue par catégorie	01:30
vue du profil de l'utilisateur	02:00
vue du calendrier	02:30
vue ajouter et modifier de matériel	02:00
vue les prêts en attente (admin)	01:00
FONCTIONS (modèle + contrôleur)	
T1 système de login	02:00
T2 affichage du matériel	02:00
T3 affichage du matériel par catégorie	02:30
T4 affichage du profil(modification)	02:00
T5 ajout, modification et suppression du matériel	02:00
T6 accepte les prêts	02:00
T7 calendrier emprunt du matériel	03:00
T8 affichage des utilisateurs(admin)	02:00
T9 pouvoir ajouter un commentaire	02:00
mvc	02:00
création d'un répertoire git et un readme	01:00
GENERAL	
Test et débogage	16:00
Documentation	24:00
	88:00

Ci-dessous on voit la première semaine comment j'ai réparti les tâches.



Ci-dessous on voit la deuxième semaine comment j'ai réparti les tâches.

Ci-dessous on voit la dernière semaine comment j'ai réparti les tâches.

Lundi 16.05.2022	Mardi 17.05.2022	Mercredi 18.05.2022	Total
			02:00
	02:30	06:00	10:00
			02:00
			01:00
			01:30
			01:30
			02:00
			02:30
			02:00
			01:00
			02:00
			02:00
			02:30
			02:00
			02:00
			02:00
			02:00
			03:00
			02:00
			02:00
	02:00		02:00
			02:00
			01:00
	02:30	01:00	16:00
	03:30	04:30	24:00
			00:00
08:00	08:00	08:00	
			88:00
			88:00

6.0.2 Planning Réel

Voici ci-dessous toutes les tâches de mon planning réel et avec le temps que sa ma pris à réaliser les tâches.

Tâches à réaliser	Temps
Lire et compréhension de l'énoncer	02:00
Planification	10:00
HTML/CSS (vue)	
Maquette	02:00
vue de login	01:00
vue accueil	01:30
vue par catégorie	01:30
vue du profil de l'utilisateur	02:00
vue du calendrier	02:30
vue ajout et modifier de matériel	02:00
vue les prêts en attente (admin)	01:00
FONCTIONS (modèle + contrôleur)	
T1 système de login	02:00
T2 affichage du matériel	02:00
T3 affichage du matériel par catégorie	02:30
T4 affichage du profil(modification)	02:00
T5 ajout, modification et suppression du matériel	02:00
T6 accepte les prêts	02:00
T7 calendrier emprunt du matériel	03:00
T8 affichage des utilisateur(admin)	02:00
T9 pouvoir ajouter un commentaire	02:00
mvc	02:00
création d'un répertoire git et un readme	01:00
GENERAL	
Test et débogage	16:00
Documentation	24:00
	88:00

Ci-dessous voici le déroulement de la première semaine de tpi. On peut déjà voir une différence entre le planning prévu et le planning réel j'ai décidé de changer l'ordre des tâches faire les tâches les plus simples en premier et les plus compliqués à la fin.



Ci-dessous voici le déroulement de la deuxième semaine de tpi. On peut toujours voir la différence entre les plannings.

Lundi 9.05.2022	Mardi 10.05.2022	Mercredi 11.05.2022	Jeudi 12.05.2022
		01:30	01:00
		01:00	
	02:30:00		
	01:00	01:00	
		02:00	
			01:30
		01:00	01:00
	02:30	02:00	02:30
	02:00	02:00	02:00
			02:30
08:00	08:00	08:00	08:00

Ci-dessous la dernière semaine du déroulement du tpi.

Lundi 16.05.2022	Mardi 17.05.2022	Mercredi 18.05.2022	Total
			02:00
			10:00
	02:30	06:00	
			02:00
			01:00
			01:30
			01:30
			02:00
			02:30
			02:00
			01:00
			02:00
			02:00
			02:30
			02:00
			02:00
			02:00
01:30			03:00
	02:00		02:00
			02:00
			02:00
			01:00
			02:00
			02:00
			02:30
			02:00
			02:00
			02:00
			01:00
02:30			16:00
04:00	03:30	02:00	24:00
			00:00
08:00	08:00	08:00	88:00
			88:00

6.1 Product backlog

Nom	T1 système de login
Description (user story)	en tant que visiteur permettre de s'inscrire ou de se connecter
Critère d'acceptation	
Priorité	P0 : Bloquant

Nom	T2 affichage du matériel
Description (user story)	en tant que visiteur et utilisateur, je dois voir le matériel disponible à emprunts
Critère d'acceptation	
Priorité	P0 : Bloquant

Nom	T3 affichage du matériel par catégorie
Description (user story)	en tant que visiteur et utilisateur, je dois voir le matériel par catégorie
Critère d'acceptation	
Priorité	P1 : Critique

Nom	T4 affichage et modification du profil
Description (user story)	en tant qu'utilisateur, je dois pouvoir voir mon profil et le modifier
Critère d'acceptation	
Priorité	P2 : Important

Nom	T5 ajout, modification et suppression du matériel
Description (user story)	en tant qu'administrateur, je peux ajouter, modifier et supprimer du matériel
Critère d'acceptation	
Priorité	P1 : Critique

Nom	T6 accepter les prêts
Description (user story)	en tant qu'administrateur, je dois accepter les prêts qui sont en attente
Critère d'acceptation	
Priorité	P1 : Critique

Nom	T7 calendrier d'emprunt du matériel
Description (user story)	en tant qu'utilisateur, lors du choix du matériel un calendrier s'affiche pour choisir la date de l'emprunt
Critère d'acceptation	
Priorité	P0 : bloquant

Nom	T8 affichage des utilisateurs
------------	--------------------------------------

Description (user story)	en tant qu'administrateur on peut voir tous les utilisateurs
Critère d'acceptation	
Priorité	P3 : Secondaire

Nom	T9 pouvoir ajouter un commentaire
Description (user story)	en tant qu'utilisateur on peut ajouter un commentaire lors du rendu du matériels
Critère d'acceptation	
Priorité	P3 : Secondaire

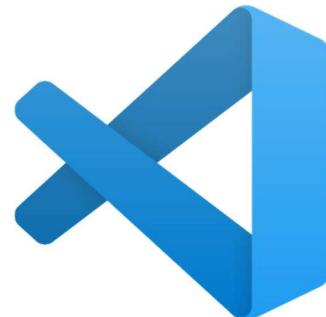
6.2Diagramme de Gantt

Voici le diagramme utilisé lors de la conception du site. Il m'a permis d'être organisé ainsi que de savoir précisément les tâches à réaliser de chaque jour.

7. Généralités concernant l'implémentation

7.1 librairies et outils externes utilisés

Visual Studio Code ou **VsCode** est un environnement de développement (IDE) développé par *Microsoft*. C'est selon moi le meilleur outil de développement on peut quasiment tout faire. Avec le nombre d'extensions qui existe.



Pour ma part, j'ai utilisé VsCode pour quelque des ces extensions. PHP getter and setter qui permet de créer automatiquement les get et set des propriétés de classe. Cela permet de gagner du temps. VsCode nous propose aussi des extensions pour l'utilisation d'un débogueur en PHP, ou bien de formater votre code ce qui permet de bien structurer votre code, que j'ai évidemment utilisé.

L'application Ubuntu on Windows ma permit de faire fonctionner localement des scripts PHP. J'ai installé tous les packages nécessaires pour créer un serveur LAMP comprenant les trois serveurs *apache2*, *mariaDB* et *PHP*. Il est simple d'utilisation si on a un minimum de connaissant de l'environnement linux.



Pour ma part, j'ai utilisé ce logiciel pour la simplicité et la rapidité en cas de problème on peut rapidement redémarrer nos service *apache* ou *mariaDB*.

La version d'*apache2* utilisé : 2.4.41

La version de *PHP* utilisé : PHP 8.1.5 .

La version de *mariaDB* utilisé : mysqld Ver 10.3.34-MariaDB-0ubuntu0.20.04.1 for debian-linux-gnu on x86_64 (Ubuntu 20.04).

DBeaver est un logiciel client SQL qui offre un outil d'administration de bases de données. Il fournit un éditeur qui prend en charge la compléction de code et la coloration syntaxique. Son interface graphique permet de facilement effectuer des tâches qui peuvent sembler compliquées. Il offre également la possibilité d'afficher sous forme de diagramme une base de données.



Pour ma part, ce logiciel m'a permis de modifier la base de données avec facilité. Mais pour pouvoir créer et vérifier facilement mes requêtes avant de les implémenter dans mon code. J'ai directement fait avec Ubuntu je me suis connecter en ligne de commande à MySQL. l'éditeur script SQL je n'arrive pas à comprendre le fonctionnement.

Git est un outil de gestion de version décentralisé, gratuit et open source. Il est simple à apprendre et permet de gérer du plus petit projet au plus grand. Pour exploiter cet outil, j'ai utilisé l'intermédiaire **GitHub** qui est service web d'hébergement de projet. De plus étant étudiant, je profite de l'intégralités qu'il propose.



Pour ma part, j'ai utilisé ces outils dans le but de pouvoir sauvegarder mes projets à un autre endroit que sur mon ssd.

Bootstrap 5 est un Framework CSS qui permet de concevoir et personnalisez très rapidement des sites web, en prenant compte de la taille de l'écran utilisé. C'est l'un des Framework CSS les plus utilisés au monde. Il est très complet et permet l'utilisation de variable, de système de grille responsive, ainsi que de nombreux composants préconstruits. Par conséquent, certaines de ces fonctionnalités utilisent de puissants plug-ins JavaScript.



Pour ma part, j'ai utilisé ce Framework CSS, car il m'a permis d'avoir un site web visuellement bon et rapide sans perdre beaucoup de temps. J'ai choisi ce Framework CSS, parce que j'ai déjà utilisé plusieurs fois dans plusieurs projets, ayant la main dessus, je tenais à le réutiliser. De plus il y a beaucoup d'exemple ce qui permet de trouver la bonne class.

7.1.2 langages utilisés

Le *HyperText Markup Language*, généralement abrégé *HTML* ou, dans sa dernière version, *HTML5*, est le langage de balisage conçu pour représenter les pages web.

Les *feuilles de style en cascade*, généralement appelées CSS de l'anglais *Cascading Style Sheets*, forment un langage informatique qui décrit la présentation des documents *HTML*.



PHP: Hypertext Preprocessor, plus connu sous son sigle *PHP* (sigle auto-référentiel), est un langage de programmation libre, principalement utilisé pour produire des pages Web dynamiques via un serveur *HTTP*, mais pouvant également fonctionner comme n'importe quel langage interprétré de façon locale. *PHP* est un langage impératif orienté objet.



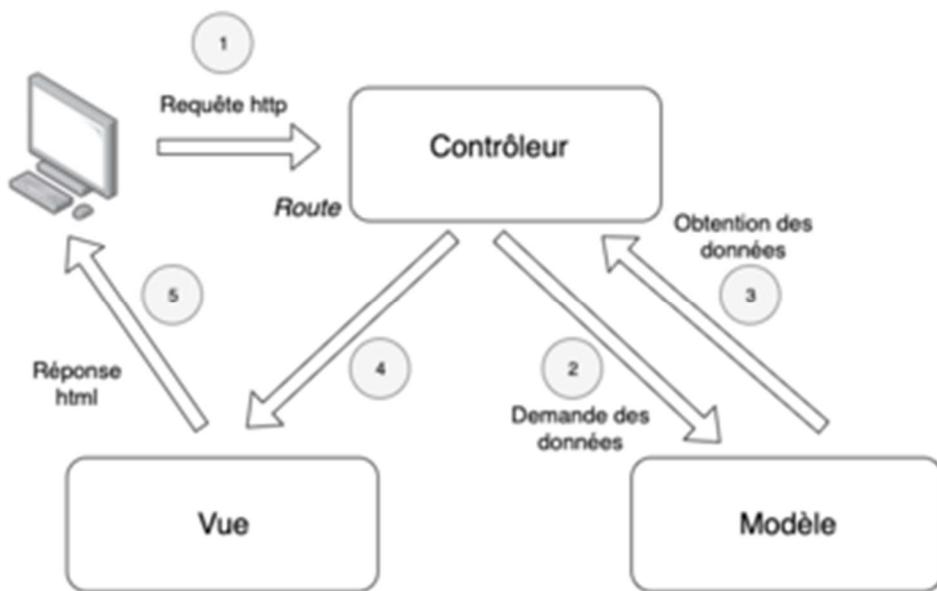
7.2 Architecture du projet

7.2.1 modèle Vue Contrôleur

L'architecture de mon projet se base sur le motif Modèle-vue-contrôleur ou MVC. Comme on peut s'y attendre, ce type d'architecture sépare le projet en trois parties distinctes, la première étant le modèle, la seconde la vue et la dernière le contrôleur. Chaque de ces parties à sa propre fonction.

- Le modèle contient ou récupère les données à afficher.
- La vue contient la partie graphique du projet, autrement dit l'affichage.
- Le contrôleur contient la logique concernant les actions, c'est lui qui va filtrer les données et les analyser.

Voici une image qui schématise bien l'architecture MVC :

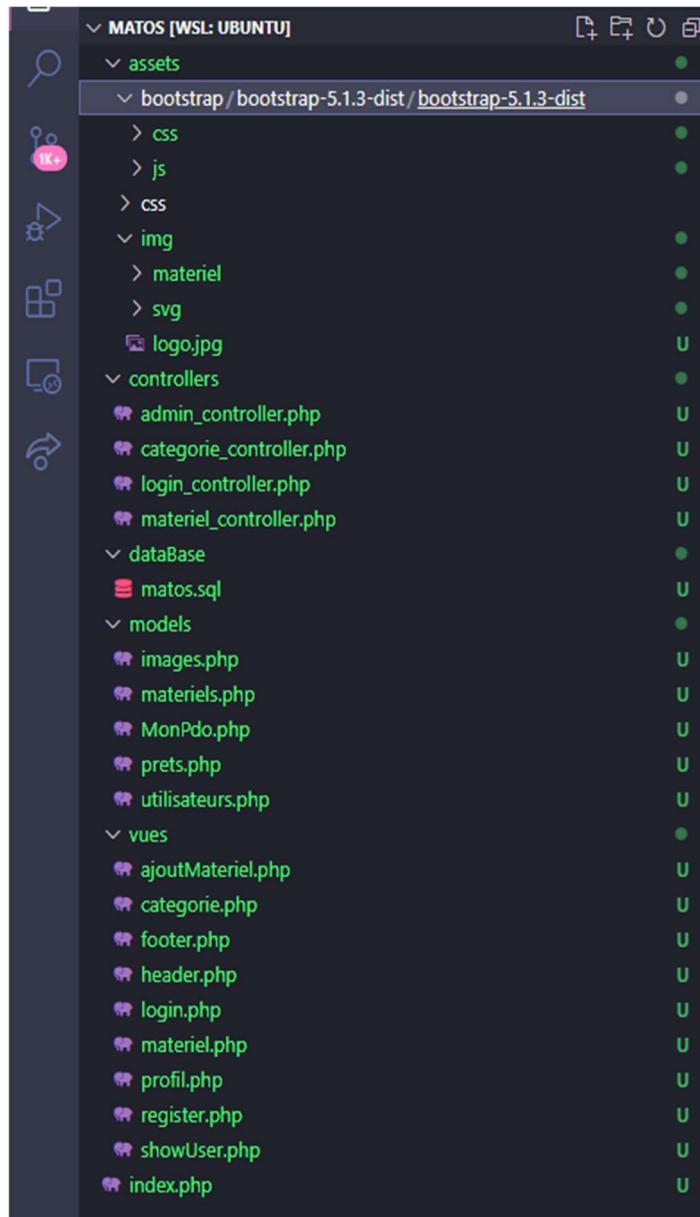


Si j'ai choisi cette façon de procéder, c'est parce que l'architecture MVC permet de séparer les couches, ce qui améliore considérablement la maintenabilité de l'application.

7.2.2 Arborescence

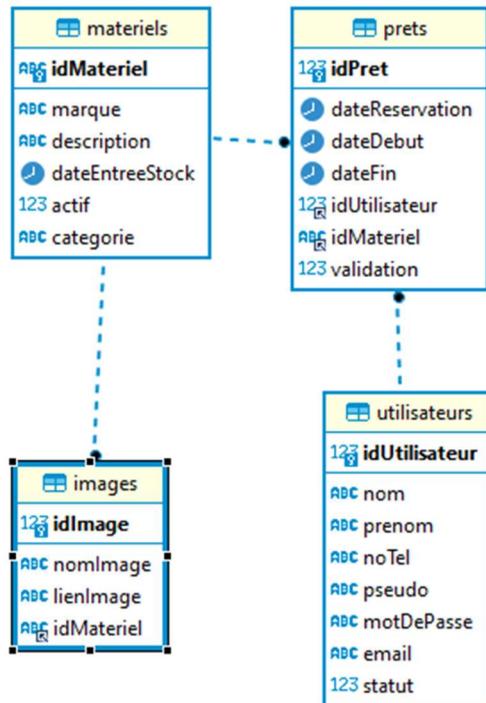
L'arborescence de mon projet s'organise autour de plusieurs répertoires importants :

- **asset** : contiens toutes les ressources du projet.
 - **Css** : comporte les fichiers de style du site web.
 - **Img** : inclus l'entièreté des images de l'application web.
- **Controllers** : englobe tous les contrôleurs du projet qui détiennent la logique du projet.
- **Database** : contiens la structure de la base de données et avec quelques données.
- **Models** : renferme les différents modèles du projet, ils disposent des données du projet.
- **Vues** : contiens les pages du projet, c'est lui qui fait le lien entre l'utilisateur et le site.



7.3 bases de données

La structure et ainsi le script SQL de la base de données a été donné par le responsable du TPI le diagramme est ci-dessous.



J'ai alors modifié quelques champs. Elle est encodée en utf8 avec comme collation par défaut utf8_general_ci et utilise un moteur InnoDB.

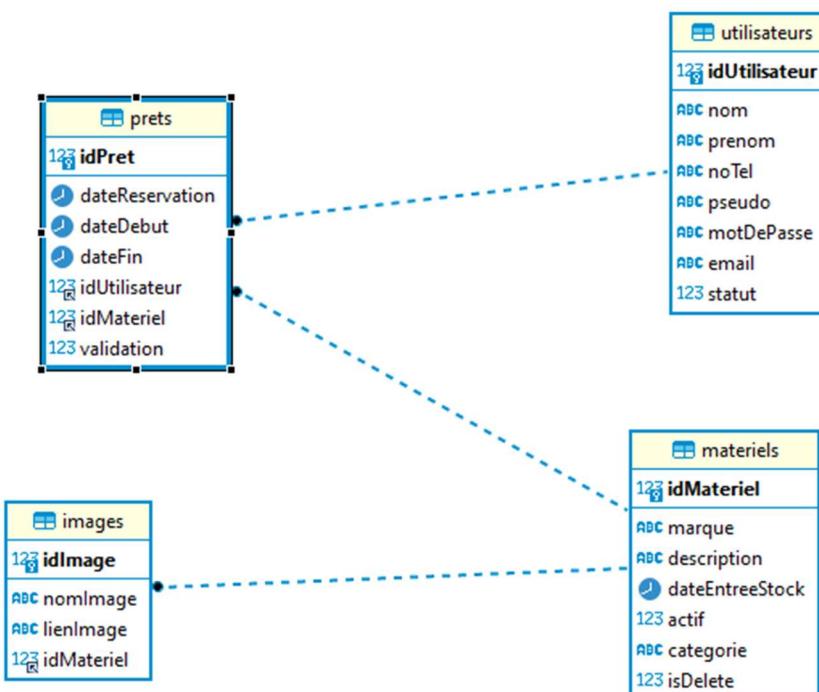


Figure 2 - modèle de la base de données de matos

7.3.1 Dictionnaire de données

Voici le dictionnaire de données de mon application :

Tables Images :

Colonne	Type	Null	Valeur par Défaut	Commentaires
idImage (Primaire)	int(11)	Non		
nomImage	varchar(100)	Non		
lienImage	varchar(100)	Non		
idMateriel	int(11)	Non		

Index :

Nom index	Type	Unique	Compressé	Colonne	Cardinalité	Interclassement	Null
PRIMARY	BTREE	Oui	Non	idImage	16	A	Non
idMateriel	BTREE	Non	Non	idMateriel	16	A	Non

Tables Matériels :

Colonne	Type	Null	Valeur par défaut	Commentaires
idMateriel (Primaire)	int(11)	Non		
marque	varchar(50)	Non		
description	text	Non		
dateEntreeStock	date	Non		
actif	tinyint(1)	Non	1	1 = disponible 0= pas disponible
categorie	varchar(50)	Non		
isDelete	tinyint(4)	Non	1	1=visible 0=invisible

Index :

Nom de l'index	Type	Unique	Compressé	Colonne	Cardinalité	Interclassement	Null
PRIMARY	BTREE	Oui	Non	idMateriel	15	A	Non

Tables Prêts :

Colonne	Type	Null	Valeur par défaut	Commentaires
idPret (Primaire)	int(11)	Non		
dateReservation	date	Non		
dateDebut	date	Non		
dateFin	date	Non		

idUtilisateur	int(11)	Non		
idMateriel	int(11)	Non		
validation	tinyint(1)	Non	0	0 = en attente & 1 = valider

Index :

Nom de l'index	Type	Unique	Compressé	Colonne	Cardinalité	Interclassement	Null
PRIMARY	BTREE	Oui	Non	idPret	2	A	Non
idUtilisateur	BTREE	Non	Non	idUtilisateur	2	A	Non
				idMateriel	2	A	Non
fk_idMateriel_prets	BTREE	Non	Non	idMateriel	2	A	Non

Tables Utilisateurs :

Colonne	Type	Null	Valeur par défaut	Commentaires
idUtilisateur (Primaire)	int(11)	Non		
nom	varchar(50)	Non		
prenom	varchar(50)	Non		
noTel	varchar(20)	Non		
pseudo	varchar(50)	Non		
motDePasse	varchar(32)	Non		
email	varchar(255)	Non		
statut	tinyint(1)	Non	1	1 = enseignant / 2 = admin
isDeleted	tinyint(4)	Non	0	0 = activer / 1 = désactiver

Nom de l'index	Type	Unique	Compressé	Colonne	Cardinalité	Interclassement	Null
PRIMARY	BTREE	Oui	Non	idUtilisateur	2	A	Non

7.4 Fonctionnalités intégrées

Les fonctionnalités que j'ai implémentées dans mon projet sont décrites ci-dessous. Je les ai séparés par rapport aux pages sur lesquels elle agisse.

7.5 Navigation

7.5.1 fonction des liens

- **Accueil** renvoi sur la page *index.php*
- **Catégorie** renvoi sur la page *categorie.php*
- **Barre recherche** renvoi les produits rechercher
- **Ajout** renvoi sur la page *ajoutMateriel.php*
- **Gérer/list enseignant** renvoi la liste de tous les enseignant
- **Gérer/list prêt** renvoi la liste de tous les prêts
- **Profil** renvoi sur la page *profil.php*
- **Se déconnecter** te déconnecte du site

7.5.1.1 Visiteur

Accueil Catégorie marque Soumettre Login Register

Matos

profil Se Dé

7.5.1.2 Enseignant

7.5.1.3 Admin

7.6 Connexion

7.6.1 Login

- a) Saisir l'email
- b) Saisir le password
- c) Lien pour aller à register
- d) Bouton pour se connecter

7.6.2 Register

Register

First Name :

Enter your First name .. **A**

Last Name :

Enter your Last name .. **B**

E-mail :

Enter your email .. **C**

Pseudo :

Enter your pseudo .. **D**

Numéro de téléphone :

enter your number .. **E**

Password :

Enter your password .. **F**[Déjà membre ? Connectez-vous ici](#) **G****Register****H**

- a) Saisir **le prénom**
- b) Saisir **le nom**
- c) Saisir **l'email**
- d) Saisir **le pseudo**
- e) Saisir **le numéro de téléphone**
- f) Saisir **le password**
- g) lien pour aller au login
- h) bouton pour s'inscrire

7.7 Accueil

Accueil Catégorie marque Soumettre Login Register

A

B

C

D

Apple Macbook Pro Processeur Apple M1 Max à dix coeurs avec 8 coeurs de performance.	HP Elite Dragonfly G2 Intel Core i7-1165G7 (2.8 GHz/4.7 GHz Turbo) Quad Core,	MSI Stealth GS77 Processeur Intel Core i9-12900H Tedra-Core de 3.8 à 5.0 GHz.
plus info	plus info	plus info

© 2022 Copyright : David & **CPFL**

- navbar
- le produit à louer avec l'image, marque et description
- bouton pour voir en plus grand
- footer

7.8 Matériel

7.8.1 visiteur et enseignant

Accueil Soumettre Login Register

A

B

C

Q

Apple Macbook Pro
 Processeur Apple M1 Max à dix coeurs avec 8 coeurs de performance.

louer

© 2022 Copyright : David & **CPFL**

- navbar

- b) le matériel à louer avec l'image, marque et description
- c) bouton pour louer le matériel
- d) footer

7.8.2 admin

The screenshot shows a web page with a header "Matos" and a "Soumettre" button. A blue letter "A" is handwritten above the page. Below the header is a photograph of an Apple MacBook Pro. A blue mouse cursor icon is drawn over the trackpad area. The text "Apple Macbook Pro" is displayed below the image, followed by the text "Processeur Apple M1 Max à dix coeurs avec 8 coeurs de performance.". At the bottom, there are two buttons: a blue "louer" button and a red "supprimer" button. A blue letter "D" is handwritten next to the "supprimer" button. A blue letter "C" is handwritten next to the "louer" button. A blue arrow points from the "louer" button towards the "supprimer" button. The footer contains the text "© 2022 Copyright : David & CEPTE".

- a) navbar
- b) le matériel à louer avec l'image, marque et description
- c) bouton pour louer le matériel
- d) bouton pour supprimer le matériel
- e) footer

7.9 Catégorie

© 2022 Copyright : David & CEFPI-J

- navbar
- affichage de tous les catégorie existant
- bouton pour lister le matériel par la catégorie sélectionner
- footer

7.10 MatérielByCatégorie

© 2022 Copyright : David & CEFPI-J

- navbar
- le matériel à louer avec la catégorie selectionner

- c) bouton pour voir en plus grand
- d) footer

7.11 ajout de Matériel

Seulement en mode admin qu'on peut accéder à la page

Matos Accueil Catégorie marque ajout gérer ▾

Soumettre

ajout de matériel

marque : catégorie :

images du matériel :
Choisir un fichier Aucun fichier n'a été sélectionné

description :

ajouter F

© 2022 Copyright : David & CEFTE-J

- a) navbar
- b) saisir la marque
- c) sélectionner la catégorie
- d) choisir une image
- e) saisir une description
- f) bouton ajouter un matériel
- g) footer

G

7.12 page liste enseignant

E-mail	Pseudo	Numéro de téléphone	Statut
admin@admin.ch	admin	0777777777	★ admin
david.mchdb@eduge.ch	azrod2K	0766154106	☛ enseigants
test@test.com	tesst	798561535	☛ enseigants

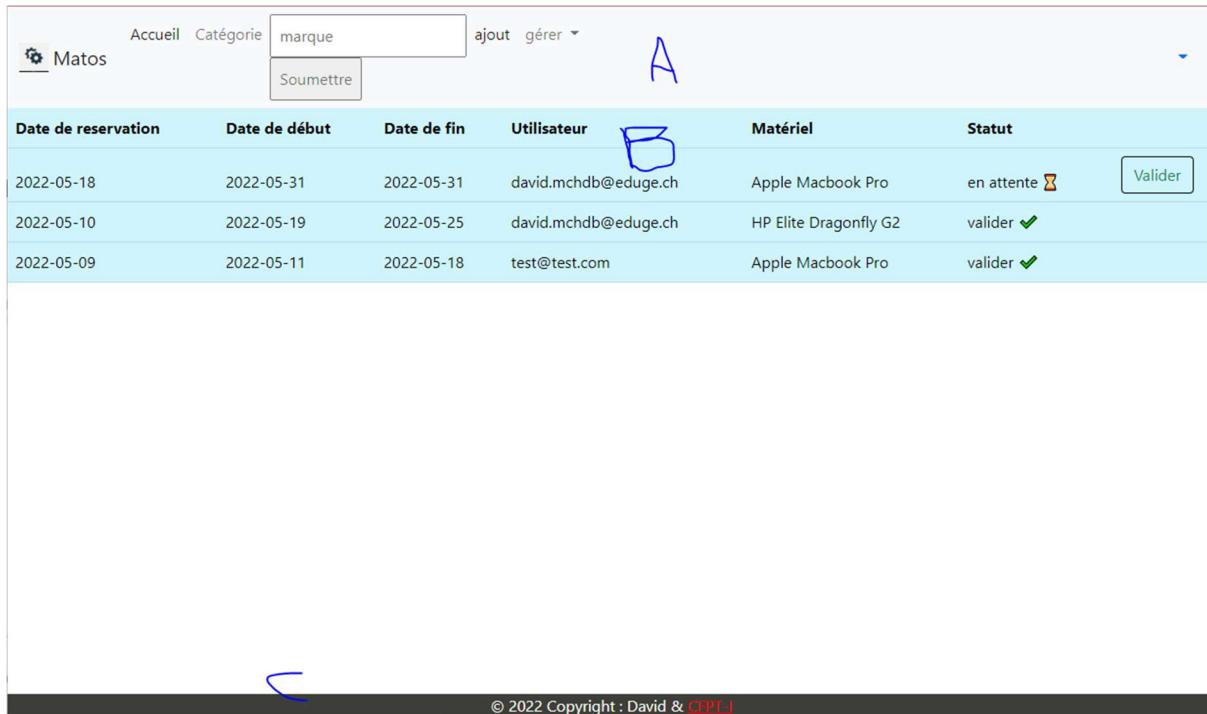
© 2022 Copyright : David & CIPT.I

- a) navbar
- b) affichage de tous les enseignant comme information l'email, pseudo, numéro de téléphone et statut plus un bouton qui supprime l'enseignant
- c) footer

la requête qui permet d'afficher la liste d'enseignant :

```
SELECT * FROM utilisateurs WHERE isDeleted = 0 ORDER BY statut DESC
```

7.13 page liste prêt



Accueil Catégorie marque ajout gérer ▾

Soumettre

A

B

C

Date de réservation	Date de début	Date de fin	Utilisateur	Matériel	Statut	
2022-05-18	2022-05-31	2022-05-31	david.mchdb@eduge.ch	Apple Macbook Pro	en attente ⏳	<button>Valider</button>
2022-05-10	2022-05-19	2022-05-25	david.mchdb@eduge.ch	HP Elite Dragonfly G2	valider ✓	
2022-05-09	2022-05-11	2022-05-18	test@test.com	Apple Macbook Pro	valider ✓	

© 2022 Copyright : David & CEPI

- a) navbar
- b) affichage de tous les prêt comme information la date de réservation, date de début, date de fin, utilisateur, matériel et en plus avec un bouton pour valider le prêt en attente
- c) footer

la requête qui permet d'afficher la liste des prêts :

```
SELECT idPret,dateReservation,dateDebut,dateFin,email,marque,validation FROM
prets as p,materiels as m, utilisateurs as u WHERE u.idUtilisateur =
p.idUtilisateur and m.idMateriel = p.idMateriel ORDER BY validation ASC
```

8. Plan et rapport de test

8.1 Périmètre de test

J'ai choisi des protocoles de tests pour les actions qu'un *visiteur*, *enseignant* ou *Administrateur* pourraient effectuer sur le site web.

8.2 Equipement de test

Lors du développement du projet, j'ai utilisé l'environnement suivant pour effectuer les tests :

- Microsoft EDGE Version 101.0.1210.39 (Version officielle) (64 bits) sur Windows 10 éducation, version 21H2

Servers :

- Apache2 2.4.41
- MySQL 10.3.34-MariaDB

8.3 Scénarios de test

J'ai élaboré les scénarios de tests suivants afin de m'assurer que les besoins du cahier des charges soient respectés. Les scénarios de tests sont précis dans le but d'être facilement reproductibles par une autre personne si c'est nécessaire.

Nº Test	Auteur	Scénario	Résultat Attendu
T01	Machado	Création d'un enseignant cliquer sur le button register en haut à droite. Affiche le formulaire. Sur le formulaire remplissez tous les champs puis cliquer sur le button en bas du formulaire register.	Message l'utilisateur à bien été créée
T02	Machado	connexion cliquer sur le button Login. Remplissez tous les champs du formulaire et appuyer sur le button login	Message la connexion réussite
T03	Machado	page d'accueil (visiteur ou enseignant) 1. Affichage de la barre de navigation. 2. Affichage du matériel disponible à louer. 2.1 Affiche un button dans chaque matériel avec le texte plus d'info 3. Affichage du bas de page.	Message 1. La nav Affiche le logo, l'accueil, catégorie, profil et déconnexion. 2. Le body affiche le matériel disponible. 2.1 affiche le matériel sélectionné en une seule page et avec un button louer. 3. Le pied de page

			affiche mon prénom et le cfpt.
T04	Machado	page catégorie	Message
		<p>1. Affichage de la barre de navigation.</p> <p>2. Affichage de tous les catégories existantes.</p> <p>2.1 Dans chaque catégorie un button afficher avec le texte lister.</p> <p>3. Affichage du bas de page.</p>	<p>1. La nav Affiche le logo, l'accueil, catégorie, profil et déconnexion.</p> <p>2. Le body affiche les catégories existantes disponibles .</p> <p>2.1 affiche tout le matériel disponible par catégorie sélectionnée.</p> <p>3. Le pied de page affiche mon prénom et le cfpt.</p>
T05	Machado	Page Louer	Message
		après avoir cliqué sur le button louer un premier calendrier s'affiche avec les dates après avoir sélectionné une date le deuxième s'affiche pour la durée du prêt.	le calendrier s'affiche avec des couleurs vert la date est disponible, rouge le matériel n'est pas disponible ce jour-là et en noir le matériel est endommagé ou en réparation.
T06	Machado	Page Profil	Message
		affiche le numéro de téléphone, le pseudo et le champs mot de passe. En bas du formulaire un button update	Affiche bien les informations personnelles de la personne connectée. pouvoir modifier le pseudo , le numéro ou le mot de passe.
T07	Machado	page d'accueil (admin)	Message
		<p>1. Affichage de la barre de navigation.</p> <p>2. Affichage du matériel disponible à louer.</p> <p>2.1 Affiche un button dans chaque matériel avec le texte plus d'info</p> <p>3. Affichage du bas de page.</p>	<p>1. La nav Affiche le logo, l'accueil, catégorie, ajout, liste enseignant, liste de prêt, profil et déconnexion.</p> <p>2. Le body affiche le matériel disponible.</p> <p>2.1 affiche le matériel sélectionné en une seule page et avec un button louer et un</p>

			button update. 3. Le pied de page affiche mon prénom et le cfpt.
T08	Machado	Page d'ajout matériel	Message
		le formulaire d'ajout de matériel s'affiche. Remplissez tous les champs puis appuyer sur le button ajouter.	si tous les champs on était remplis le matériel a bien été ajouter.
T09	Machado	Page liste enseignant	Message
		affiche tous les enseignant inscrit.	affiche l'email, le pseudo, le numéro, et le statut (admin ou enseignant) plus un button pour supprimer le compte.
T10	Machado	Page liste prêt	Message
		affiche tous les prêts	affiche la date de réservation, la date de début, la date de fin, utilisateur (mail), le matériel emprunter, le statut(en attente ou valider), et un button valider pour valider les prêts en attente.
T11	Machado	Page d'update un matériel	Message
		le formulaire de modifier un matériel s'affiche. Les champs sont déjà remplis en fonction du matériel sélectionner. Modifier les champs que vous voulez puis appuyer sur l'update	le matériel a bien été modifier.

8.4 Rapport de test

Date	Testeur	Nº test	Résultat obtenu	OK/KO
03.05.22	Machado	T01	Message le client est créé.	OK
03.05.22	Machado	T02	Message connexion réussite.	OK
05.05.22	Machado	T03	Message	OK

			tout s'affiche correctement.	
09.05.22	Machado	T04	Message	OK
			tout s'affiche correctement.	
12.05.22	Machado	T05	Message	KO
			le calendrier s'affiche mais pas les couleurs et il n'y a pas de deuxième calendrier.	
05.05.22	Machado	T06	Message	OK
			affiche bien les informations personnelles.	
10.05.22	Machado	T07	Message	OK
			tout s'affiche correctement.	
10.05.22	Machado	T08	Message	OK
			le matériel est bien créé.	
12.05.22	Machado	T09	Message	OK
			tous l'enseignants s'affichent correctement.	
12.05.22	Machado	T10	Message	OK
			tous les prêts s'affichent correctement.	
17.05.22	Machado	T11	Message	KO
			Pas de système de update	

9. Conclusion

9.1 Difficultés rencontrées

Je n'ai pas vraiment eu de gros problèmes. J'ai eu un problème avec la base de données dans la table matériels sur le champs idMateriel qui était un type varchar et j'avais besoin d'un type Int. Lors du change je ne pouvais pas changer à cause des restriction je les ai enlevés ça ne marcher pas sa fessais depuis 45minute que j'étais bloqué. Ducoup j'ai décidé de prendre le script SQL et le changer directement dans le script.

L'autre problème était avec le calendriers je n'ai pas réussi à faire un calendrier comme le cahier des charges demander. Je n'avais jamais fait un calendriers de ce type la j'ai essayé de tout les moyens mais en vain.

9.2 Améliorations possibles

1. Le double calendriers demande plus le choix de la date du rendu du matériel.
2. Pouvoir update un matériel en mode admin

3. Pouvoir update un profil enseignant en mode admin directement sur le site.

9.3 Bilan personnel

Je suis satisfait du résultat final. Même si je n'ai pas pu tous finir, le projet que j'avais à réaliser pourrait me servir plus tard vu que dans quasiment tous les domaines on a besoins d'un gestionnaire.

C'est plutôt de mon attitude que je ne suis pas satisfait j'ai pris la confiance les premier jours, parce que j'avais pris de l'avance et au final sa ma pénaliser que j'ai pas pu finir à 100%

Si je devais refaire ce mon projet je changerais la façon de le réaliser, je regarderais avec plus de minutie le cahier des charges, afin de bien savoir avec exactitude chaque petite partie du projet. Comme ça je serai exactement le temps que sa prendra pour chaque tache et mieux me repartie pour tout finir dans les temps.

9.4 Remerciements

Je tiens à remercier :

- Monsieur Garcia pour son accompagnement durant mon TPI Madame Terrier pour ses précieux conseils sur la documentation et l'organisation qu'il faut adopter sur des travaux comme celui-ci.
- Ma famille qui m'a aidé à garder le moral dans les moments les plus difficiles de ce TPI.

10. Annexe

Pour les question de PHP je suis allé sur les sites suivant : php.net ou staroverflow.

Pour le calendriers je l'ai trouvé sur ce site : <https://startutorial.com/view/how-to-build-a-web-calendar-in-php>.

Pour les question de MySQL sur le site de staroverflow.

c:\tpi\Matos\index.php

```
1 <!--
2 Auteur: David Machado
3 Date: 18.05.2022
4 Projet: Matos      -->
5
6 <?php
7 session_start();
8 if (!isset($_SESSION['userConnected'])) {
9     $_SESSION['userConnected'] = [
10         "email" => "",
11         "online" => false,
12         "idutilisateur" => "",
13         "statut" => null,
14         "noTel" => "",
15         "pseudo" => "",
16         "motDePasse" => ""
17     ];
18
19     $_SESSION['alertMessage'] = [
20         "type" => null,
21         "message" => null
22     ];
23 }
24 ini_set('display_errors', 1);
25 require("models/MonPdo.php");
26 require("models/images.php");
27 require("models/utilisateurs.php");
28 require("models/prets.php");
29 require("models/materIELS.php");
30
31 include "vues/header.php";
32
33 $uc = filter_input(INPUT_GET, 'uc') == null ? "accueil" : filter_input(INPUT_GET, 'uc'); // affiche la page accueil par défaut
34 switch ($uc) {
35     case 'accueil':
36         include "controllers/materIEL_controller.php";
37         break;
38     case 'login':
39         include "controllers/login_controller.php";
40         break;
41     case 'categorie':
42         include "controllers/categorie_controller.php";
43         break;
44     case 'materIEL':
45         include "controllers/materIEL_controller.php";
46         break;
47     case 'admin':
48         if ($_SESSION['userConnected']['idutilisateur']==2) {
49             include "controllers/admin_controller.php";
50         }
51         else {
52             header("Location: index.php?uc=accueil&action=show");
53         }
54         break;
55     case 'calendriers':
56         $idMaterIEL = filter_input(INPUT_GET, 'idMaterIEL');
57         require_once("function/calendrier.php");
58         include "vues/calendar.php";
59         break;
```

```
60     case 'search':
61         include "controllers/search_controller.php";
62         break;
63     case 'pret':
64         include "controllers/pret_controller.php";
65         break;
66 }
67 include "vues/footer.php";
68 error_reporting(E_ALL);
```

c:\tpi\Matos\controllers\admin_controller.php

```
1 <!--
2 Auteur: David Machado
3 Date: 18.05.2022
4 Projet: Matos
5 -!>
6 <?php
7 $action = filter_input(INPUT_GET, 'action');
8
9 switch ($action) {
10     case 'showAllUser':
11
12         if ($_SESSION['userConnected']['idutilisateur']==2) {
13             utilisateurs::getAllUser();
14         }
15         else {
16             header("Location: index.php?uc=accueil&action=show");
17         }
18
19         break;
20     case 'showAllPret':
21
22         if ($_SESSION['userConnected']['idutilisateur']==2) {
23             prets::getAllLoan();
24         }
25         else {
26             header("Location: index.php?uc=accueil&action=show");
27         }
28         break;
29     case 'accept':
30         $idPret = filter_input(INPUT_GET, 'idPret');
31         prets::setValidate($idPret);
32         header("Location: index.php?uc=accueil&action=show");
33         break;
34 }
```

c:\tpi\Matos\controllers\categorie_controller.php

```
1 <!--
2 Auteur: David Machado
3 Date: 18.05.2022
4 Projet: Matos
5 -!>
6 <?php
7 $action = filter_input(INPUT_GET, 'action');
8
9 switch ($action) {
10     case 'show':
11         include "vues/categorie.php";
12         break;
13     case 'liste':
14         $selected = filter_input(INPUT_GET, 'selected');
15         materiels::getMaterielByCategorie($selected);
16         include "vues/materielByCategorie.php";
17         break;
18 }
```

c:\tpi\Matos\controllers\login_controller.php

```
1 <!--
2 Auteur: David Machado
3 Date: 18.05.2022
4 Projet: Matos
5 -!>
6 <?php
7 $action = filter_input(INPUT_GET, 'action');
8 switch ($action) {
9
10     // affichage du formulaire de login
11     case 'show':
12         include 'vues/login.php';
13         break;
14
15     // affichage du formulaire de register
16     case 'showRegister':
17         include 'vues/register.php';
18         break;
19
20     // connexion de l'utilisateur
21     case 'validateLogin':
22         $mail = filter_input(INPUT_POST, 'email', FILTER_SANITIZE_EMAIL);
23         $password = filter_input(INPUT_POST, 'password', FILTER_SANITIZE_STRING);
24
25         if ($mail != "" && $password != "") {
26             // si les champs sont remplis
27             $user = new utilisateurs();
28             $user->setEmail($mail)
29                 ->setMotDePasse($password);
30
31             $reqResult = utilisateurs::CheckConnected($user);
32             if ($reqResult != false) {
33                 if ($reqResult->getEmail() == $mail && $reqResult->getMotDePasse() ==
utilisateurs::Crypter($password)) {
34                     $_SESSION['userConnected'] = [
35                         "email" => $reqResult->getEmail(),
36                         "online" => true,
37                         "idUtilisateur" => $reqResult->getIdUtilisateur(),
38                         "statut" => $reqResult->getStatut(),
39                         "noTel" => $reqResult->getNoTel(),
40                         "pseudo" => $reqResult->getPseudo(),
41                         "motDePasse" => $reqResult->getMotDePasse()
42                     ];
43                     header('Location: index.php?uc=accueil&action=show');
44                 } else {
45                     $_SESSION['alertMessage'] = [
46                         "type" => "danger",
47                         "message" => "Email ou mot de passe incorrect"
48                     ];
49                     header("Location: index.php?uc=login&action=show");
50                 }
51             } else {
52                 $_SESSION['alertMessage'] = [
53                     "type" => "danger",
54                     "message" => "Cet email n'existe pas"
55                 ];
56                 header("Location: index.php?uc=login&action=show");
57             }
58         } else {
59             $_SESSION['alertMessage'] = [
```

```

60         "type" => "danger",
61         "message" => "Merci de remplir les champs !"
62     ];
63     header("Location: index.php?uc=login&action=show");
64 }
65 break;
66
67 case 'validateRegister':
68     $firstName = filter_input(INPUT_POST, 'prenom', FILTER_SANITIZE_STRING);
69     $lastName = filter_input(INPUT_POST, 'nom', FILTER_SANITIZE_STRING);
70     $email = filter_input(INPUT_POST, 'email', FILTER_SANITIZE_EMAIL);
71     $noTel = filter_input(INPUT_POST, 'noTel', FILTER_SANITIZE_NUMBER_INT);
72     $password = filter_input(INPUT_POST, 'password', FILTER_SANITIZE_STRING);
73     $pseudo = filter_input(INPUT_POST, 'pseudo', FILTER_SANITIZE_STRING);
74
75     //si les champs ne sont pas vides
76     if ($firstName != "" && $lastName != "" && $email != "" && $pseudo != "" && $noTel
!= "" && $password != "") {
77         if (utilisateurs::IsEmailExisting($email, $pseudo) == false) {
78             $user = new utilisateurs();
79             $user->setNom($lastName)
80                 ->setPrenom($firstName)
81                 ->setEmail($email)
82                 ->setPseudo($pseudo)
83                 ->setNoTel($noTel)
84                 ->setMotDePasse($password);
85             utilisateurs::AddUser($user);
86             $_SESSION['alertMessage'] = [
87                 "type" => "success",
88                 "message" => "l'utilisateur a bien été crée"
89             ];
90             header("Location: index.php?uc=login&action=show");
91         } else {
92             $_SESSION['alertMessage'] = [
93                 "type" => "danger",
94                 "message" => "l'email ou le pseudo existe déjà"
95             ];
96             header("Location: index.php?uc=login&action=showRegister");
97         }
98     } else {
99         $_SESSION['alertMessage'] = [
100             "type" => "danger",
101             "message" => "Merci de remplir les champs !"
102         ];
103         header("Location: index.php?uc=login&action>ShowRegister");
104     }
105 break;
106
107 case 'showProfil':
108     include "vues/profil.php";
109     break;
110
111     // déconnexion de l'utilisateur
112 case 'deconnecter':
113     session_destroy();
114     session_unset();
115     header('Location: index.php?uc=accueil&action=show');
116     break;
117
118 case 'update':
119     $pseudo = filter_input(INPUT_POST, 'pseudo', FILTER_SANITIZE_STRING);

```

```
120 $noTel = filter_input(INPUT_POST, 'notel', FILTER_SANITIZE_NUMBER_INT);
121 $password = filter_input(INPUT_POST, 'mdp', FILTER_SANITIZE_STRING);
122 $sbm = filter_input(INPUT_POST, 'update');
123 if ($password == "") {
124     $user = new utilisateurs();
125     $user->setPseudo($pseudo)
126         ->setNoTel($noTel)
127         ->setMotDePasse($_SESSION['userConnected']['motDePasse'])
128         ->setIdUtilisateur($_SESSION['userConnected']['idUtilisateur']);
129     utilisateurs::Update($user);
130     header("Location: index.php?uc=login&action=showProfil");
131 }else if ($password != "") {
132     $user = new utilisateurs();
133     $user->setPseudo($pseudo)
134         ->setNoTel($noTel)
135         ->setMotDePasse($password)
136         ->setIdUtilisateur($_SESSION['userConnected']['idUtilisateur']);
137     utilisateurs::Update($user);
138 }
139 break;
140
141 default:
142     header('Location: index.php?uc=login&action=show');
143     break;
144 }
```

c:\tpi\Matos\controllers\materiel_controller.php

```
1 <!--
2 Auteur: David Machado
3 Date: 18.05.2022
4 Projet: Matos
5 -!>
6 <?php
7 $action = filter_input(INPUT_GET, 'action');
8 switch ($action) {
9
10     // affichage du formulaire de login
11     case 'show':
12         materiels::GetAllMateriel();
13         break;
14     case 'ajout':
15         if ($_SESSION['userConnected']['idutilisateur']==2) {
16             include "vues/ajoutMateriel.php";
17         }
18         else {
19             header("Location: index.php?uc=accueil&action=show");
20         }
21
22         break;
23     case 'validateAdd':
24
25         $marque = filter_input(INPUT_POST, 'marque', FILTER_SANITIZE_STRING); //récupération
de la marque
26         $categorie = filter_input(INPUT_POST, 'categorie', FILTER_SANITIZE_STRING);
//récupération de la catégorie
27         $desc = filter_input(INPUT_POST, 'description', FILTER_SANITIZE_STRING);
//récupération de la description
28         $image = $_FILES['image']; //récupération de l'image
29
30         // vérification que tout les champs sont bien remplis
31         if ($marque != "" && $categorie != "" && $desc != "" && $image['name'][0] != "") {
32
33             if (explode("/", $image["type"])[0] != "image") {
34                 $_SESSION['alertMessage'] = [
35                     'type' => "danger",
36                     'message' => "le fichier ne peut être que une image"
37                 ];
38                 header('Location: index.php?uc=materiel&action=ajout');
39             }
40             //début de la transaction
41             MonPdo::getInstance()->beginTransaction();
42
43             //on crée le matériel
44             $materiels = new materiels();
45             $materiels->setMarque($marque)
46                 ->setCategorie($categorie)
47                 ->setDescription($desc);
48             $idMateriel = materiels::AddMaterial($materiels);
49
50             $dirFile = "./assets/img/materiel/";
51             $randomName = uniqid() . "." . $image['name'];
52             if (move_uploaded_file($image['tmp_name'], $dirFile . $randomName)) {
53                 $images = new images();
54                 $images->setLienImage($dirFile)
55                     ->setNomImage($randomName)
56                     ->setIdMateriel($idMateriel);
57                 images::AddImage($images);
58             }
59         }
60     }
61 }
```

```

58     } else {
59         //si il y a une image qui ne se push pas on rollback et cancel les requêtes
60         MonPdo::getInstance()->rollback();
61         $_SESSION['alertMessage'] = [
62             'type' => "danger",
63             'message' => "une image n'a pas pu être publié !"
64         ];
65         header('Location: index.php?uc=materiel&action=ajout');
66     }
67     //on push les infos dans la base de données avec le commit
68     MonPdo::getInstance()->commit();
69
70     //message de success de création de matériel et de l'image
71     $_SESSION['alertMessage'] = [
72         'type' => "success",
73         'message' => "le matériel à bien été créé et tous les fichiers ont été
importés"
74     ];
75     header('Location: index.php?uc=accueil&action=show');
76 } else {
77     //retourne un message d'erreur si les champs ne sont pas remplis
78     $_SESSION['alertMessage'] = [
79         'type' => "danger",
80         'message' => "merci de remplir tous les champs !"
81     ];
82     header("Location: index.php?uc=materiel&action=ajout");
83 }
84 break;
85 case 'delete':
86     $idMateriel=filter_input(INPUT_GET,'idMateriel');
87     materiels::SetDelete($idMateriel);
88     header("Location: index.php?uc=accueil&action=show");
89 break;
90 case 'info':
91     $idMateriel = filter_input(INPUT_GET,'idMateriel');
92     $materiel = materiels::getMaterielById($idMateriel);
93     $image = images::getImageById($idMateriel);
94     include "vues/materiel.php";
95 break;
96 }

```

c:\tpi\Matos\controllers\pret_controller.php

```
1 <!--
2 Auteur: David Machado
3 Date: 18.05.2022
4 Projet: Matos
5 -!>
6 <?php
7 $action = filter_input(INPUT_GET, 'action');
8
9 switch ($action) {
10     case 'validate':
11         $idmateriel = filter_input(INPUT_GET, 'idMateriel');
12         $date = filter_input(INPUT_GET, 'date');
13         $currentDate = date('Y-m-d');
14         $pret = new pret();
15         $pret->setDateReservation($currentDate)
16             ->setDateDebut($date)
17             ->setDateFin($date)
18             ->setIdMateriel($idmateriel)
19             ->setIdUtilisateur($_SESSION['userConnected']['idUtilisateur']);
20         pret::addPret($pret);
21         header("Location: index.php?uc=accueil&action=show");
22         break;
23 }
```

c:\tpi\Matos\controllers\search_controller.php

```
1 <!--
2 Auteur: David Machado
3 Date: 18.05.2022
4 Projet: Matos
5 -!>
6 <?php
7 $action = filter_input(INPUT_GET, 'action');
8
9 switch ($action) {
10     case 'show':
11         $word = filter_input(INPUT_POST, 'word', FILTER_SANITIZE_STRING);
12         $sbm = filter_input(INPUT_POST, 'validate');
13         if (isset($sbm)) {
14             materiels::getMaterielBySearch("%".$word."%");
15         }
16         break;
17 }
```

c:\tpi\Matos\function\calendrier.php

```
1 <!--
2 Auteur: David Machado
3 Date: 18.05.2022
4 Projet: Matos
5 -!>
6 <?php
7 class Calendar
8 {
9
10    /**
11     * Constructor
12     */
13    public function __construct()
14    {
15        $this->navigateUrl = htmlentities($_SERVER['PHP_SELF']);
16    }
17
18    /***** PROPERTY *****/
19    private $dayLabels = array("Lun", "Mar", "Mer", "Jeu", "Ven", "Sam", "Dim");
20
21    private $currentYear = 0;
22
23    private $currentMonth = 0;
24
25    private $currentDay = 0;
26
27    private $currentDate = null;
28
29    private $daysInMonth = 0;
30
31    private $navigateUrl = null;
32
33    /***** PUBLIC *****/
34
35    /**
36     * print out the calendar
37     */
38    public function show()
39    {
40        $year = null;
41
42        $month = null;
43
44        if (null == $year && isset($_GET['year'])) {
45
46            $year = $_GET['year'];
47        } else if (null == $year) {
48
49            $year = date("Y", time());
50        }
51
52        if (null == $month && isset($_GET['month'])) {
53
54            $month = $_GET['month'];
55        } else if (null == $month) {
56
57            $month = date("m", time());
58        }
59    }
}
```

```

60     $this->currentYear = $year;
61
62     $this->currentMonth = $month;
63
64     $this->daysInMonth = $this->_daysInMonth($month, $year);
65
66     $content = '<div id="calendar">' .
67         '<div class="box">' .
68         $this->_createNavi() .
69         '</div>' .
70         '<div class="box-content">' .
71         '<ul class="label">' . $this->_createLabels() . '</ul>';
72     $content .= '<div class="clear"></div>';
73     $content .= '<ul class="dates">';
74
75     $weeksInMonth = $this->_weeksInMonth($month, $year);
76     // Create weeks in a month
77     for ($i = 0; $i < $weeksInMonth; $i++) {
78
79         //Create days in a week
80         for ($j = 1; $j <= 7; $j++) {
81             $content .= $this->_showDay($i * 7 + $j);
82         }
83     }
84
85     $content .= '</ul>';
86
87     $content .= '<div class="clear"></div>';
88
89     $content .= '</div>';
90
91     $content .= '</div>';
92     return $content;
93 }
94
95 /***** PRIVATE *****/
96 /**
97  * create the li element for ul
98 */
99 private function _showDay($cellNumber)
100 {
101
102     if ($this->currentDay == 0) {
103
104         $firstDayOfTheWeek = date('N', strtotime($this->currentYear . '-' . $this-
105 >currentMonth . '-01'));
106
107         if (intval($cellNumber) == intval($firstDayOfTheWeek)) {
108
109             $this->currentDay = 1;
110         }
111
112     if (($this->currentDay != 0) && ($this->currentDay <= $this->daysInMonth)) {
113
114         $this->currentDate = date('Y-m-d', strtotime($this->currentYear . '-' . $this-
115 >currentMonth . '-' . ($this->currentDay)));
116
117         $cellContent = $this->currentDay;
118
119         $this->currentDay++;
120     } else {

```

```

120
121     $this->currentDate = null;
122
123     $cellContent = null;
124 }
125
126     //récupérer les dates non disponibles
127     $state = "<a style='color: #08FF00;' href='index.php?uc=pret&action=validate&date=$this->currentDate&idMateriel=" . $_GET['idMateriel'] . "'>" .
128     $cellContent . "</a>";
129     $lockedDates = prets::getUnavailableDatesByPlate($_GET['idMateriel']);
130     foreach ($lockedDates as $lockedDate) {
131         if (date('Y-m-d', strtotime($lockedDate->getDateDebut())) == $this-
132 >currentDate) {
133             $state = "<p style='font-size: 80%;background-color: #ff0000'>Réservé</p>";
134         }
135         if (date('Y-m-d', strtotime($lockedDate->getDateFin())) == $this->currentDate)
136         {
137             $state = "<p style='font-size: 80%;background-color: #ff0000'>Réservé</p>";
138         }
139         if (date('Y-m-d') > $this->currentDate) {
140             $state = "";
141         }
142
143         return '<li id="li-' . $this->currentDate . '" class="" . ($cellNumber % 7 == 1 ? '
144 start ' : ($cellNumber % 7 == 0 ? ' end ' : ' ')) .
145             ($cellContent == null ? 'mask' : '') . '">' . $state . '</li>';
146     }
147 /**
148 * create navigation
149 */
150 private function _createNavi()
151 {
152
153     $nextMonth = $this->currentMonth == 12 ? 1 : intval($this->currentMonth) + 1;
154
155     $nextYear = $this->currentMonth == 12 ? intval($this->currentYear) + 1 : $this-
156 >currentYear;
157
158     $preMonth = $this->currentMonth == 1 ? 12 : intval($this->currentMonth) - 1;
159
160     $preYear = $this->currentMonth == 1 ? intval($this->currentYear) - 1 : $this-
161 >currentYear;
162
163     return
164         '<div class="header">' .
165             '<a class="prev" href="index.php?uc=calendriers&idMateriel=' .
166             $_GET['idMateriel'] . '&month=' . sprintf('%02d', $preMonth) . '&year=' . $preYear .
167             '' . "&#11013;&#65039;</a>' .
168             '<span class="title">' . date('m / Y', strtotime($this->currentYear . '-' .
169             $this->currentMonth . '-1')) . '</span>' .
170             '<a class="next" href="index.php?uc=calendriers&idMateriel=' .
171             $_GET['idMateriel'] . '&month=' . sprintf("%02d", $nextMonth) . '&year=' . $nextYear .
172             '' . "&#10145;&#65039;</a>' .
173             '</div>';
174     }
175 /**
176 * create calendar week labels

```

```

171     */
172     private function _createLabels()
173     {
174
175         $content = '';
176
177         foreach ($this->dayLabels as $index => $label) {
178
179             $content .= '<li class="' . ($label == 6 ? 'end title' : 'start title') . ' title">' . $label . '</li>';
180         }
181
182         return $content;
183     }
184
185
186
187     /**
188      * calculate number of weeks in a particular month
189      */
190     private function _weeksInMonth($month = null, $year = null)
191     {
192
193         if (null == ($year)) {
194             $year = date("Y", time());
195         }
196
197         if (null == ($month)) {
198             $month = date("m", time());
199         }
200
201         // find number of days in this month
202         $daysInMonths = $this->_daysInMonth($month, $year);
203
204         $numOfweeks = ($daysInMonths % 7 == 0 ? 0 : 1) + intval($daysInMonths / 7);
205
206         $monthEndingDay = date('N', strtotime($year . '-' . $month . '-' . $daysInMonths));
207
208         $monthStartDay = date('N', strtotime($year . '-' . $month . '-01'));
209
210         if ($monthEndingDay < $monthStartDay) {
211
212             $numOfweeks++;
213         }
214
215         return $numOfweeks;
216     }
217
218     /**
219      * calculate number of days in a particular month
220      */
221     private function _daysInMonth($month = null, $year = null)
222     {
223
224         if (null == ($year))
225             $year = date("Y", time());
226
227         if (null == ($month))
228             $month = date("m", time());
229
230         return date('t', strtotime($year . '-' . $month . '-01'));

```

231 | }
232 | }

c:\tpi\Matos\models\images.php

```
1 <!--
2 Auteur: David Machado
3 Date: 18.05.2022
4 Projet: Matos
5 -!>
6 <?php
7 class images
8 {
9     private $idImage;
10    private $nomImage;
11    private $lienImage;
12    private $idMateriel;
13
14 /**
15 * Get the value of idImage
16 */
17 public function getIdImage()
18 {
19     return $this->idImage;
20 }
21
22 /**
23 * Set the value of idImage
24 *
25 * @return self
26 */
27 public function setIdImage($idImage)
28 {
29     $this->idImage = $idImage;
30
31     return $this;
32 }
33
34 /**
35 * Get the value of nomImage
36 */
37 public function getNomImage()
38 {
39     return $this->nomImage;
40 }
41
42 /**
43 * Set the value of nomImage
44 *
45 * @return self
46 */
47 public function setNomImage($nomImage)
48 {
49     $this->nomImage = $nomImage;
50
51     return $this;
52 }
53
54 /**
55 * Get the value of lienImage
56 */
57 public function getLienImage()
58 {
59     return $this->lienImage;
```

```

60 }
61 /**
62 * Set the value of lienImage
63 *
64 * @return self
65 */
66 public function setLienImage($lienImage)
67 {
68     $this->lienImage = $lienImage;
69
70     return $this;
71 }
72
73 /**
74 * Get the value of idMateriel
75 */
76 public function getIdMateriel()
77 {
78     return $this->idMateriel;
79 }
80
81 /**
82 * Set the value of idMateriel
83 *
84 * @return self
85 */
86 public function setIdMateriel($idMateriel)
87 {
88     $this->idMateriel = $idMateriel;
89
90     return $this;
91 }
92
93 //ajout d'une image dans la bases de données
94 public static function AddImage(images $images)
95 {
96     $nomImage = $images->getNomImage();
97     $lienImage = $images->getLienImage();
98     $idMateriel = $images->getIdMateriel();
99
100    $sql = MonPdo::getInstance()->prepare('INSERT INTO images(nomImage, lienImage,
101                                            idMateriel) VALUES(:nom,:lien,:id)');
102    $sql->setFetchMode(PDO::FETCH_CLASS | PDO::FETCH_PROPS_LATE, 'images');
103    $sql->bindParam(':nom',$nomImage);
104    $sql->bindParam(':lien',$lienImage);
105    $sql->bindParam(':id',$idMateriel);
106    $sql->execute();//executer la requette
107 }
108 public static function getImageById($idMateriel){
109     $res=MonPdo::getInstance()->prepare('SELECT nomImage FROM images as i, materiels as
m WHERE i.idMateriel = m.idMateriel AND m.idMateriel = :id');
110     $res->setFetchMode(PDO::FETCH_CLASS|PDO::FETCH_PROPS_LATE, 'images');
111     $res->bindParam(':id', $idMateriel);
112     $res->execute();
113     $result=$res->fetch();
114
115     return $result;
116 }

```

c:\tpi\Matos\models\materiels.php

```
1 <!--
2 Auteur: David Machado
3 Date: 18.05.2022
4 Projet: Matos
5 -!>
6 <?php
7 class materiels
8 {
9     private $idMateriel;
10    private $marque;
11    private $description;
12    private $dateEntreeStock;
13    private $actif;
14    private $categorie;
15    private $isDelete;
16
17 /**
18 * Get the value of idMateriel
19 */
20 public function getIdMateriel()
21 {
22     return $this->idMateriel;
23 }
24
25 /**
26 * Set the value of idMateriel
27 *
28 * @return self
29 */
30 public function setIdMateriel($idMateriel)
31 {
32     $this->idMateriel = $idMateriel;
33
34     return $this;
35 }
36
37 /**
38 * Get the value of marque
39 */
40 public function getMarque()
41 {
42     return $this->marque;
43 }
44
45 /**
46 * Set the value of marque
47 *
48 * @return self
49 */
50 public function setMarque($marque)
51 {
52     $this->marque = $marque;
53
54     return $this;
55 }
56
57 /**
58 * Get the value of description
59 */
```

```
60     public function getDescription()
61     {
62         return $this->description;
63     }
64
65     /**
66      * Set the value of description
67      *
68      * @return self
69      */
70     public function setDescription($description)
71     {
72         $this->description = $description;
73
74         return $this;
75     }
76
77     /**
78      * Get the value of dateEntreeStock
79      */
80     public function getDateEntreeStock()
81     {
82         return $this->dateEntreeStock;
83     }
84
85     /**
86      * Set the value of dateEntreeStock
87      *
88      * @return self
89      */
90     public function setDateEntreeStock($dateEntreeStock)
91     {
92         $this->dateEntreeStock = $dateEntreeStock;
93
94         return $this;
95     }
96
97     /**
98      * Get the value of actif
99      */
100    public function getActif()
101    {
102        return $this->actif;
103    }
104
105    /**
106     * Set the value of actif
107     *
108     * @return self
109     */
110    public function setActif($actif)
111    {
112        $this->actif = $actif;
113
114        return $this;
115    }
116
117    /**
118     * Get the value of categorie
119     */
120    public function getCategorie()
```

```

121     {
122         return $this->categorie;
123     }
124
125     /**
126      * Set the value of categorie
127      *
128      * @return self
129      */
130     public function setCategorie($categorie)
131     {
132         $this->categorie = $categorie;
133
134         return $this;
135     }
136
137     /**
138      * Get the value of isDelete
139      */
140     public function getIsDelete()
141     {
142         return $this->isDelete;
143     }
144
145     /**
146      * Set the value of isDelete
147      *
148      * @return self
149      */
150     public function setIsDelete($isDelete)
151     {
152         $this->isDelete = $isDelete;
153
154         return $this;
155     }
156
157
158     //fonction qui permet de selection tout le matériel disponible
159     public static function GetAllMateriel()
160     {
161         $req = MonPdo::getInstance()->prepare("SELECT
m.idMateriel,marque,description,nomImage FROM materiels as m ,images as i WHERE
i.idMateriel = m.idMateriel AND actif = 1 AND isDelete = 1");
162         $req->setFetchMode(PDO::FETCH_CLASS | PDO::FETCH_PROPS_LATE, 'materiels');
163         $req->execute();
164         $returnSQL = $req->fetchAll();
165     ?>
166
167         <div class="container">
168             <div class="row" style="flex-wrap: wrap; justify-content: center;">
169                 <?php
170                 foreach ($returnSQL as $materiel) {
171                     ?>
172                     <div class="card" style="width: 18rem;">
173                         
174                         <div class="card-body">
175                             <h5><?= $materiel->getMarque() ?></h5>
176                             <p class="card-text"><?= $materiel->getDescription() ?></p>
177                             <a class="btn btn-primary" href="index.php?
uc=materiel&action=info&idMateriel=<?= $materiel->getIdMateriel() ?>">plus info</a>
178                         </div>

```

```

179             </div>
180         <?php
181     }
182     ?>
183     </div>
184   </div>
185 <?php
186 }
187
188 //fonction qui permet d'ajouter du matériel
189 public static function AddMaterial(materiels $materiels)
190 {
191     $actif = 1;
192     $categorie = $materiels->getCategorie();
193     $dateEntreeStock = date("Y-m-d H:i:s");
194     $description = $materiels->getDescription();
195     $marque = $materiels->getMarque();
196     $isDelete = 1;
197
198     $sql = MonPdo::getInstance()->prepare('INSERT INTO materiels(actif, categorie,
dateEntreeStock, description, marque, isDelete) VALUES(:Actif, :Categorie,
:DateEntreeStock, :Description, :Marque, :IsDelete)');
199     $sql->setFetchMode(PDO::FETCH_CLASS | PDO::FETCH_PROPS_LATE, 'materiels');
200     $sql->bindParam(':Actif', $actif);
201     $sql->bindParam(':Categorie', $categorie);
202     $sql->bindParam(':DateEntreeStock', $dateEntreeStock);
203     $sql->bindParam(':Description', $description);
204     $sql->bindParam(':Marque', $marque);
205     $sql->bindParam(':IsDelete', $isDelete);
206     $sql->execute();
207
208     return MonPdo::getInstance()->lastInsertId();
209 }
210
211 //fonction qui delete le matériel(en invisible pour les utilisateurs)
212 public static function SetDelete($idMateriel)
213 {
214     $req = MonPdo::getInstance()->prepare('UPDATE materiels SET isDelete = 0 WHERE
idMateriel = :IdMateriel');
215     $req->setFetchMode(PDO::FETCH_CLASS | PDO::FETCH_PROPS_LATE, 'materiels');
216     $req->bindParam(':IdMateriel', $idMateriel);
217     $req->execute();
218 }
219
220 //fonction qui permet mettre disponible ou pas
221 public static function setAvailability($idMateriel)
222 {
223     $req = MonPdo::getInstance()->prepare('UPDATE materiels SET actif = 0 WHERE
idMateriel = :IdMateriel');
224     $req->setFetchMode(PDO::FETCH_CLASS | PDO::FETCH_PROPS_LATE, 'materiels');
225     $req->bindParam(':IdMateriel', $idMateriel);
226     $req->execute();
227 }
228 public static function GetAllCategorie(): array
229 {
230     $req = MonPdo::getInstance()->prepare('SELECT DISTINCT categorie, COUNT(CASE actif
WHEN 0 THEN 0 ELSE NULL END) Disponible FROM materiels WHERE isDelete = 0 GROUP BY
idMateriel');
231     $req->setFetchMode(PDO::FETCH_CLASS | PDO::FETCH_PROPS_LATE, 'materiels');
232     $req->execute();
233     $returnSQL = $req->fetchAll();
234     return $returnSQL;

```

```

235     }
236
237     public static function getMaterielById($idMateriel)
238     {
239         $res = MonPdo::getInstance()->prepare('SELECT m.idMateriel,
marque,description,nomImage FROM materiels as m, images as i WHERE i.idMateriel =
m.idMateriel AND m.idMateriel = :id');
240         $res->setFetchMode(PDO::FETCH_CLASS | PDO::FETCH_PROPS_LATE, 'materiels');
241         $res->bindParam(':id', $idMateriel);
242         $res->execute();
243         $result = $res->fetch();
244
245         return $result;
246     }
247     public static function getMaterielBySearch($word)
248     {
249         $sql = MonPdo::getInstance()->prepare("SELECT
m.idMateriel,marque,description,nomImage FROM materiels as m, images as i WHERE
m.idMateriel = i.idMateriel and m.marque LIKE :marque");
250         $sql->setFetchMode(PDO::FETCH_CLASS | PDO::FETCH_PROPS_LATE, 'materiels');
251         $sql->bindParam(':marque', $word);
252         $sql->execute();
253
254         $returnSQL = $sql->fetchAll();
255     ?>
256
257     <div class="container">
258         <div class="row" style="flex-wrap: wrap; justify-content: center;">
259             <?php
260             foreach ($returnSQL as $materiel) {
261                 ?>
262                     <div class="card" style="width: 18rem;">
263                         
264                         <div class="card-body">
265                             <h5><?= $materiel->getMarque() ?></h5>
266                             <p class="card-text"><?= $materiel->getDescription() ?></p>
267                             <a class="btn btn-primary" href="index.php?
uc=materiel&action=info&idMateriel=<?= $materiel->getIdMateriel() ?>">plus info</a>
268                         </div>
269                     </div>
270                 <?php
271                 }
272                 ?>
273             </div>
274         </div>
275     <?php
276     }
277
278     public static function getMaterielByCategorie($selected)
279     {
280         $res = MonPdo::getInstance()->prepare('SELECT
m.idMateriel,marque,description,nomImage FROM materiels as m, images as i WHERE m.categorie =
:categorie and m.idMateriel = i.idMateriel');
281         $res->setFetchMode(PDO::FETCH_CLASS | PDO::FETCH_PROPS_LATE, 'materiels');
282         $res->bindParam(':categorie', $selected);
283         $res->execute();
284         $result = $res->fetchAll();
285
286     ?>
287
288     <div class="container">

```

```
289         <div class="row" style="flex-wrap: wrap; justify-content: center;">
290             <?php
291             foreach ($result as $materiel) {
292                 ?>
293                     <div class="card" style="width: 18rem;">
294                         
295                         <div class="card-body">
296                             <h5><?= $materiel->getMarque() ?></h5>
297                             <p class="card-text"><?= $materiel->getDescription() ?></p>
298                             <a class="btn btn-primary" href="index.php?
uc=materiel&action=info&idMateriel=<?= $materiel->getIdMateriel() ?>">plus info</a>
299                             </div>
300                         </div>
301                     <?php
302                     ?
303                     ?>
304                     </div>
305                 </div>
306             <?php
307         }
308     }
```

c:\tpi\Matos\models\MonPdo.php

```
1 <!--
2 Auteur: David Machado
3 Date: 18.05.2022
4 Projet: Matos
5 -!>
6 <?php
7 /**
8 * Classe d'accès aux données Utilise les services de la classe PDO
9 * Les attributs sont tous statiques, les 4 premiers pour la connexion
10 * $monPdo qui contiendra l'unique instance de la classe
11 */
12 class MonPdo
13 {
14     private static $serveur = 'mysql:host=127.0.0.1';
15     private static $bdd = 'dbname=MatosV2';
16     private static $user = 'adminMatos';
17     private static $mdp = 'Supermatos';
18     private static $monPdo;
19     private static $unPdo = null;
20
21     // Constructeur privé, crée l'instance de PDO qui sera sollicitée
22     // pour toutes les méthodes de la classe
23     private function __construct()
24     {
25         MonPdo::$unPdo = new PDO(MonPdo::$serveur.';'.MonPdo::$bdd, MonPdo::$user,
26         MonPdo::$mdp);
27         MonPdo::$unPdo->query("SET CHARACTER SET utf8");
28         MonPdo::$unPdo->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
29     }
30     public function __destruct()
31     {
32         MonPdo::$unPdo = null;
33     }
34     /**
35      * Fonction statique qui cree l'unique instance de la classe
36      * Appel : $instanceMonPdo = MonPdo::getInstance();
37      * @return l'unique objet de la classe MonPdo
38     */
39     public static function getInstance()
40     {
41         if(self::$unPdo == null)
42         {
43             self::$monPdo= new MonPdo();
44         }
45         return self::$unPdo;
46     }
47 }
48 ?>
```

c:\tpi\Matos\models\prets.php

```
1 <!--
2 Auteur: David Machado
3 Date: 18.05.2022
4 Projet: Matos
5 -!>
6 <?php
7 class prets
8 {
9     private $idPret;
10    private $dateReservation;
11    private $dateDebut;
12    private $dateFin;
13    private $idUtilisateur;
14    private $idMateriel;
15    private $validation;
16
17
18 /**
19 * Get the value of idPret
20 */
21 public function getIdPret()
22 {
23     return $this->idPret;
24 }
25
26 /**
27 * Set the value of idPret
28 *
29 * @return self
30 */
31 public function setIdPret($idPret)
32 {
33     $this->idPret = $idPret;
34
35     return $this;
36 }
37
38 /**
39 * Get the value of dateReservation
40 */
41 public function getDateReservation()
42 {
43     return $this->dateReservation;
44 }
45
46 /**
47 * Set the value of dateReservation
48 *
49 * @return self
50 */
51 public function setDateReservation($dateReservation)
52 {
53     $this->dateReservation = $dateReservation;
54
55     return $this;
56 }
57
58 /**
59 * Get the value of dateDebut
```

```
60     */
61     public function getDateDebut()
62     {
63         return $this->dateDebut;
64     }
65
66 /**
67 * Set the value of dateDebut
68 *
69 * @return self
70 */
71 public function setDateDebut($dateDebut)
72 {
73     $this->dateDebut = $dateDebut;
74
75     return $this;
76 }
77
78 /**
79 * Get the value of dateFin
80 */
81 public function getDateFin()
82 {
83     return $this->dateFin;
84 }
85
86 /**
87 * Set the value of dateFin
88 *
89 * @return self
90 */
91 public function setDateFin($dateFin)
92 {
93     $this->dateFin = $dateFin;
94
95     return $this;
96 }
97
98 /**
99 * Get the value of idUtilisateur
100 */
101 public function getIdUtilisateur()
102 {
103     return $this->idUtilisateur;
104 }
105
106 /**
107 * Set the value of idUtilisateur
108 *
109 * @return self
110 */
111 public function setIdUtilisateur($idUtilisateur)
112 {
113     $this->idUtilisateur = $idUtilisateur;
114
115     return $this;
116 }
117
118 /**
119 * Get the value of idMateriel
120 */
```

```

121 |     public function getIdMateriel()
122 |     {
123 |         return $this->idMateriel;
124 |     }
125 |
126 | /**
127 | * Set the value of idMateriel
128 | *
129 | * @return self
130 | */
131 | public function setIdMateriel($idMateriel)
132 | {
133 |     $this->idMateriel = $idMateriel;
134 |
135 |     return $this;
136 | }
137 |
138 | /**
139 | * Get the value of validation
140 | */
141 | public function getValidation()
142 | {
143 |     return $this->validation;
144 | }
145 |
146 | /**
147 | * Set the value of validation
148 | *
149 | * @return self
150 | */
151 | public function setValidation($validation)
152 | {
153 |     $this->validation = $validation;
154 |
155 |     return $this;
156 | }
157 | public static function setValidate($idPret)
158 | {
159 |     $req = MonPdo::getInstance()->prepare('UPDATE prets SET validation = 1 WHERE idPret
= :IdPret');
160 |     $req->setFetchMode(PDO::FETCH_CLASS | PDO::FETCH_PROPS_LATE, 'prets');
161 |     $req->bindParam(':IdPret', $idPret);
162 |     $req->execute();
163 | }
164 | //récup des date début et fin pour vérifier la disponibilité dans le calendriers
165 | public static function getUnavailableDatesByPlate($idMateriel)
166 | {
167 |     $req = MonPdo::getInstance()->prepare("SELECT dateDebut,dateFin FROM prets WHERE
idMateriel = :IDMateriel");
168 |     $req->setFetchMode(PDO::FETCH_CLASS | PDO::FETCH_PROPS_LATE, 'prets');
169 |     $req->bindParam(':IDMateriel', $idMateriel);
170 |     $req->execute();
171 |     $retourSQL = $req->fetchAll();
172 |     return $retourSQL;
173 | }
174 | //créer le prêt
175 | public static function addPret(prets $pret)
176 | {
177 |     echo "<pre>";
178 |     var_dump($pret);
179 |     echo "</pre>";
180 |     $dateReservation = $pret->getDateReservation();

```

```

181     $dateDebut = $pret->getDateDebut();
182     $dateFin = $pret->getDateFin();
183     $idUtilisateur = $pret->getIdUtilisateur();
184     $idMateriel = $pret->getIdMateriel();
185     $validation = 0;
186     $sql = MonPdo::getInstance()->prepare('INSERT INTO prets(dateReservation,
187 dateDebut, dateFin, idUtilisateur, idMateriel, validation) VALUES(:DateReservation,
188 :DateDebut, :DateFin, :IdUtilisateur, :IdMateriel, :Validation)');
189     $sql->setFetchMode(PDO::FETCH_CLASS | PDO::FETCH_PROPS_LATE, 'prets');
190     $sql->bindParam(':DateReservation', $dateReservation);
191     $sql->bindParam(':DateDebut', $dateDebut);
192     $sql->bindParam(':DateFin', $dateFin);
193     $sql->bindParam(':IdUtilisateur', $idUtilisateur);
194     $sql->bindParam(':IdMateriel', $idMateriel);
195     $sql->bindParam(':Validation', $validation);
196     $sql->execute();
197 }
198
199 //affichage de tous les prêts
200 public static function getAllLoan()
201 {
202     $sql = MonPdo::getInstance()->prepare("SELECT
203 idPret,dateReservation,dateDebut,dateFin,email,marque,validation FROM prets as p,materiels
204 as m, utilisateurs as u WHERE u.idUtilisateur = p.idUtilisateur and m.idMateriel =
205 p.idMateriel ORDER BY validation ASC");
206     $sql->setFetchMode(PDO::FETCH_CLASS | PDO::FETCH_PROPS_LATE, 'prets');
207     $sql->execute();
208
209     $result = $sql->fetchAll();
210     echo "<div class='table-responsive'>";
211     echo "<table class='table table-info table-responsive' >";
212     echo "<thead>";
213     echo "<tr>";
214     echo "<th scope='col'>Date de reservation</th>";
215     echo "<th scope='col'>Date de début</th>";
216     echo "<th scope='col'>Date de fin</th>";
217     echo "<th scope='col'>Utilisateur</th>";
218     echo "<th scope='col'>Matériel</th>";
219     echo "<th scope='col'>Statut</th>";
220     echo "<th scope='col'></th>";
221     echo "</tr>";
222     foreach ($result as $value) {
223         echo "<tr>";
224         echo "<td>" . $value->getDateReservation() . "</td>";
225         echo "<td>" . $value->getDateDebut() . "</td>";
226         echo "<td>" . $value->getDateFin() . "</td>";
227         echo "<td>" . $value->email . "</td>";
228         echo "<td>" . $value->marque . "</td>";
229         if ($value->getValidation() == 1) {
230             echo "<td>valider ✓</td>";
231             echo "<td></td>";
232         } else if ($value->getValidation() == 0) {
233             echo "<td>en attente 🕒</td>";
234             echo "<td style='text-align: center;'><a href='index.php?
235 uc=admin&action=accept&idPret=" . $value->getIdPret() . "' style='border: 1px solid
236 black;font-size: 100%;' class='btn btn-outline-success'>Valider</a></td>";
237         }
238         echo "</tr>";
239     }
240     echo "</thead>";
241     echo "</table>";
242     echo "</div>";

```

236 }
237 }

c:\tpi\Matos\models\utilisateurs.php

```
1 <!--
2 Auteur: David Machado
3 Date: 18.05.2022
4 Projet: Matos
5 -!>
6 <?php
7
8 use LDAP\Result;
9
10 class utilisateurs
11 {
12     private $idUtilisateur;
13     private $nom;
14     private $prenom;
15     private $noTel;
16     private $pseudo;
17     private $motDePasse;
18     private $email;
19     private $statut;
20     private $isDeleted;
21
22
23 /**
24 * Get the value of idUtilisateur
25 */
26 public function getIdUtilisateur()
27 {
28     return $this->idUtilisateur;
29 }
30
31 /**
32 * Set the value of idUtilisateur
33 *
34 * @return self
35 */
36 public function setIdUtilisateur($idUtilisateur)
37 {
38     $this->idUtilisateur = $idUtilisateur;
39
40     return $this;
41 }
42
43 /**
44 * Get the value of nom
45 */
46 public function getNom()
47 {
48     return $this->nom;
49 }
50
51 /**
52 * Set the value of nom
53 *
54 * @return self
55 */
56 public function setNom($nom)
57 {
58     $this->nom = $nom;
59 }
```

```
60         return $this;
61     }
62
63     /**
64      * Get the value of prenom
65      */
66     public function getPrenom()
67     {
68         return $this->prenom;
69     }
70
71     /**
72      * Set the value of prenom
73      *
74      * @return self
75      */
76     public function setPrenom($prenom)
77     {
78         $this->prenom = $prenom;
79
80         return $this;
81     }
82
83     /**
84      * Get the value of noTel
85      */
86     public function getNoTel()
87     {
88         return $this->noTel;
89     }
90
91     /**
92      * Set the value of noTel
93      *
94      * @return self
95      */
96     public function setNoTel($noTel)
97     {
98         $this->noTel = $noTel;
99
100        return $this;
101    }
102
103    /**
104     * Get the value of pseudo
105     */
106    public function getPseudo()
107    {
108        return $this->pseudo;
109    }
110
111    /**
112     * Set the value of pseudo
113     *
114     * @return self
115     */
116    public function setPseudo($pseudo)
117    {
118        $this->pseudo = $pseudo;
119
120        return $this;
```

```
121     }
122 
123     /**
124      * Get the value of motDePasse
125      */
126     public function getMotDePasse()
127     {
128         return $this->motDePasse;
129     }
130 
131     /**
132      * Set the value of motDePasse
133      *
134      * @return self
135      */
136     public function setMotDePasse($motDePasse)
137     {
138         $this->motDePasse = $motDePasse;
139 
140         return $this;
141     }
142 
143     /**
144      * Get the value of email
145      */
146     public function getEmail()
147     {
148         return $this->email;
149     }
150 
151     /**
152      * Set the value of email
153      *
154      * @return self
155      */
156     public function setEmail($email)
157     {
158         $this->email = $email;
159 
160         return $this;
161     }
162 
163     /**
164      * Get the value of statut
165      */
166     public function getStatut()
167     {
168         return $this->statut;
169     }
170 
171     /**
172      * Set the value of statut
173      *
174      * @return self
175      */
176     public function setStatut($statut)
177     {
178         $this->statut = $statut;
179 
180         return $this;
181     }
```

```

182 /**
183 * Get the value of isDeleted
184 */
185 public function getIsDeleted()
186 {
187     return $this->isDeleted;
188 }
189
190 /**
191 * Set the value of isDeleted
192 *
193 * @return self
194 */
195 public function setIsDeleted($isDeleted)
196 {
197     $this->isDeleted = $isDeleted;
198
199     return $this;
200 }
201 //crypter le mot de passe
202 public static function Crypter($mdpClair)
203 {
204     return md5($mdpClair);
205 }
206
207 // verifier les informations de connection
208 public static function CheckConnected(utilisateurs $user)
209 {
210     $email = $user->getEMail();
211     $req = MonPdo::getInstance()->prepare("SELECT * FROM utilisateurs WHERE email =
212 :email;");
213     $req->setFetchMode(PDO::FETCH_CLASS | PDO::FETCH_PROPS_LATE, 'utilisateurs'); //214 mettre le nom de la classe
214     $req->bindParam(":email", $email);
215     $req->execute();
216     $result = $req->fetch();
217     return $result;
218 }
219
220 //ajouter un utilisateur
221 public static function AddUser(utilisateurs $user)
222 {
223     $nom = $user->getNom();
224     $prenom = $user->getPrenom();
225     $noTel = $user->getNoTel();
226     $pseudo = $user->getPseudo();
227     $motDePasse = utilisateurs::Crypter($user->getMotDePasse());
228     $email = $user->getEmail();
229     $statut = 1;
230     $isDeleted = 0;
231     $req = MonPdo::getInstance()->prepare("INSERT INTO
232 utilisateurs(nom,prenom,noTel,pseudo,motDePasse,email,statut,isDeleted) VALUES
233 (:Nom,:Prenom,:NoTel,:Pseudo,:MotDePasse,:Email,:Statut,:IsDeleted)");
234     $req->setFetchMode(PDO::FETCH_CLASS | PDO::FETCH_PROPS_LATE, 'utilisateurs');
235     $req->bindParam(':Nom', $nom);
236     $req->bindParam(':Prenom', $prenom);
237     $req->bindParam(':NoTel', $noTel);
238     $req->bindParam(':Pseudo', $pseudo);
239     $req->bindParam(':MotDePasse', $motDePasse);
240     $req->bindParam(':Email', $email);
241     $req->bindParam(':Statut', $statut);
242     $req->bindParam(':IsDeleted', $isDeleted);

```

```

240     $req->execute();
241 }
242 public static function Update(utilisateurs $user)
243 {
244     $pseudo = $user->getPseudo();
245     $noTel = $user->getNoTel();
246     $motDePasse = utilisateurs::Crypter($user->getMotDePasse());
247     $idUtilisateur = $user->getIdUtilisateur();
248     $sql = MonPdo::getInstance()->prepare("UPDATE utilisateurs SET pseudo = :pseudo,
249 noTel = :noTel, motDePasse = :mdp where idUtilisateur = :id");
250     $sql->setFetchMode(PDO::FETCH_CLASS | PDO::FETCH_PROPS_LATE, 'utilisateurs');
251     $sql->bindParam(":pseudo", $pseudo);
252     $sql->bindParam(":noTel", $noTel);
253     $sql->bindParam(":mdp", $motDePasse);
254     $sql->bindParam(":id", $idUtilisateur);
255     $sql->execute();
256 }
257 public static function IsEmailExisting($email, $pseudo)
258 {
259     $req = MonPdo::getInstance()->prepare("SELECT idUtilisateur, email,pseudo FROM
260 utilisateurs");
261     $req->setFetchMode(PDO::FETCH_CLASS | PDO::FETCH_PROPS_LATE, 'utilisateurs');
262     $req->execute();
263     $result = $req->fetchAll();
264
265     foreach ($result as $r) {
266         if ($r->getEmail() == $email || $r->getPseudo() == $pseudo) {
267             return true;
268         }
269     }
270     return false;
271 }
272 //Affichage de tous les utilisateurs
273 public static function getAllUser()
274 {
275     $res = MonPdo::getInstance()->prepare("SELECT * FROM utilisateurs WHERE isDeleted =
276 0 ORDER BY statut DESC");
277     $res->setFetchMode(PDO::FETCH_CLASS | PDO::FETCH_PROPS_LATE, 'utilisateurs');
278     $res->execute();
279
280     $result = $res->fetchAll();
281     echo "<div class='table-responsive'>";
282     echo "<table class='table table-info table-responsive' >";
283     echo "<thead>";
284     echo "<tr>";
285     echo "<th scope='col'>E-mail</th>";
286     echo "<th scope='col'>Pseudo</th>";
287     echo "<th scope='col'>Numéro de téléphone</th>";
288     echo "<th scope='col'>Statut</th>";
289     echo "</th></tr>";
290     foreach ($result as $value) {
291         echo "<tr>";
292         echo "<td>" . $value->getEmail() . "</td>";
293         echo "<td>" . $value->getPseudo() . "</td>";
294         echo "<td>" . $value->getNoTel() . "</td>";
295         if ($value->getStatut() == 1) {
296             echo "<td><img alt='User icon' data-bbox='350 880 370 900' data-kind='parent' data-rs='2' data-size='1em' data-style='vertical-align: middle; height: 1em; width: 1em;'/></td>";
297         } else if ($value->getStatut() == 2) {
298             echo "<td><img alt='Star icon' data-bbox='350 910 370 930' data-kind='parent' data-rs='2' data-size='1em' data-style='vertical-align: middle; height: 1em; width: 1em;'/></td>";
299         }
300     }
301 }

```

```
299         echo "<td style='text-align: center;'><a href='index.php?
300 uc=admin&action=delete&idUtilisateur=" . $value->getIdUtilisateur() . "' style='border: 1px
301 solid black;font-size: 100%;' class='btn btn-outline-danger'>Supprimer</a></td>";
302     echo "</tr>";
303 }
304 echo "</thead>";
305 echo "</table>";
306 }
```

c:\tpi\Matos\vues\accueil.php

```
1 <!--
2 Auteur: David Machado
3 Date: 18.05.2022
4 Projet: Matos      -->
5
6 <div class="container">
7     <div class="row" style="flex-wrap: wrap; justify-content: center;">
8         <div class="card" style="width: 18rem;">
9             
10            <div class="card-body">
11                <h5>macbook pro 15pouce</h5>
12                <p class="card-text">ordinateur portable très puissant</p>
13                <a class="btn btn-primary">Louer</a>
14            </div>
15        </div>
16    </div>
17 </div>
```

c:\tpi\Matos\vues\ajoutMateriel.php

```
1 <!--
2 Auteur: David Machado
3 Date: 18.05.2022
4 Projet: Matos      -->
5
6 <!-- la form d'ajout -->
7 <div class="container">
8     <?php
9         if ($_SESSION['alertMessage']['type'] != null) {
10             ?>
11                 <div class="alert alert-<?= $_SESSION['alertMessage']['type'] ?> alert-dismissible fade show" role="alert">
12                     <?= $_SESSION['alertMessage']['message'] ?>
13                         <button type="button" class="btn-close" data-bs-dismiss="alert" aria-label="Close"></button>
14                 </div>
15             <?php
16                 $_SESSION['alertMessage'] = [
17                     "type" => null,
18                     "message" => null
19                 ];
20             }
21             ?>
22             <div class="row">
23                 <div class=col-lg-3></div>
24                 <div class=col-lg-6>
25                     <div id="ui">
26                         <h1 class="text-center">ajout de matériel</h1>
27                         <form method="POST" action="index.php?uc=materiel&action=validateAdd"
28                             class="form-group text-center" enctype="multipart/form-data">
29                             <div class="row">
30                                 <div class="col-lg-6">
31                                     <label>marque :</label>
32                                     <input type="text" name="marque" placeholder="Enter your First
33                                         name .." class="form-control">
34                                 </div>
35                                 <div class="col-lg-4">
36                                     <label>catégorie :</label>
37                                     <select name="categorie">
38                                         <option value="ordinateur portables">ordinateur
39                                         portables</option>
40                                         <option value="réseau">réseau</option>
41                                         <option value="connectiques">connectique</option>
42                                         <option value="téléphonie">téléphonie</option>
43                                         <option value="périphérique">périphérique</option>
44                                     </select>
45                                 </div>
46                             </div>
47                             <br>
48                             <label>images du matériel :</label>
49                             <input class="form-control" type="file" id="image" accept="image/*"
50                               name="image">
51                             <br>
52                             <label>description :</label>
53                             <input type="text" name="description" placeholder="description"
54                               class="form-control">
55                             <br>
56                             <input type="submit" name="valider" value="ajouter" class="btn btn-
57                               success btn-block btn-lg">
58                         </form>
59                     </div>
```

```
54     </div>
55     <div class=col-lg-3></div>
56   </div>
57 </div>
```

c:\tpi\Matos\vues\calendar.php

```
1 <!--
2 Auteur: David Machado
3 Date: 18.05.2022
4 Projet: Matos      -->
5
6 <?php
7 $calendar = new Calendar();
8
9 echo $calendar->show();
10 ?>
```

c:\tpi\Matos\vues\categorie.php

```
1 <!--
2 Auteur: David Machado
3 Date: 18.05.2022
4 Projet: Matos      -->
5
6 <div class="container mt-5">
7   <div class="row">
8
9     <div class="col-md-3">
10       <div class="card mb-4 ">
11         <div class="categorie-name" style="text-align: center;">Ordinateur
portable</div>
12         
13         <div class="card-body" style="margin: auto;">
14           <a class="btn btn-outline-info mt-4" id="showTable" href="index.php?
uc=categorie&action=liste&selected=ordinateur portables">
15             <h2>lister</h2>
16           </a>
17         </div>
18       </div>
19     </div>
20
21     <div class="col-md-3">
22       <div class="card mb-4 ">
23         <div class="categorie-name" style="text-align: center;">Réseau</div>
24         
25         <div class="card-body" style="margin: auto;">
26           <a class="btn btn-outline-info mt-4" id="showTable" href="index.php?
uc=categorie&action=liste&selected=réseau">
27             <h2>lister</h2>
28           </a>
29         </div>
30       </div>
31     </div>
32
33     <div class="col-md-3">
34       <div class="card mb-4 ">
35         <div class="categorie-name" style="text-align: center;">connectiques</div>
36         
37         <div class="card-body" style="margin: auto;">
38           <a class="btn btn-outline-info mt-4" id="showTable" href="index.php?
uc=categorie&action=liste&selected=connectiques">
39             <h2>lister</h2>
40           </a>
41         </div>
42       </div>
43     </div>
44
45     <div class="col-md-3">
46       <div class="card mb-4 ">
47         <div class="categorie-name" style="text-align: center;">Téléphonie</div>
48         
49         <div class="card-body" style="margin: auto;">
50           <a class="btn btn-outline-info mt-4" id="showTable" href="index.php?
uc=categorie&action=liste&selected=téléphonie">
51             <h2>lister</h2>
52           </a>
53         </div>
54       </div>
55     </div>
```

```
56
57     <div class="col-md-3">
58         <div class="card mb-4 ">
59             <div class="categorie-name" style="text-align: center;">Périphérique</div>
60             
61             <div class="card-body" style="margin: auto;">
62                 <a class="btn btn-outline-info mt-4" id="showTable" href="index.php?uc=categorie&action=liste&selected=périphérique">
63                     <h2>lister</h2>
64                 </a>
65             </div>
66         </div>
67     </div>
68
69     </div>
70 </div>
```

c:\tpi\Matos\vues\footer.php

```
1 <!--
2 Auteur: David Machado
3 Date: 18.05.2022
4 Projet: Matos      -->
5
6 <!-- Footer -->
7 <footer class="page-footer font-small blue pt-4">
8     <!-- Copyright -->
9     <div class="footer navbar-dark">© 2022 Copyright : David &
10        <a class="lienCfpt" href="https://edu.ge.ch/site/cfpt-informatique/">CFPT-I</a>
11    </div>
12    <!-- Copyright -->
13 </footer>
14 <!-- Footer -->
15 <script src="https://code.jquery.com/jquery-3.3.1.slim.min.js%22%3E"></script>
16 <script
17   src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.3/umd/popper.min.js%22%3E">
18 </script>
19 <script src="assets/bootstrap/bootstrap-5.1.3-dist/bootstrap-5.1.3-
20 dist/js/bootstrap.min.js"></script>
</body>
</html>
```

c:\tpi\Matos\vues\header.php

```
1 <!--
2 Auteur: David Machado
3 Date: 18.05.2022
4 Projet: Matos      -->
5
6 <!DOCTYPE html>
7 <html lang="fr">
8
9 <head>
10   <meta charset="UTF-8">
11   <meta name="viewport" content="width=device-width, initial-scale=1.0">
12   <!--<link rel="stylesheet"
13 href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css"
14 integrity="sha384-Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJ1SAwiGgFAW/dAiS6JXm"
15 crossorigin="anonymous">-->
16   <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css"
17 rel="stylesheet" integrity="sha384-BmE4kWBq78iYHFlvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3" crossorigin="anonymous">
18   <link rel="stylesheet" type="text/css" href="assets/css/style.css">
19   <title>Matos</title>
20 </head>
21
22 <body>
23   <nav class="navbar navbar-expand-lg navbar-light bg-light">
24     <div class="container-fluid">
25       &ampnbsp;
26       <a class="navbar-brand" href="index.php?uc=accueil&action=show">Matos</a>
27       <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-
28 target="#navbarScroll" aria-controls="navbarScroll" aria-expanded="false" aria-label="Toggle
29 navigation">
30         <span class="navbar-toggler-icon"></span>
31     </button>
32     <div class="collapse navbar-collapse" id="navbarScroll">
33       <ul class="navbar-nav me-auto my-2 my-lg-0 navbar-nav-scroll" style="--bs-scroll-
34 height: 100px;">
35         <li class="nav-item">
36           <a class="nav-link active" aria-current="page" href="index.php?
37 uc=accueil&action=show">Accueil</a>
38         </li>
39         <li class="nav-item">
40           <a class="nav-link" href="index.php?uc=categorie&action=show">Catégorie</a>
41         </li>
42         <form action="index.php?uc=search&action=show" method="POST">
43           <input class="nav-link" type="search" name="word" placeholder="marque">
44           <input class="nav-link" type="submit" name="validate">
45         </form>
46         <?php
47         if ($_SESSION['userConnected']['statut'] == 2) {
48           ?>
49             <li class="nav-item">
50               <a class="nav-link active" aria-current="page" href="index.php?
51 uc=materiel&action=ajout">ajout</a>
52             </li>
53             <li class="nav-item dropdown" style="list-style-type: none;">
54               <a class="nav-link dropdown-toggle" href="#" id="navbarScrollingDropdown"
55 role="button" data-bs-toggle="dropdown" aria-expanded="false">
56                 gérer
57               </a>
58               <ul class="dropdown-menu" aria-labelledby="navbarScrollingDropdown">
59                 <li><a class="dropdown-item" href="index.php?
60 uc=admin&action=listAllUser">list enseignant</a></li>
```

```
50         <li><a class="dropdown-item" href="index.php?
51 uc=admin&action=showAllPret">list prêt</a></li>
52     </ul>
53     </li>
54     <?php
55     }
56     ?>
57   </ul>
58   <form class="d-flex">
59     <?php
60     if ($_SESSION['userConnected']['online']) {
61     ?>
62       <li class="nav-item dropdown" style="list-style-type: none;">
63         <a class="nav-link dropdown-toggle" href="#" id="navbarScrollingDropdown"
64 role="button" data-bs-toggle="dropdown" aria-expanded="false">
65           </a>
66           <ul class="dropdown-menu" aria-labelledby="navbarScrollingDropdown">
67             <li><a class="dropdown-item" href="index.php?
68 uc=login&action=showProfil">profil</a></li>
69             <li><a class="dropdown-item" href="index.php?uc=login&action=deconnecter">Se
70 Déconnecter</a></li>
71           </ul>
72         </li>
73     <?php
74     } else {
75       echo '<a class="btn btn-success" href="index.php?uc=login&action=show">Login</a>
76 <a class="btn btn-success" href="index.php?uc=login&action=showRegister">Register</a>';
77     }
78     ?>
79   </form>
80   </div>
81 </div>
82 </nav>
```

c:\tpi\Matos\vues\login.php

```
1 <!--
2 Auteur: David Machado
3 Date: 18.05.2022
4 Projet: Matos      -->
5
6 <!-- la form du logins -->
7 <section class="container-fluid">
8     <?php
9         if ($_SESSION['alertMessage']['type'] != null) {
10             ?>
11                 <div class="alert alert-<?= $_SESSION['alertMessage']['type'] ?> alert-
dismissible fade show" role="alert">
12                     <?= $_SESSION['alertMessage']['message'] ?>
13                     <button type="button" class="btn-close" data-bs-dismiss="alert" aria-
label="Close"></button>
14                 </div>
15
16
17         <?php
18             $_SESSION['alertMessage'] = [
19                 "type" => null,
20                 "message" => null
21             ];
22         }
23         ?>
24     <section class="row justify-content-center">
25         <section class="col-12 col-sm-6 col-md-3">
26             <form method="POST" action="index.php?uc=login&action=validateLogin"
class="form-container ">
27                 <h1>Login</h1>
28                 <div class="form-group">
29                     <label for="mail">mail</label>
30                     <input type="text" name="email" class="form-control" id="mail" aria-
describedby="emailHelp">
31                     </div>
32                     <div class="form-group">
33                         <label for="password">Password</label>
34                         <input type="password" name="password" class="form-control"
id="password">
35                     </div>
36                     <a href="index.php?uc=login&action=showRegister">Nouveau ? Créer un
compte ici</a>
37                     <input type="submit" name="valider" class="btn btn-success btn-block"
value="Login">
38                 </form>
39             </section>
40         </section>
41     </section>
```

c:\tpi\Matos\vues\materiel.php

```
1 <!--
2 Auteur: David Machado
3 Date: 18.05.2022
4 Projet: Matos      -->
5
6 <div class="container">
7   <div class="row">
8     
9     <h1><?= $materiel->getMarque() ?></h1>
10    </div>
11    <h3><?= $materiel->getDescription() ?></h3>
12    <a class="btn btn-primary" href="index.php?uc=calendriers&idMateriel=<?= $materiel->getIdMateriel()?>">louer</a>
13    <?php
14      if ($_SESSION['userConnected']['statut'] == 2) {
15        ?
16        <a class="btn btn-danger" href="index.php?uc=materiel&action=delete&idMateriel=<?= $materiel->getIdMateriel()?>">supprimer</a>
17      <?php
18      }
19    ?
20 </div>
```

c:\tpi\Matos\vues\materielByCategorie.php

```
1 <!--
2 Auteur: David Machado
3 Date: 18.05.2022
4 Projet: Matos      -->
```

c:\tpi\Matos\vues\profil.php

```
1 <!--
2 Auteur: David Machado
3 Date: 18.05.2022
4 Projet: Matos      -->
5
6 <div class="container bootstrap snippets bootdey">
7     <h1 class="text-primary">Editer le Profil</h1>
8     <hr>
9
10    <div class="row">
11        <!-- left column -->
12        <div class="col-md-3">
13            <div class="text-center">
14                
15            </div>
16        </div>
17        <!-- edit form column -->
18        <div class="col-md-9 personal-info">
19            <h3>information personnelle</h3>
20            <form class="form-horizontal" role="form" method="POST" action="index.php?uc=login?
action=update">
21                <div class="form-group">
22                    <label class="col-lg-3 control-label">Téléphone</label>
23                    <div class="col-lg-8">
24                        <input class="form-control" type="text" name="notel" value=<?=
$_SESSION['userConnected']['noTel'] ?>">
25                    </div>
26                </div>
27                <div class="form-group">
28                    <label class="col-lg-3 control-label">Pseudo</label>
29                    <div class="col-lg-8">
30                        <input class="form-control" type="text" name="pseudo" value=<?=
$_SESSION['userConnected']['pseudo'] ?>">
31                    </div>
32                </div>
33                <div class="form-group">
34                    <label class="col-lg-3 control-label">Nouveau mot de passe</label>
35                    <div class="col-lg-8">
36                        <input class="form-control" type="password" name="mdp" value="">
37                    </div>
38                </div>
39                <br>
40                <div class="form-group">
41                    <div class="col-lg-8">
42                        <input class="btn btn-success btn-block btn-lg" type="submit" name="update"
value="update">
43                    </div>
44                </div>
45            </form>
46        </div>
47    </div>
48 </div>
49 <hr>
```

c:\tpi\Matos\vues\register.php

```
1 <!--
2 Auteur: David Machado
3 Date: 18.05.2022
4 Projet: Matos      -->
5
6 <!-- la form du register -->
7 <div class="container">
8 <?php
9
10 if ($_SESSION['alertMessage']['type'] != null) {
11 ?>
12     <div class="alert alert-<?= $_SESSION['alertMessage']['type'] ?> alert-dismissible fade
show" role="alert">
13         <?= $_SESSION['alertMessage']['message'] ?>
14         <button type="button" class="btn-close" data-bs-dismiss="alert" aria-
label="Close"></button>
15     </div>
16
17
18 <?php
19     $_SESSION['alertMessage'] = [
20         "type" => null,
21         "message" => null
22     ];
23 }
24 ?>
25     <div class="row">
26         <div class=col-lg-3></div>
27         <div class=col-lg-6>
28             <div id="ui">
29                 <h1 class="text-center">Register</h1>
30                 <form method="POST" action="index.php?uc=login&action=validateRegister"
class="form-group text-center">
31                     <div class="row" >
32                         <div class="col-lg-6">
33                             <label>First Name :</label>
34                             <input type="text" name="prenom" placeholder="Enter your
First name .."
35                                     class="form-control">
36                         </div>
37                         <div class="col-lg-6">
38                             <label>Last Name :</label>
39                             <input type="text" name="nom" placeholder="Enter your Last
name .." class="form-control">
40                         </div>
41                     </div>
42                     <br>
43                     <div class="row" >
44                         <div class="col-lg-6">
45                             <label>E-mail :</label>
46                             <input type="text" name="email" placeholder="Enter your
email .." class="form-control">
47                         </div>
48                         <div class="col-lg-6">
49                             <label>Pseudo :</label>
50                             <input type="text" name="pseudo" placeholder="Enter your
pseudo .." class="form-control">
51                         </div>
52                     </div>
53                     <br>
54                     <label>Numéro de téléphone :</label>
```

```
55         <input type="text" name="noTel" placeholder="enter your number .."  
56         class="form-control">  
57         <br>  
58         <div class="row">  
59             <div class="col-lg-12">  
60                 <label>Password :</label>  
61                 <input type="password" name="password" placeholder="Enter  
your password .." class="form-control">  
62                     </div>  
63                     <br>  
64                     <a href="index.php?uc=login&action=show">Déjà membre ? Connectez-  
vous ici</a>  
65                     <br>  
66                     <input type="submit" name="valider" value="Register" class="btn btn-  
success btn-block btn-lg">  
67                 </form>  
68             </div>  
69             </div>  
70         <div class=col-lg-3></div>  
71     </div>  
72 </div>
```

c:\tpi\Matos\vues\showUser.php

```
1 |<!--  
2 Auteur: David Machado  
3 Date: 18.05.2022  
4 Projet: Matos      -->
```

c:\tpi\Matos\assets\css\style.css

```
1 .footer {
2   position: fixed;
3   left: 0;
4   bottom: 0;
5   width: 100%;
6   background-color: #3e3f3a;
7   color: white;
8   text-align: center;
9 }
10
11 .lienCfpt:link {
12   color: red;
13 }
14
15 .lienCfpt:visited {
16   color: red;
17   background-color: transparent;
18   text-decoration: none;
19 }
20
21 .lienCfpt:hover {
22   color: red;
23   background-color: transparent;
24   text-decoration: none;
25 }
26
27 .lienCfpt:active {
28   color: red;
29   background-color: transparent;
30   text-decoration: underline;
31 }
32
33 .categorie-name {
34   margin: auto;
35   font-size: 200%;
36   margin-top: 10px;
37   margin-bottom: 10px;
38 }
39
40 .icon-img {
41   margin: auto;
42   width: 180px;
43   height: 180px;
44   border-radius: 10%;
45 }
46
47 div#calendar {
48   margin: 0px auto;
49   padding: 0px;
50   width: 602px;
51   font-family: Helvetica, "Times New Roman", Times, serif;
52 }
53
54 div#calendar div.box {
55   position: relative;
56   top: 0px;
57   left: 0px;
58   width: 100%;
59   height: 40px;
```

```
60 background-color: #787878;
61 }
62
63 div#calendar div.header {
64   line-height: 40px;
65   vertical-align: middle;
66   position: absolute;
67   left: 11px;
68   top: 0px;
69   width: 582px;
70   height: 40px;
71   text-align: center;
72 }
73
74 div#calendar div.header a.prev,
75 div#calendar div.header a.next {
76   position: absolute;
77   top: 0px;
78   height: 17px;
79   display: block;
80   cursor: pointer;
81   text-decoration: none;
82   color: #FFF;
83 }
84
85 div#calendar div.header span.title {
86   color: #FFF;
87   font-size: 18px;
88 }
89
90 div#calendar div.header a.prev {
91   left: 0px;
92 }
93
94 div#calendar div.header a.next {
95   right: 0px;
96 }
97
98
99 /*****Calendar Content Cells*****/
100
101 div#calendar div.box-content {
102   border: 1px solid #787878;
103   border-top: none;
104 }
105
106 div#calendar ul.label {
107   float: left;
108   margin: 0px;
109   padding: 0px;
110   margin-top: 5px;
111   margin-left: 5px;
112 }
113
114 div#calendar ul.label li {
115   margin: 0px;
116   padding: 0px;
117   margin-right: 5px;
118   float: left;
119   list-style-type: none;
120   width: 80px;
```

```
121 height: 40px;  
122 line-height: 40px;  
123 vertical-align: middle;  
124 text-align: center;  
125 color: #000;  
126 font-size: 15px;  
127 background-color: transparent;  
128 }  
129  
130 div#calendar ul.dates {  
131 float: left;  
132 margin: 0px;  
133 padding: 0px;  
134 margin-left: 5px;  
135 margin-bottom: 5px;  
136 }  
137  
138  
139 /** overall width = width+padding-right**/  
140  
141 div#calendar ul.dates li {  
142 margin: 0px;  
143 padding: 0px;  
144 margin-right: 5px;  
145 margin-top: 5px;  
146 line-height: 80px;  
147 vertical-align: middle;  
148 float: left;  
149 list-style-type: none;  
150 width: 80px;  
151 height: 80px;  
152 font-size: 25px;  
153 background-color: #DDD;  
154 color: #000;  
155 text-align: center;  
156 }  
157  
158 :focus {  
159 outline: none;  
160 }  
161  
162 div.clear {  
163 clear: both;  
164 }
```