

RAPPORT DE PROJET

Groupe 6



Système Ticketing
Dyain Guiducci & Osama Shalhoub



Rapport de projet – Groupe 6

Système Ticketing - MPTickets

Table des matières

Présentation du projet	2
Logo	2
Croquis des Pages.....	3
Pages Web du Projet	4
Base de données.....	18
Langage.....	22
Softwares.....	27
Hardware.....	35
Membres et aide à la réalisation du projet.....	36
Organisation du travail – Partage de fichiers	37
Partie réseaux.....	37
Planification du travail.....	38
Structogramme - Nassi-Schneidermann.....	38
Arborescence des fichiers.....	39
Sécurité du site	40
Problèmes rencontrés	43
Amélioration pour les suites des versions du Site Web	50



Rapport de projet – Groupe 6

Système Ticketing - MPTickets

Présentation du projet

But et Objectif

Le but du projet est de développer un site **Web** qui fera rôle d'un système de Ticketing pour une entreprise.

Les utilisateurs auront la possibilité de créer un ticket en expliquant leur problème via le site web et va l'envoyer par la suite l'équipe IT. L'IT s'identifie via une page login sur le site pour consulter et traiter les tickets.

L'avantage que la solution MPTickets fourni à l'équipe IT est le classement des tickets selon la date de création du ticket et en un clin d'œil, nous pouvons voir le niveau d'urgence pour traiter les tickets en priorité. De plus la solution permet de répertorier les résolutions pour les prochains tickets qui seront similaires.

Logo

Le logo du site Web MPTickets a été créé avec le logiciel Adobe InDesign par Aurelien Kollbrunner, un ami à un des membres de projet. Il y a 2 versions de logo. La version colorée en JPG et une version noir et blanc en format PNG. Le logo en format JPG est utilisé dans le rapport de travail et dans les autres documents pour la présentation du projet. Le logo en format PNG est utilisé sur le site Web dans le header et le footer.

Logo pour les documents



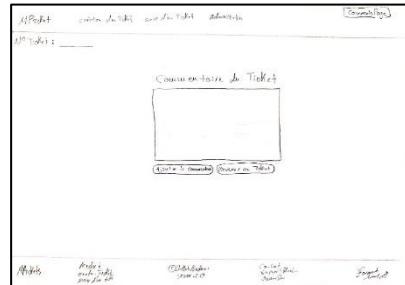
Logo pour les pages Web



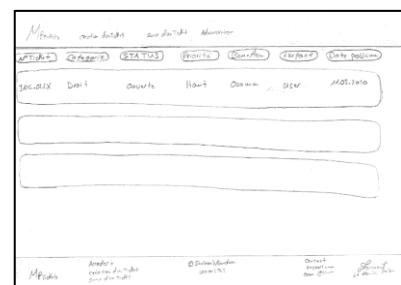
Croquis des Pages

Le croquis est un outil très utilisé dans la conception. C'est un dessin qui reflète souvent la réalité d'un endroit ou une idée qui vient de l'imagination personnelle. Avoir un croquis de chaque page Web pour le projet nous a aidés pour ajouter, modifier ou même enlever des fonctionnalités sur le projet et se donner une image plus concrète du site avant la création des pages.

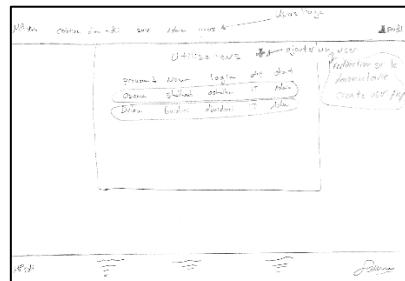
Plus on avançait dans notre projet, plus nous avions d'autres idées pour des fonctionnalités et des pages. Mais malheureusement, il n'y a pas toutes les pages Web qui ont un croquis parce que nous avions déjà une idée fixe de comment les réaliser.



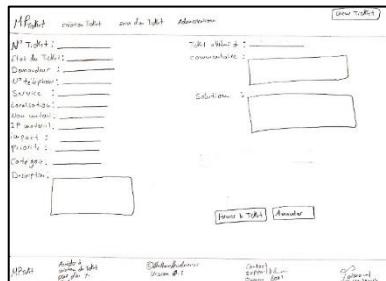
Page Add Comment



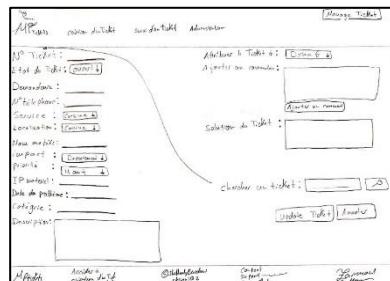
Page Dashboard Ticket



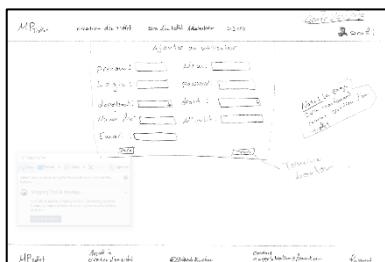
Page Dashboard User



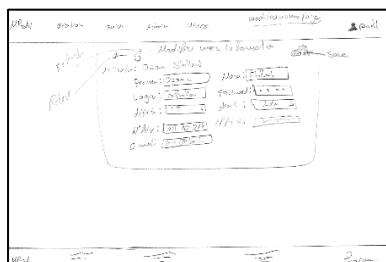
Page Consulter le ticket



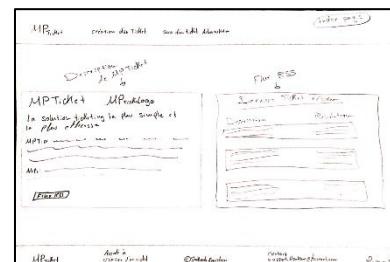
Page Manage Ticket



Page Création d'un utilisateur



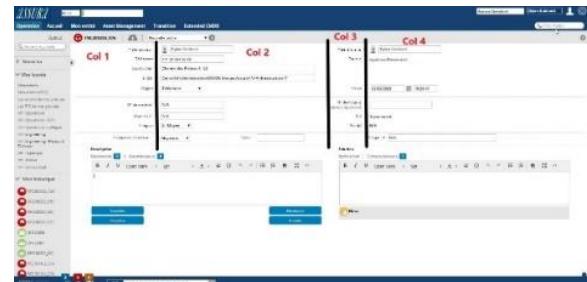
Page Manage User



Page Index



Page Remerciant



Création d'un ticket - système ASSURA



Rapport de projet – Groupe 6

Système Ticketing - MPTickets

Pages Web du Projet

Header

Le header est appelé sur toutes les pages. Il est codé en HTML, CSS et PHP. Au début du header, il y a les importations des classes Bootstrap, CSS et JavaScript.

Si l'utilisateur n'est pas authentifié sur le site Web, le header aura juste les liens pour accéder aux pages suivantes :

- Index
- Création d'un ticket
- Suivi d'un ticket
- Administration



Mais lorsque l'utilisateur est authentifié / connecté, le header sera modifié. L'onglet « Administration » devient un sous menu qui affichera les liens du Dashboard Tickets et Users et un nouveau sous-menu existera « Mon Profil (nom du profil) » qui affichera les liens suivants :

- Mes Tickets
- Mes Infos
- Déconnexion (Lien vers la page qui va fermer la session)



Footer

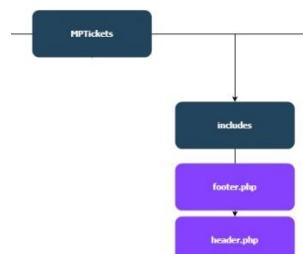
Le Footer, comme le header, est appelé sur toutes les pages. Il est codé en HTML et CSS.

Il ne se modifie pas si un utilisateur est connecté ou pas mais il contient :

- Les logos du site Web MPTickets et Montreux Palace
- Liens pour accéder aux différentes pages
- Copyright de la création du site
- Version du site
- Information du contact du support IT



Les fichiers de sources pour ces 2 parties du site se trouvent dans le même dossier includes vu que les 2 sont inclus partout sur le site.





Rapport de projet – Groupe 6

Système Ticketing - MPTickets

Page Index

The screenshot shows the main index.php page of the MPTickets system. At the top, there's a navigation bar with links for 'Création d'un ticket', 'Suivi d'un ticket', and 'Administration'. Below the header, there's a large image of the Fairmont Le Montreux Palace hotel. On the left, a sidebar contains information about the solution and an RSS feed icon. The main content area features a table titled 'Derniers tickets résolus' (Last resolved tickets) with columns for 'Description' and 'Résolution'. The table lists five resolved tickets with their descriptions and resolutions.

La page principale index.php est codés en HTML, CSS, et du PHP. Du Bootstrap est utilisé pour le design de la page.

La page contient un lien pour le flux RSS qui affiche 5 derniers tickets résolus grâce à une requête SQL qui se connecte sur la base de données et récupère la description du problème ainsi que la résolution. Ensuite les données sont affichées dans un tableau HTML. La récupération des 5 tickets résolus sont affichés en tant que flux RSS. Pour faire l'importation des données la fonction import_LastTickets() est appelée depuis la page.

Sur la gauche de la page, c'est la présentation du site Web MPTickets.

Page Création d'un ticket

The screenshot shows the 'Création de ticket' (Ticket Creation) form. The form is titled 'Création de ticket' and includes fields for 'Nº Ticket' (set to 20200607_001), 'Demandeur' (Dylan Guiducci), 'Téléphone Demandeur' (+41 21 544 38 48), 'Département' (Achats), 'Localisation' (Funky Claude's Bar), 'Nº du matériel' (PC160065), 'Nom du matériel (Demande)' (adsl-84-226-80-150.adsl), 'Date du problème' (mm/dd/yyyy), 'IP du matériel (Demande)' (84.226.80.150), 'Impact' (Medium), 'Priorité' (Basse - Priorité Basse), 'Catégorie' (Informatique/Profil bloqué), and 'Description'. At the bottom, there are 'Annuler' and 'Terminer' buttons. The page has a footer with links for 'Accéder à...', 'Création d'un ticket', 'Suivi d'un ticket', 'Administration', contact information (support.montreux@fairmont.com, Osama Shalhoub - 8004), and the Fairmont Le Montreux Palace logo.

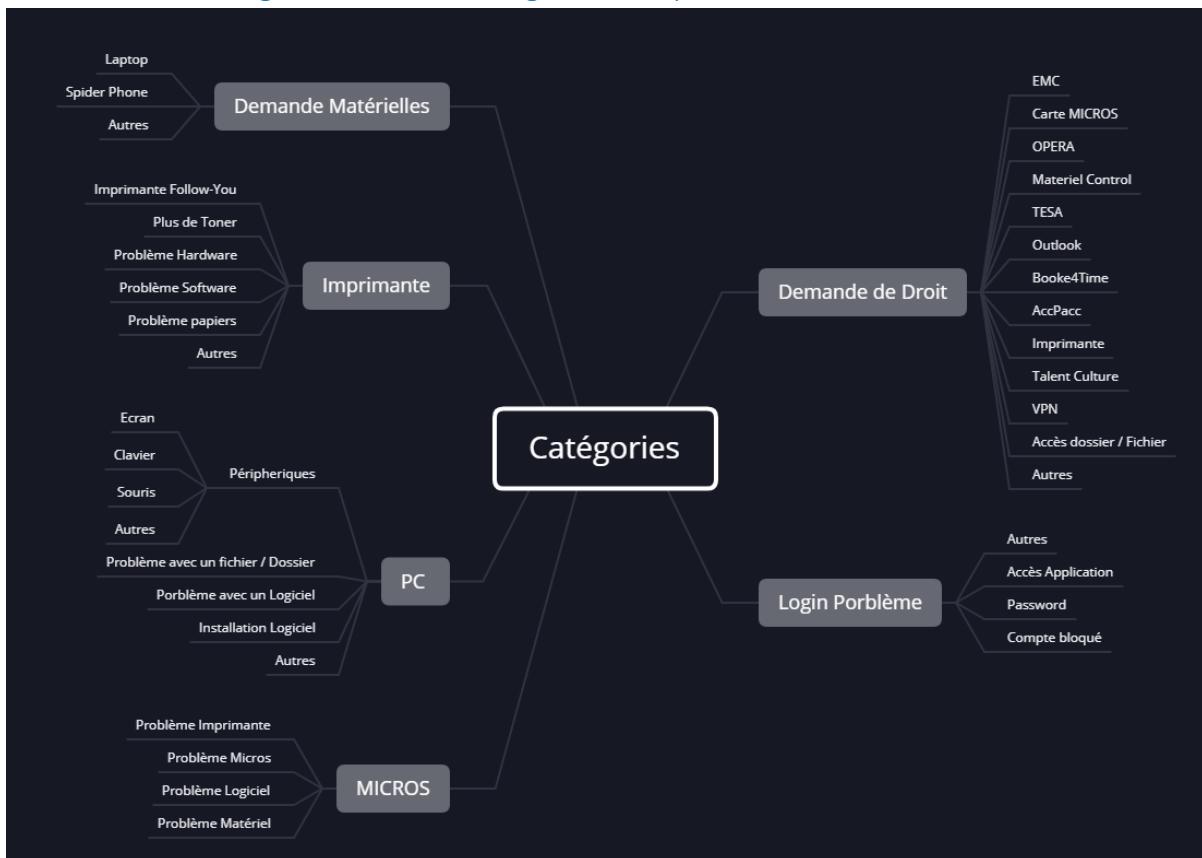
creationTicket.php Création d'un ticket est la page du site Web qui permet aux utilisateurs de pouvoir créer leur ticket. La page est développée en HTML, CSS, PHP et utilise du SQL depuis la page "insertTicket.php".

Le formulaire doit être rempli par l'utilisateur en spécifiant les informations concernant le problème comme l'utilisateur qui crée la demande, son département, son numéro de téléphone et autres.

Certaines informations sont remplies automatiquement grâce à des fonctions PHP comme l'adresse IP de la machine qui a été utilisée pour créer le ticket, le nom du matériel et la date de la création du ticket qui n'est pas d'ailleurs visible sur le formulaire.

Afin de choisir la catégorie du problème, l'utilisateur pourra accéder à différentes catégories en appuyant sur le bouton qui se trouve juste à côté du champ. Tout cela est géré avec JavaScript pour l'affichage, du SQL pour récupérer les catégories depuis la base de données et du PHP pour les fonctions.

La structure et l'organisation des catégories des problèmes sont comme ceci :



L'utilisateur aura directement l'ID du ticket affiché en haut du formulaire. L'ID du ticket est créé à l'aide d'un script PHP stocké dans "Fonction_PHP-SQL.php" du dossier BD. Pour créer l'ID du ticket, le script fonctionne de la manière suivante :

1. Récupère la date d'aujourd'hui au format YYYYMMDD
2. Compte le nombre de ticket du jour.
3. Incrémente 1 au nombre de ticket



Rapport de projet – Groupe 6

Système Ticketing - MPTickets

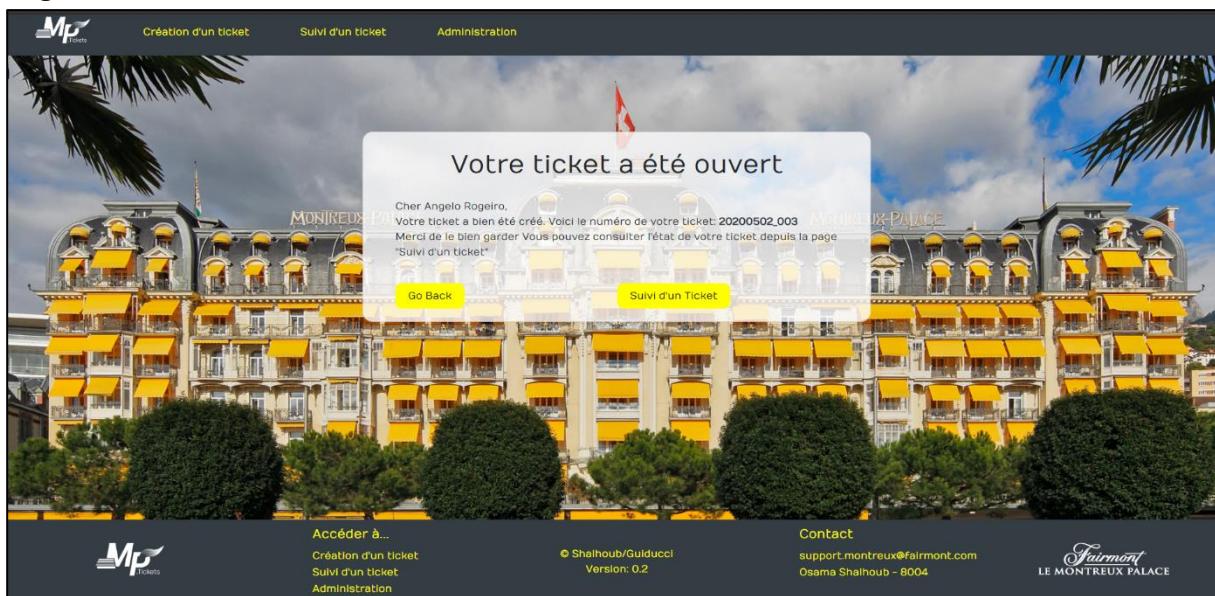
```
// Créer le nouvel id du ticket du jour
$nbRequest++;
$nbRequest = substr("000".$nbRequest,-
3); // Passer en affichage 000 au lieu de 0
$idTicket = $date."_".$nbRequest;

return $idTicket;
```

Puis finalement, en appuyant sur la touche "Terminer" cela va appeler la requête SQL qui va permettre d'insérer le ticket avec toutes ses informations dans la table T_Tickets de la base de données.

```
INSERT INTO T_Tickets (id_ticket, Utilisateur_demandeur, Numero_tel_demandeur,
fk_id_status, fk_id_impact, fk_id_priorite, fk_id_departement, Date_Probleme,
IP, Nom_machine_creation, Nom_machine_probleme, Localisation, Description_ticket)
VALUES (''.$id_ticket.'', ''.$Utilisateur_demandeur.'', ''.$Numero_tel_demandeur.'',
''.$fk_id_status.'', ''.$fk_id_impact.'', ''.$fk_id_priorite.'', ''.
.$fk_id_departement.'', ''.$Date_Probleme.'', ''.$IP.'', ''.$Nom_machine_creation.'',
''.$Nom_machine_probleme.'', ''.$Localisation.'', ''.$Description_ticket.'');
```

Page Remerciements d'ouverture du ticket



Tout simplement la page `remerciementCreationTicketPage.php` est une page qui va afficher un message de remerciement après avoir ouvert le ticket en affichant le nom du demandeur et l'ID du ticket.

On récupère aussi la valeur des variables `Demandeur` et `NumTicket` avec la méthode GET afin de les afficher dans le message.

```
<?php echo $_GET['Demandeur']; ?>
<?php echo $_GET['NumTicket']; ?>
```

Du JavaScript a été utilisé aussi sur le bouton « Go Back » pour faire un retour arrière.

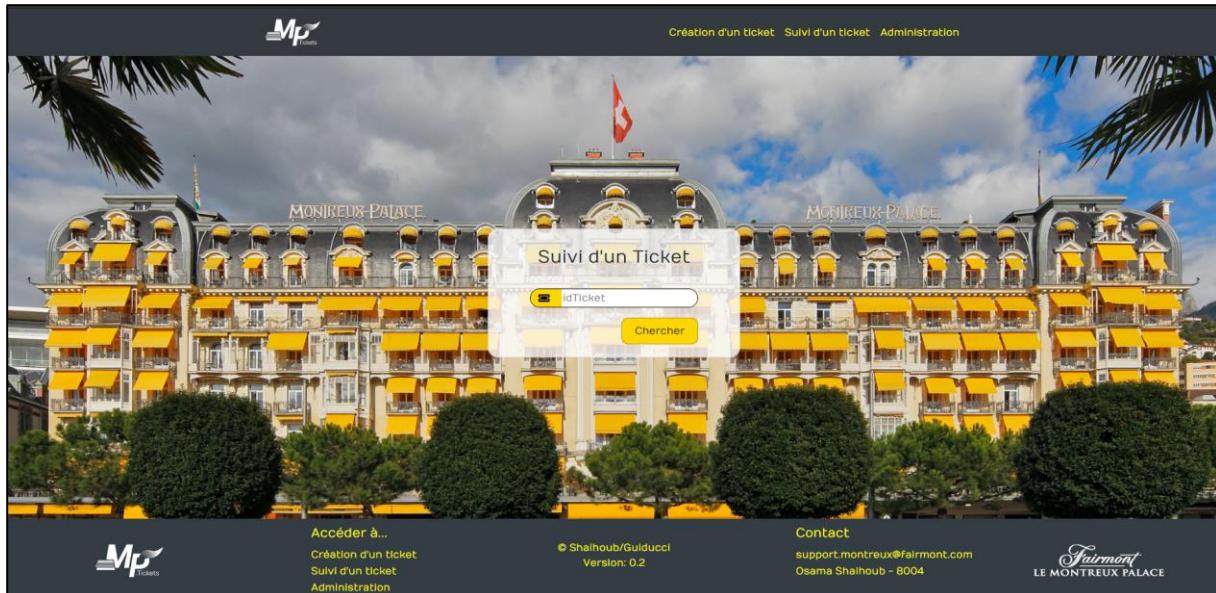
Le bouton « Suivi du Ticket » permet à l'utilisateur d'être redirigé sur la page `Suivi Ticket` pour suivre le ticket.

```

<tr>
    <td class="Td_Remerciement"><button class="Bouton centerH" type="button" onClick="history.back()">Go Back</button></td>
    <td class="Td_Remerciement"><a href="suiviTicketPage.php" target="_blank"> <button class="Bouton centerH" type="button">Suivi d'un Ticket</button> </a></td>
</tr>

```

Page Suivi d'un ticket



suiviTicketPage.php est la page Suivi d'un Ticket. Son rôle est de permettre aux utilisateurs de voir la progression de leur ticket. Pour visualiser le ticket, il suffit d'insérer l'ID du ticket et celui-ci sera directement affiché. L'utilisateur pourra y voir les commentaires et le statut du ticket.

On récupère l'ID du ticket grâce à la méthode POST et on l'assigne au variable \$currentNumTicket.

Ensuite nous vérifions si le ticket existe à l'aide la fonction **check_Ticket()** avec comme paramètre le numéro du ticket inséré par l'utilisateur. C'est une fonction booléane donc si le ticket n'existe pas elle va retourner **false** et s'il existe **true**. Donc le code ci-dessous vérifie si le retour est égal à **false** et si c'est le cas. L'utilisateur est redirigé sur la page « **ticketNotFoundPage.php** ».

```

$currentNumTicket = $_POST['idTicket'];

// Vérifie si le ticket n'existe pas
if(check_Ticket($currentNumTicket) === false)
{
    // Redirige vers la page ticket pas trouvé
    header("Location: /MPTickets/body/ticketNotFoundPage.php?idTicket='".$currentNumTicket "'");
}

```



Rapport de projet – Groupe 6

Système Ticketing - MPTickets

Page Ticket inexistant

The screenshot shows a web page with a large background image of the Fairmont Le Montreux Palace hotel. Overlaid on the image is a white modal dialog box with a yellow border. The dialog box contains the text "Ticket inéxistant" and "Nos excuses, votre ticket 'merde', n'a pas été trouvé. Merci de bien vérifier l'ID du ticket." Below this text is a yellow button labeled "Suivi d'un Ticket". At the top of the page, there is a navigation bar with links: "Création d'un ticket", "Suivi d'un ticket", and "Administration". At the bottom, there is a footer with links: "Accéder à...", "Création d'un ticket", "Suivi d'un ticket", "Administration", "Shalhoub/Guiducci Version: 0.2", "Contact support.montreux@fairmont.com Osama Shalhoub - 8004", and the Fairmont logo.

La page `ticketNotFoundPage.php` Ticket inéxistant récupère l'ID du ticket avec la méthode GET.

La page permet d'informer l'utilisateur que l'ID du ticket qu'il a inséré n'existe pas. L'utilisateur est invité à répéter l'opération l'aide d'un bouton qui le redirige vers la page « Suivi d'un Ticket ».

Page View Ticket

The screenshot shows a web page with a large background image of the Fairmont Le Montreux Palace hotel. Overlaid on the image is a detailed ticket information form. The form includes fields for ticket number (20200502_001), status (Résolu), assignee (Osama Shalhoub), and various dates and times. It also includes sections for problem description, comments, and resolution. A sidebar on the left provides access to other ticketing functions. At the bottom, there is a footer with links: "Accéder à...", "Création d'un ticket", "Suivi d'un ticket", "Administration", "Shalhoub/Guiducci Version: 0.2", "Contact support.montreux@fairmont.com Osama Shalhoub - 8004", and the Fairmont logo.

`viewTicketPage.php` permet aux utilisateurs non-authentifiés de consulter en lecture le ticket, afin de voir l'avancement du travail ou à quel dépanneur le ticket est attribué ainsi s'il y a eu un commentaire ou la solution qui s'est ajouté.

Pour accéder à cette page, il faut insérer l'ID du ticket sur la page « Suivi d'un Ticket » qui a été fourni lors de sa création. Ensuite grâce à l'ID du ticket, toutes les informations s'ajoutent aux champs sur la page afin que l'utilisateur puisse consulter le ticket grâce à la variable :

```
$currentNumTicket = $_POST['idTicket'];
```



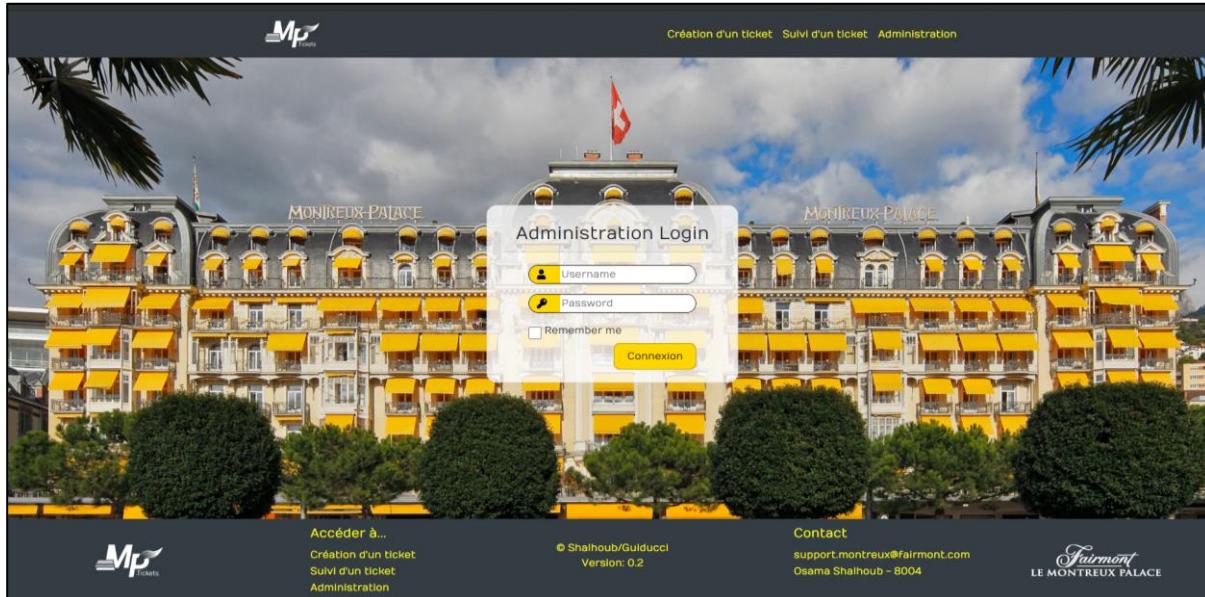
Rapport de projet – Groupe 6

Système Ticketing - MPTickets

Pareil pour les commentaires. Les commentaires qui concernent le ticket y sont affichés sur la page avec la fonction :

```
<?php import_TicketCommentaire($currentNumTicket); ?>
```

Page Administration



`loginAdminPage.php` **Administration Login** permet aux utilisateurs de s'identifier et pouvoir accéder aux différents tableaux de bord afin de gérer les tickets et les utilisateurs. La page utilise du HTML, CSS, PHP, Bootstrap. L'identification se fait grâce à une requête SQL depuis la page « login.php » grâce au bouton « Connexion ».

La connexion est sécurisée avec la fonction PHP `htmlspecialchars()` pour le nom d'utilisateur et le mot de passe afin d'éviter les injections SQL.

Les étapes de la connexion :

1. Récupérer le nom d'utilisateur et le mot de passe inséré en évitant les injections SQL.
2. Vérifier si l'utilisateur existe dans la base de données.
3. Si l'utilisateur n'existe pas,
 - a. Rediriger l'utilisateur sur la page Login.
4. Si l'utilisateur existe,
 - a. Récupérer le SALT de l'utilisateur et le hash du mot de passe.
5. Générer le hash du mot de passe avec le SALT.
6. Vérifier si le hash du mot de passe de l'utilisateur avec celui qui a été renseigné.
7. Si les hashes correspondent cela veut dire que l'identification s'est faite avec succès.
 - a. Ouvrir une session
 - b. Rediriger l'utilisateur sur la page Dashboard Ticket.
8. Si les mots de passe ne correspondent pas
 - a. Rediriger l'utilisateur sur la page Login.

Tout le processus se fait sur la page `Login.php` qui se trouve dans le dossier BD.



Rapport de projet – Groupe 6

Système Ticketing - MPTickets

Page Dashboard Tickets

La page Dashboard Tickets dashboardTicketPage.php est accessible uniquement par les administrateurs qui se sont connectés depuis la page Administration.

Son rôle est d'afficher uniquement les tickets ouverts ou suspendus et qui ne sont toujours pas attribués à un dépanneur.

Fonctionnement de la requête :

1. Récupère l'ID de chaque ticket, sa catégorie, sa priorité, son statut, le demandeur, son impact et la date du problème.
2. Fait le lien avec les autres tables pour récupérer le nom et pas l'ID de l'entrée.
3. Filtrer pour que le statut soit égal à Ouvert ou Suspendu et qu'il n'y ait pas de dépanneur.
4. Trier par défaut du plus récent au plus vieux.

La requête SQL se trouve dans une fonction PHP qui va ensuite permettre l'affichage dans le tableau.

Il est possible de trier les tickets par colonne ainsi qu'inverser le trier. Pour la gérer les tris, le fichier sort_TicketDashboard.php est utilisé.

```
SELECT T.id_ticket, C.Categorie, P.Priorite, S.Status, T.Utilisateur_demandeur
, I.Impact, T.Date_Probleme
    FROM T_Tickets AS T
    LEFT JOIN T_Priorites AS P ON P.id_priorite = T.fk_id_priorite
    LEFT JOIN T_Categories AS C ON C.id_categorie = T.fk_id_categorie
    LEFT JOIN T_Status AS S ON S.id_status = T.fk_id_status
    LEFT JOIN T_Impacts AS I ON I.id_impact = T.fk_id_impact
    WHERE S.Status IN ("Ouvert", "Suspendu")
    AND T.fk_id_utilisateur_depanneur IS NULL
    ORDER BY '.$fun_colSort.' '.$fun_sort;;
```

Pour modifier le ticket, il suffit de cliquer sur le ticket voulu et l'utilisateur est redirigé vers la page sur la page « manageTicketPage.php ».

Page Manage Ticket

The screenshot shows the 'Manage Ticket' page with the following details:

- N° Ticket :** 20200502_002
- Status du Ticket :** Ouvert
- Demandeur :** Angelo Rogeiro
- Département :** Direction
- Dépanneur :** Osama, Shalhoub
- Téléphone Demandeur :** +41 70 700 00 00
- Localisation :** chemin de la rue
- Date d'ouverture :** 2020-05-02 17:11:43
- Date de fermeture :** [Empty]
- Impact :** Utilisateur
- Catégorie :** Priorité : Basse
- Description :** J'ai un problème avec ma souris. Merci de m'aider au plus vite, sinon vous êtes virés. Merci et bisou Mr.Rogeiro Directeur générale
- Commentaire :**

DATE	NOM	COMMENTAIRE
2020-05-02 17:36:26	Shalhoub Osama	Hello Mr. Rogeiro. Your problem will be solved when you go to pay us our salary. Best regards Osama
- Boutons :** Add Commentaire, Suspendre, Résoudre

At the bottom, there are links to 'Accéder à...', 'Création d'un ticket', 'Suivi d'un ticket', 'Administration', and contact information for 'Shalhoub/Guiducci'.

`manageTicketPage.php` Manage Ticket est la page qui permet de modifier le ticket créé par les utilisateurs. La page est seulement accessible par les utilisateurs identifiés.

Il est possible de modifier certaines informations comme le nom du matériel, l'impact, la priorité et la catégorie. De plus, il est possible aussi d'assigner le ticket à un dépanneur.

La page contient aussi un champ commentaire avec la date et le nom du commentaire.

Pour parvenir sur cette page, il faut passer par la page Dashboard Ticket ou Mes Tickets.

Il est possible de changer l'état du ticket pour 3 états différents grâce aux boutons qui se trouvent en bas où il est obligatoire d'ajouter un commentaire afin de justifier le changement de l'état du ticket.

Pages Update Statut Ticket

The three screenshots show the following update status pages:

- Suspendre un ticket**: N° Ticket: 20200502_001, Auteur: Dylan, Guiducci. Commentaire: Le ticket est suspendu du au retard de la commande de la nouvelle WebCam pour monsieur Lohri. Boutons: Retour, Suspendre.
- Dépendre un ticket**: N° Ticket: 20200502_001, Auteur: Dylan, Guiducci. Commentaire: Le fournisseur de la WebCam a bien livré le colis. En attendant d'un rendez-vous avec M.Lohri pour changer la WebCam. Boutons: Retour, Dépendre.
- Résoudre le ticket**: N° Ticket: 20200502_001, Auteur: Osama, Shalhoub. Commentaire: Le changement de la cam a été effectué. Une augmentation de salaire a été fait aussi. Osam. Boutons: Retour, Résoudre.

Il est possible de changer l'état du ticket grâce aux boutons qui se trouvent dans la page `ManageTicket.php`, `resolveTicket.php`, `suspendTicket.php` contiennent les fonctions et les requêtes SQL UPDATE afin de modifier l'état du ticket.

L'état du ticket peut se changer en :

- **Suspend** : Le ticket est suspendu pour le moment (Attente livraison, le problème dépend d'un autre problème).
- **Dépendre** : Le ticket est dépendu car nous pouvons le traiter (Matériel reçu, le problème parent a été résolu).
- **Résolu** : Le problème est résolu/terminer :



Rapport de projet – Groupe 6

Système Ticketing - MPTickets

Page Add Comments

The screenshot shows a web application interface for adding a comment to a ticket. The background features a scenic view of the Fairmont Le Montreux Palace hotel. The main content is a modal dialog titled "Ajouter un commentaire". Inside the dialog, there are fields for "N° Ticket" (set to 20200502_002), "Auteur" (set to Osama, Shalhoub), and a "Commentaire" text area containing a message. At the bottom of the dialog are two buttons: "Retour" (Return) and "Créer Commentaire" (Create Comment). The footer of the page includes links for "Accéder à...", "Contact", and the Fairmont logo.

La page `commentsPage.php` sert à ajouter des commentaires sur un ticket. Afin d'ajouter un commentaire, il faut sélectionner qui est l'auteur du commentaire puis ensuite appuyer sur le bouton « Crée Commentaire ».

Tous les commentaires, peu importe le ticket, sont stockés dans la table `T_Commentaires`.

Voici la requête SQL qui permet l'insertion du commentaire dans la base de données :

```
'INSERT INTO T_Commentaires (fk_id_ticket, fk_id_utilisateur, Commentaire)
VALUES ('".$NumTicket."', '".$Auteur."', '".$Commentaire."');
```

Page Dashboard Users

The screenshot shows a dashboard for managing users. The background features a scenic view of the Fairmont Le Montreux Palace hotel. In the center, there is a table titled "Dashboard Users" displaying user information for five individuals. The table has columns for "Prénom", "Nom", "Login", "Email", "Département", and "Droit". The users listed are Angelo Rogeiro, De Test, Didier Lohri, Dylan Guiducci, and Osama Shalhoub, each with their respective details. At the bottom of the page is a footer with links for "Accéder à...", "Contact", and the Fairmont logo.



Rapport de projet – Groupe 6

Système Ticketing - MPTickets

`dashboardUsersPage.php` Dashboard Users est une page qui a un rôle comme « Dashboard Tickets » mais pour les utilisateurs. Tous les dépanneurs qui peuvent s'identifier sur le site sont affichés sur cette page.

En cliquant sur un des utilisateurs, cela va permettre de modifier les données de l'utilisateur comme le n° de téléphone, l'email, le département, et bien sûr le mot de passe avec la page « Manage User ». Cette page permet aussi de créer de nouvel utilisateur grâce au bouton qui fait le lien avec la page « Création d'un utilisateur ».

Tout cela est grâce à la fonction :

```
import_AllUsers($sort_col, $sort);
```

Page Création d'un utilisateur

`creationUserPage.php` permet de créer un nouvel utilisateur en remplissant le formulaire. La création des utilisateurs se fait exactement comme la création d'un ticket. Cela veut dire avec une requête SQL `INSERT` depuis la page `insertUser.php`.

Le fonctionnement de la création de l'utilisateur :

1. Récupère les données du formulaire
2. Crée le Salt de l'utilisateur
 - a. Créer une chaîne de 8 caractères
 - b. Chiffré le Salt en sha-256
3. Hasher le mot de passe avec le Salt venant d'être créer
4. Importer les données de l'utilisateur, le Salt et le hash du mot de passe dans la base de données

```
// Créer le Salt du user et le hasher
$salt_user = create_SaltUser();

// Hasher le mot de passe
$Password = crypt($Password, '$6$rounds=5000$' . $salt_user); // crypt(password, "Type de Hash".$salt); "$6$rounds=5000$" == SHA512 & "$5$rounds=5000$" == SHA256
```



Rapport de projet – Groupe 6

Système Ticketing - MPTickets

```
INSERT INTO T_Utilisateurs (fk_departement, fk_droit, Login, salt_user, Password, Nom, Prenom, Numero_Fix, Numero_Mob, Email)
VALUES (''.{$fk_departement}'',''.{$fk_droit}'',''.{$Login}'',''.{$salt_user}'', ''{$Password}'', ''{$Nom}'', ''{$Prenom}'', ''{$Numero_Fix}'',''.{$Numero_Mob}'', ''{$Email}'');
```

Page Manage User



`manageUserPage.php` Manage User gère les infos des utilisateurs. La page est accessible en cliquant sur l'utilisateur en question depuis la page « Dashboard Users » ou « Mes Infos ».

Les données des utilisateurs sont toutes importées grâce à la fonction `import_UserInfo()` :

```
$currentUser = $_GET['Login'];
$DonneesUser = import_UserInfo($currentUser);
```

Les informations des utilisateurs qu'ils peuvent être modifiés sont :

- Le numéro de téléphone fixe
- Le numéro de téléphone portable
- L'email
- Le mot de passe
- Le département
- Le droit

Concernant la réinitialisation du mot de passe, l'utilisateur devra entrer le nouveau mot de passe dans le champ Nouveau mot de passe et Répéter le nouveau mot de passe puis ensuite appuyer sur le bouton Réinitialiser.

Tout cela se fait avec l'ordre suivant :

1. Vérifier que les deux mots de passe soit identique
2. Création du nouveau `salt_user`
3. Hasher le mot de passe
4. Création et envoi de la requête
5. Modifier le mot de passe

Page Remerciements de création d'un utilisateur



Exactement comme la page `remerciementCreationTicketPage.php`, la page `remerciementCreationUserPage.php` est une page qui va afficher un message de confirmation lors de la création d'un nouvel utilisateur en affichant le nom de l'utilisateur.

Afin d'afficher le nom de l'utilisateur qui a été créé, on récupère la valeur de la variable avec `nomNouveauUser` avec la méthode GET afin de l'afficher dans le message.

```
<tr>
    <td class="Td_Remerciement" colspan="2"><p>Le nouvel utilisateur Mr.<span class="Gras"> <?php echo $_GET['nomNouveauUser']; ?></span> a bien été crée.<br>Vous pouvez consulter les informations de Mr.<span class="Gras"> <?php echo $_GET['nomNouveauUser']; ?></span> depuis la page Dashbpard User sous Administration</p></td>
</tr>
```

Finalement, la page contient aussi un bouton Go Back qui permet de retourner en arrière et un bouton Dashboard User qui permet de rediriger l'utilisateur sur la page Dashboard User.

```
<tr>
    <td class="Td_Remerciement"><button class="Bouton centerH" type="button" onClick="history.back()">Go Back</button></td>
    <td class="Td_Remerciement"><a href="dashboardUsersPage.php" target="_blank"> <button class="Bouton centerH" type="button">Dashboard User</button> </a></td>
</tr>
```



Rapport de projet – Groupe 6

Système Ticketing - MPTickets

Page Mes Tickets

The screenshot shows a dashboard titled 'Mes Tickets' with a list of three tickets. The columns are labeled: ID du Ticket, Catégorie, Priorité, Status, Demandeur, Impact, and Date du problème. The first ticket is '20200604_002' (Souris) with 'Haute' priority, 'Ouvert' status, 'Dylan Guiducci' as the demander, 'Utilisateur' impact, and '2020-06-05 12:06:00' as the date. The second ticket is '20200502_002' (Login Problème) with 'Basse' priority, 'Ouvert' status, 'Dylan Guiducci' as the demander, 'Utilisateurs' impact, and '2020-04-27 00:06:00' as the date. The third ticket is '20200427_001' (Souris) with 'Basse' priority, 'Ouvert' status, 'Angelo Rogeiro' as the demander, 'Utilisateur' impact, and '2020-12-01 01:00:00' as the date. The page also includes a footer with links to 'Accéder à...', 'Contact', and the 'Fairmont LE MONTREUX PALACE' logo.

dashboardMyTicketsPage est la page qui dépend de chaque utilisateur. Le rôle cette page est exactement comme la page Dashboard Ticket mais elle importe uniquement les tickets qui sont assignés au dépanneur connecté.

Les tickets sont tous importés grâce à la fonction :

```
import_AllMyTicket($sort_col, $sort, $_SESSION['username']);
```

Par exemple, voici la page « Mes Tickets » pour le dépanneur Dylan. Si on vérifie le dépanneur de chaque ticket, on voit bien que c'est bien le dépanneur Dylan.

The screenshot shows a 'Mes Tickets' page for Dylan. It lists two tickets: '20200504_002' (Compte bloqué) and '20200330_002' (Demande de droit). Both tickets are 'Ouvert' (Open) and assigned to Dylan Guiducci.

Manage Ticket

Nº Ticket : 20200330_002

Status du Ticket :	Ouvert	Dépanneur :	Dylan, Guiducci
Demandeur :	Moussa Shalhoub	Téléphone Demandeur :	34 54
Département :	Comptabilité	Localisation :	Montreux, bureau 469

Manage Ticket

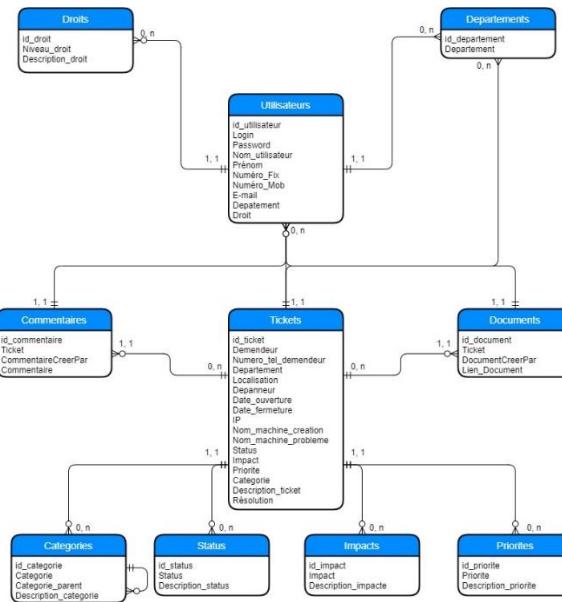
Nº Ticket : 20200504_002

Status du Ticket :	Ouvert	Dépanneur :	Dylan, Guiducci
Demandeur :	Béatrice Piguet	Téléphone Demandeur :	+41215443848
Département :	Comptabilité	Localisation :	Bioley

Base de données

MCD

Le Modèle Conceptuel de Données contient 11 tables qui sont :



- Utilisateurs**: La table qui contient les données sur les dépanneurs qui peuvent se connecter sur le site.
 - ID_Utilisateur** : L'id de l'utilisateur
 - Login** : Le nom d'utilisateur
 - Password** : Le mot de passe de l'utilisateur qui permet de se connecter
 - Nom_Utilisateur** : Le nom de l'utilisateur
 - Prenom** : Le prénom de l'utilisateur
 - Numero_Fix** : Le numéro fixe de l'utilisateur
 - Numero_Mob** : Le numéro mobile de l'utilisateur
 - Email** : L'adresse mail de l'utilisateur
 - Departement** : Le département de l'utilisateur
 - Droit** : Niveau de droit de l'utilisateur
- Cookie** : La table permet de stocker les cookies qui sont utilisés pour le remember me sur la page de login
 - ID_Cookie** : L'ID du cookie
 - Utilisateur** : La clé étrangère de l'ID Utilisateur
 - Token** : Le résultat du chiffrement en sha-516 de l'IP de l'utilisateur avec son nom d'utilisateur
- Catégories** : Les différentes catégories qu'un ticket peut avoir
 - Id_categorie** : L'id de la catégorie
 - Categorie** : le nom de la catégorie
 - Categorie_Parent** : La catégorie parent
 - Description_categorie** : La description de la catégorie
- Département** : Les départements de l'entreprise (Marketing, Comptabilité, cuisine...)
 - id_departement** : L'id du département
 - Departement** : Le nom du département



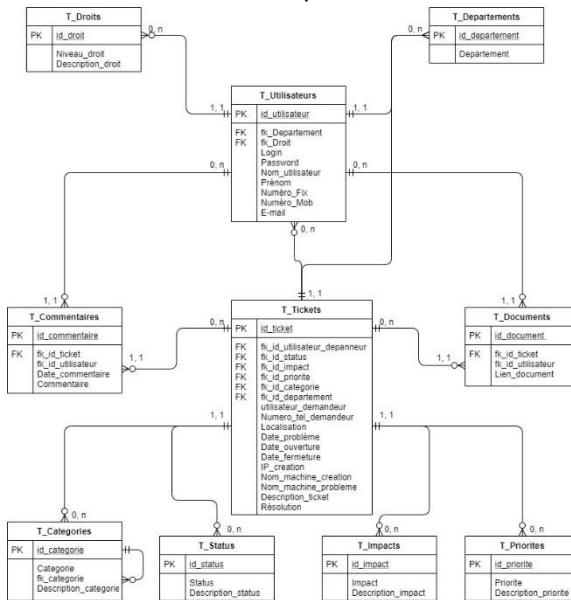
Rapport de projet – Groupe 6

Système Ticketing - MPTickets

- **Tickets** : Les informations des tickets comme la date, la description et autres.
 - **id_ticket** : L'id du ticket
 - **Demendeur** : Nom et prénom de la personne qui a créé le ticket
 - **Numero_tel_demandeur** : Numéro de téléphone de la personne
 - **Localisation** : La localisation où il y a eu le problème
 - **Depanneur** : Qui va s'occuper de la résolution du ticket
 - **Date_ouverture** : La date de l'ouverture du ticket
 - **Date_probleme** : La date quand le problème est apparu
 - **Date_fermeture** : La date de la fermeture du ticket
 - **IP** : l'adresse IP de la machine avec laquelle le demandeur a ouvert le ticket
 - **Nom_machine_creation** : Nom de la machine qui a créé le ticket
 - **Nom_machine_Probleme** : Nom de la machine qui a le problème
 - **Status** : Le statut du ticket ex. ouvert, fermé, en suspendu
 - **Impact** : l'impact du problème ex. entreprise, département, utilisateur
 - **Priorite** : La priorité du ticket comme haut, bas, moyen
 - **Categorie** : La catégorie du problème
 - **Description_ticket** : La description du problème
 - **Resolution** : La résolution du ticket
- **Droits** : Listes déroulante des droits des utilisateurs (Contrôle Total, Modification et lecture seule)
 - **id_droit** : L'id du droit
 - **Niveau_droit** : Le nom du droit
 - **Description_droit** : La description du droit
- **Commentaires** : Commentaire sur le ticket
 - **id_commentaire** : L'id du commentaire
 - **Ticket** : Le numéro du ticket auquel le commentaire est lié
 - **Commentaire** : Les commentaires
 - **CommentaireCreePar** : Nom de la personne qui a créé le commentaire
- **Status** : Les statuts qu'un ticket peut avoir (Fermé, Ouvert, en cours)
 - **id_status** : L'id du status
 - **Status** : Le nom du statut
 - **Description_status** : La description du statut
- **Impacts** : Les impacts que le problème peut avoir (User, Users, Départements)
 - **id_impacte** : L'id de l'impact
 - **Impacte** : Le nom de l'impact
 - **Description_impacte** : La description de l'impact
- **Priorites** : Les niveaux de la priorité qu'un ticket peut avoir (Urgent, Critique, Bas)
 - **id_priorite** : L'id de la priorité
 - **Priorite** : Le nom de la priorité
 - **Description_priorite** : La description de la priorité
- **Documents** : Les documents qu'un ticket peut avoir en annexe
 - **Id_document** : L'id du document
 - **Ticket** : Le numéro du ticket auquel le document est lié
 - **DocumentCreePar** : Par qui le document est créé
 - **Lien_Document** : Le lien du document

MLD

Le Modèle Logique de Donné contiendra 10 tables qui sont :



- T_Utilisateurs** : La table qui contient les données sur les administrateurs qui peuvent eux seules se connecter sur le projet
 - Fk_Departement** : Clé étrangère de la table T_Dpeartement
 - Fk_Droit** : Clé étrangère de la table T_Droit
 - Id_Utilisateur** : L'id de l'utilisateur
 - Login** : Le login de l'utilisateur
 - Password** : Le mot de passe qui permet de se connecter
 - Nom_Utilisateur** : Le nom de l'utilisateur
 - Prenom** : Le prénom de l'utilisateur
 - Numero_Fix** : Le numéro fixe de l'utilisateur
 - Numero_Mob** : Le numéro mobile de l'utilisateur
 - Email** : L'adresse mail de l'utilisateur
 - Departement** : Le département de l'utilisateur
 - Droit** : Niveau de droit de l'utilisateur
- T_Cookie** : La table permet de stocker les cookies qui sont utilisés pour le remember me sur la page de login
 - ID_Cookie** : L'ID du cookie
 - Utilisateur** : La clé étrangère de l'ID Utilisateur
 - Token** : Le résultat du chiffrement en sha-516 de l'IP de l'utilisateur avec son nom d'utilisateur
- T_Tickets** : Les informations des tickets comme la date, la description et autres.
 - Fk_id_utilisateur_depanneur** : Clé étrangère avec la table T_Utilisateurs
 - Fk_id_status** : Clé étrangère de la table T_Status
 - Fk_id_impact** : Clé étrangère de la table T_Impacts
 - Fk_id_priorite** : Clé étrangère de la table T_Priorites
 - Fk_id_catégorie** : Clé étrangère de la table T_Categories
 - Fk_id_departement** : Clé étrangère de la table T_Departements
 - id_ticket** : L'id du ticket



Rapport de projet – Groupe 6

Système Ticketing - MPTickets

- **Utilisatur_demandeur** : Nom du demandeur du ticket
- **Numero_tel_demandeur** : Numéro du demandeur
- **Localisation** : La localisation où il y a eu le problème
- **Date_probleme** : La date quand le problème est apparu
- **Date_fermeture** : La date de la fermeture du ticket
- **IP** : l'adresse IP de la machine avec laquelle le demandeur a ouvert le ticket
- **Nom_machine_creation** : Nom de la machine qui a créé le ticket
- **Nom_machine_probleme** : Nom de la machine qui a le problème
- **Description_ticket** : La description du problème
- **Resolution** : La résolution du ticketB
- **T_Departements** : Les départements de l'entreprise (Marketing, Comptabilité, cuisine...)
 - **id_departement** : L'id du département
 - **Departement** : Le nom du département
- **T_Droits** : Listes déroulante des droits des utilisateurs (Administrator, Normal)
 - **id_droit** : L'id du droit
 - **Niveau_droit** : Le nom du droit
 - **Description_droit** : La description du droit
- **T_Commentaires** : Commentaire sur le ticket
 - **id_commentaire** : L'id du commentaire
 - **Fk_id_ticket** : Clé étrangère avec la table ticket
 - **Fk_id_utilisateur** : Clé étrangère avec la table utilisateur
 - **Date_commentaire** : Date de l'ajout du commentaire
 - **Commentaire** : Commentaires du ticket
- **T_Status** : Les status qu'un ticket peut avoir (Fermé, Ouvert, en cours)
 - **id_status** : L'id du status
 - **Status** : Le nom du status
 - **Description_status** : La description du status
- **T_Impacts** : Les impacts que le problème peut avoir (Utilisateur, Utilisateurs, Départements)
 - **id_impacte** : L'id de l'impact
 - **Impacte** : Le nom de l'impact
 - **Description_impacte** : La description de l'impact
- **T_Priorites** : Les niveaux de la priorité qu'un ticket peut avoir (Urgent, Critique, Bas)
 - **id_priorite** : L'id de la priorité
 - **Priorite** : Le nom de la priorité
 - **Description_priorite** : La description de la priorité
- **T_Documents** : Les documents qu'un ticket peut avoir en annexe
 - **Id_document** : L'id du document
 - **Fk_id_ticket** : Clé étrangère avec la table ticket
 - **Fk_id_utilisateur** : Clé étrangère avec la table utilisateur
 - **Lien_Document** : Le lien du document
- **T_Categories** : Les différentes catégories qu'un ticket peut avoir
 - **Id_categorie** : L'id de la catégorie
 - **Categorie** : Le nom de la catégorie
 - **Fk_categorie** : Clé étrangère avec la table sur elle même
 - **Description_categorie** : Description de la catégorie



Rapport de projet – Groupe 6

Système Ticketing - MPTickets

Langage

HTML

Le Hypertext Markup Language est le langage de balisage conçu pour représenter les pages web. Ce langage est indispensable pour notre projet. Tous les pages du site continent du code HTML.

Nous pouvons trouver dans le code source de la page `Index.php`, et toutes les pages qui se trouvent dans les dossiers body et includes mais aussi dans certaines fonctions.

The screenshot shows the 'Création de ticket' (Ticket Creation) page. The form contains fields for 'Demandeur' (Dylan Guiducci), 'Téléphone Demandeur' (+41 21 544 38 48), 'Département' (Achats), 'Localisation' (Funky Claude's Bar), 'N° du matériel' (PC160065), 'Date du problème' (mm/dd/yyyy), 'Nom du matériel (Demande)' (adsl-84-226-80-150.adsl), 'IP du matériel (Demande)' (84.226.80.150), 'Impact' (Utilisateur - Niveau Ba), 'Priorité' (Basse - Priorité Basse), 'Catégorie' (Informatique/Profil bloqué), and a large 'Description' text area. At the bottom are 'Annuler' and 'Terminer' buttons. The footer includes links to 'Accéder à...', 'Création d'un ticket', 'Suivi d'un ticket', 'Administration', 'Contact' (support.montreux@fairmont.com, Osama Shalhoub - 8004), and the Fairmont Le Montreux Palace logo. The developer tools' Elements tab shows the corresponding HTML structure for the page, including the form and its fields.

Exemple de HTML dans le projet :

```
<table id="tableForm">
    <form id="formulaire" method="post" action="../BD/insertTicket.php">

        <tr>
            <td class="TdCol_1"><label for="Demandeur">*Demandeur : </label></td>
            <td class="TdCol_2"><input required type="text" name="Demandeur" id="Demandeur" placeholder="Ex : Dylan Guiducci"/></td>
            <td class="TdCol_3"><label for="TelDemandeur">*Téléphone Demandeur : </label></td>
            <td class="TdCol_4"><input required type="text" name="TelDemandeur" id="TelDemandeur" placeholder="+41 21 544 38 48"/></td>
        </tr>
        <tr>
            <td class="TdCol_1 Separation-border-bottom"><label for="Departement">*Département : </label></td>
            <td class="TdCol_2 Separation-border-bottom"><select required name="Departement" id="Departement">
                <?php import_Departements(); ?>
                </select>
            </td>
```



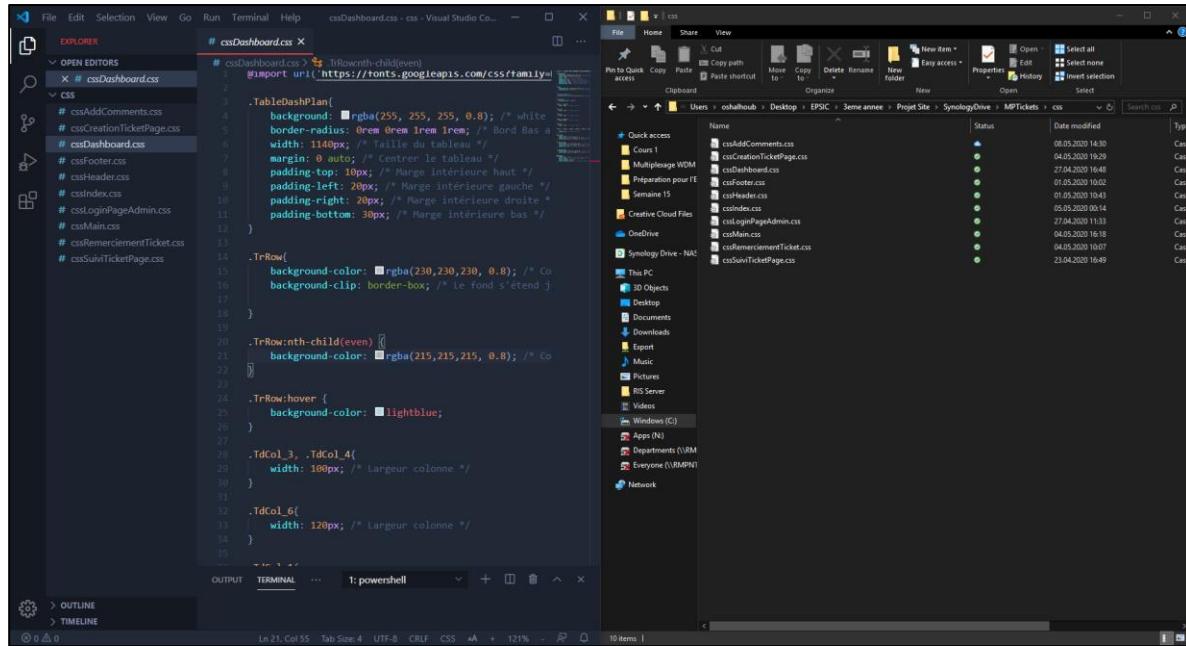
Rapport de projet – Groupe 6

Système Ticketing - MPTickets

CSS

Cascading Style Sheets est un langage de feuille de style. C'est-à-dire qu'il permet d'appliquer des styles sur différents éléments sélectionnés dans un document HTML. Comme l'HTML, ce langage de style est indispensable pour la forme et le désigne des pages du projet. C'est avec [du](#) codes CSS que nous pouvons placer le header, le footer, les formulaires ainsi que les boutons.

Il existe un fichier CSS pour un ou plusieurs pages WEB et ils trouvent tous dans le dossier CSS.



The screenshot shows the Visual Studio Code interface with the CSS dashboard file open. The left sidebar shows a tree view of files, including 'cssDashboard.css' which is currently selected. The main editor area displays the CSS code for the dashboard. A status bar at the bottom indicates the file is 10 items long.

```
# cssDashboard.css
# cssDashboard {
    background-color: #fff;
    border-radius: 0px;
    border: 1px solid #ccc;
    width: 100px;
    margin: 0 auto;
    padding-top: 10px;
    padding-left: 20px;
    padding-right: 20px;
    padding-bottom: 30px;
}

.TrRow {
    background-color: #fff;
    background-clip: border-box;
}

.TrRow:nth-child(even) {
    background-color: #f2f2f2;
}

.TrRow:hover {
    background-color: #e6f2ff;
}

.TdCol_3, .TdCol_4 {
    width: 100px;
}

.TdCol_6 {
    width: 120px;
}
```

Exemple de CSS dans le projet :

```
input{
    background-
    color: white; /* Couleur de Fond Blanche pour les Inputs désactivés */
    color: black; /* Couleur de la police Noir pour les Inputs désactivés */
    padding-left: 3px; /*Marge intérieure gauche */
}

.Formulaire{
    background: rgba(255, 255, 255, 0.8); /* white background with 80% opacity
    width: 1000px; /* Largeur du tableau */
    margin: 0 auto; /* Centrer le tableau */
    padding-top: 10px; /* Marge intérieure haut */
    padding-left: 20px; /* Marge intérieure gauche */
    padding-right: 20px; /* Marge intérieure droite */
    padding-bottom: 27px; /* Marge intérieure bas */
}

.TdCol_1, .TdCol_2, .TdCol_3, .TdCol_4{
    width: 250px; /* Largeur colonne */
    padding: 0; /* Marge intérieure */
    padding-bottom: 4px;
```



Rapport de projet – Groupe 6

Système Ticketing - MPTickets

PHP

Hypertext Preprocessor est un langage de programmation côté serveur conçu pour les pages dynamiques. Le langage PHP nous permet de créer des fonctions et des scripts. [Nous utilisons ce langage pour communiquer avec la base de données et traiter les informations des formulaires.](#)

Le fichier FonctionPHP-SQL.php dans le dossier BD contient toutes les fonctions PHP du site, comme la connexion à la base de données, le login, l'importation des tickets, l'importation des départements et même la création du l'ID du ticket.

```
<?php
// Connexion au serveur
$link=mysqli_connect($dbhost, $dbuser, $dbpass) or die("Erreur de connexion au serveur base de données ." | ".$dbhost." | ".$dbuser." | ".$dbpass);

mysqli_select_db($link, $dbname) or die("Base de données pas trouvée");

?>
```

Exemple de PHP dans le projet :

```
<?php

$dbname='MPTicket';
$dbuser='MPTicket';
$dbpass='R0ge1r0-M0dule151';
$dbhost='localhost';

// Connexion au serveur
$link=mysqli_connect($dbhost, $dbuser, $dbpass) or die("Erreur de connexion au serveur base de données ." | ".$dbhost." | ".$dbuser." | ".$dbpass);

mysqli_select_db($link, $dbname) or die("Base de données pas trouvée");

?>
```

```
<?php
// Deconnexion au serveur
    mysqli_close($link);
?>
```



Rapport de projet – Groupe 6

Système Ticketing - MPTickets

SQL

Structured Query Language est un langage informatique normalisé servant à exploiter des bases de données relationnelles. La partie langage de manipulation des données de **SQL permet de rechercher, d'ajouter, de modifier ou de supprimer des données dans les bases de données**. Le dossier BD à la racine contient tous les fichiers qui communiquent avec la base de données. Comme la création d'un ticket, ajouter un commentaire ou changer l'état du ticket.

The screenshot shows a Visual Studio Code interface. On the left is the Explorer sidebar with several PHP files listed under 'OPENEDERS'. The main editor window displays the code for 'updateTicket.php'. The code is a PHP script that interacts with a MySQL database to update a ticket record. It includes logic for setting session variables, preparing SQL queries, and handling exceptions. On the right, a file explorer window shows a directory structure with many PHP files, indicating a large project. The file names include 'updateTicket.php', 'editComment.php', 'addTicket.php', 'insertTicket.php', 'login.php', 'resolveTicket.php', 'sortTicketDashboard.php', 'suspendTicket.php', and 'FunctionPHP_SQL.php'. The file explorer also shows other files like 'index.php', 'style.css', and 'script.js'.

Exemple de SQL dans le projet :

```
INSERT INTO T_Tickets (id_ticket, Utilisateur_demandeur, Numéro_tel_demandeur, fk_id_status, fk_id_impact, fk_id_priorite, fk_id_departement, Date_Probleme, IP, Nom_machine_creation, Nom_machine_probleme, Localisation, Description_ticket)
VALUES ('".$id_ticket."', '".$Utilisateur_demandeur."', '".$Numéro_tel_demandeur."', '".$fk_id_status."', '".$fk_id_impact."', '".$fk_id_priorite."', '".$fk_id_departement."', '".$Date_Probleme."', '".$IP."', '".$Nom_machine_creation."', '".$Nom_machine_probleme."', '".$Localisation."', '".$Description_ticket."');
```

JavaScript

JavaScript est un langage de programmation de scripts principalement employé dans les pages web interactives, il n'est presque pas du tout utilisé dans le projet.

Le bouton « Retour » dans la page qui permet d'ajouter un commentaire addComments.php, utilise ce langage. Le fonction « history.back() » permet de faire un retour selon l'historique de l'utilisateur.

Exemple de JavaScript dans le projet :

```
$(function () {
    $('.dropdown-menu a.dropdown-toggle').on('click', function (e) {
        if (!$(this).next().hasClass('show')) {
```



Rapport de projet – Groupe 6

Système Ticketing - MPTickets

```
$(this).parents('.dropdown-menu').first().find('.show').removeClass("show");
}
var $subMenu = $(this).next(".dropdown-menu");
$subMenu.toggleClass('show'); // appliqu  au ul
$(this).parent().toggleClass('show'); // appliqu  au li parent

$(this).parents('li.nav-item.dropdown.show').on('hidden.bs.dropdown', function (e) {
    $('.dropdown-submenu .show').removeClass('show'); // appliqu  au ul
    $('.dropdown-submenu.show').removeClass('show'); // appliqu  au li parent
});
return false;
});

$(document).ready(function () {
    $('.radio').click(function () {
        $('.radio').not(this).prop('checked', false);
    });
});
```

Bootstrap

Bootstrap est une collection d'outils utiles à la création du design de sites et d'applications web. C'est un ensemble qui contient des codes CSS et des éléments interactifs, ainsi que des extensions JavaScript en option.

Afin d'améliorer le désigne du site Web, nous avons utilisé le Bootstrap sur quelques pages web.

Importation du code Bootstrap dans le header :

```
<!-- Ajout de Bootstrap -->
    <!-- Latest compiled and minified CSS -->
    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.0/css/bootstrap.min.css">

    <!--Bootstrap 4 CDN-->
    <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.1.3/css/bootstrap.min.css" integrity="sha384-MCw98/SFnGE8fJT3GXwEOngsV7Zt27NXFoaoApmYm81iuXoPkFOJwJ8ERdknLPMO" crossorigin="anonymous">

    <!--Fontawesome CDN-->
    <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.3.1/css/all.css" integrity="sha384-mzrmE5qonljUremFsqc01SB46JvROS7bZs3IO2EmfFsd15uHvIt+Y8vEf7N7fWAU" crossorigin="anonymous">
```



Rapport de projet – Groupe 6

Système Ticketing - MPTickets

Softwares

Visual Studio Code

Un éditeur de texte est indispensable pour le développement. Le membre Osama utilise Visual Studio. Visual Studio permet d'afficher tout le répertoire de travail avec les différents fichiers visible en un clin d'œil. Cela permet de changer de fichier très simplement. Il a aussi une aide lors de la création des fichiers qui est utile lors du développement des pages Web.

```
c:\Users\oshalhoub\Desktop\EPISC\3eme annee>Projet Site>SynologyDrive>MPTickets>body>creationTicketPage.php
1 lude($_SERVER["DOCUMENT_ROOT"]."/MPTickets/includes/header.php");
2
3
4 <!-- Stylesheet href="/MPTickets/css/cssCreationTicketPage.css"/>
5
6
7 <? FirstPlan >
8 <? class="Formulaire" >
9 <? h1><center>Création de ticket</center><? /h1>
10
11 <!-- Création du formulaire en format Tableau -->
12 <? table id="tableForm" >
13 <? form id="formulaire" method="post" action="../BD/InsertTicket.php" >
14 <? tr >
15 <? td class="IdCol_1" ><label for="Demandeur">Demandeur : </label><? /td>
16 <? td class="IdCol_2" ><input required type="text" name="Demandeur" id="Demandeur" ><? /td>
17 <? td class="IdCol_3" ><label for="TelDemandeur">*Téléphone Demandeur : </label>
18 <? td class="IdCol_4" ><input required type="text" name="TelDemandeur" id="TelDemandeur" ><? /td>
19 <? /tr>
20 <? tr >
21 <? td class="IdCol_1 Separation-border-bottom" ><label for="Departement">Département : </label><? /td>
22 <? td class="IdCol_2 Separation-border-bottom" ><select required name="Departement" ><?php import_Departements(); ?>
23 <? /select><? /td>
24 <? td class="IdCol_3 Separation-border-bottom" ><label for="Localisation">*Localisation : </label>
25 <? td class="IdCol_4 Separation-border-bottom" ><input required type="text" name="Localisation" id="Localisation" ><? /td>
26 <? /tr>
27 <? tr >
28 <? td class="IdCol_1 Separation-border-top" ><label for="NoMatériel">*N° du matériel : </label>
29 <? td class="IdCol_2 Separation-border-top" ><input required type="text" name="NoMatériel" id="NoMatériel" ><? /td>
30 <? td class="IdCol_3 Separation-border-top" ><label for="DateProbleme">Date du problème : </label>
31 <? td class="IdCol_4 Separation-border-top" ><input required type="date" name="DateProbleme" id="DateProbleme" ><? /td>
32 <? /tr>
33 <? /table >
34 <?php
35 // Récupérer l'adresse IP du matériel qui a fait la requête
36 $ip = $_SERVER["REMOTE_ADDR"];
37 $host = gethostbyaddr($ip);
38
39 ><?tr >
40 <?td class="IdCol_1" ><label for="NomPcDemande">Nom du matériel (Demande) : </label>
41 <?td class="IdCol_2" ><input disabled type="text" name="NomPcDemande" id="NomPcDemande" value="$host" ><? /td>
42 <?td class="IdCol_3" ><label for="IpMatérielDemande">IP du matériel (Demande) : </label>
43 <?td class="IdCol_4" ><input type="text" name="IpMatérielDemande" id="IpMatérielDemande" value="$ip" ><? /td>
44 <? /tr>
45
46 <?php
47 include($_SERVER["DOCUMENT_ROOT"]."/MPTickets/includes/footer.php");
48 >
```

NotePad++

Le membre Dylan utilise plutôt l'éditeur NotePad++ ainsi que Visual Studio Entreprise cela dépend de la machine qu'il utilise. NotePad++ est un éditeur de texte simple et efficace mais n'a pas tous les avantages des Visual Studio.

```
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?  
creationTicketPage.php           
  
creationTicketPage.php           
  
6  
7 <FirstPlan>  
8 <ss="Formulaire">  
9 <center>Création de ticket</center></h1>  
10 <br>Création du formulaire en format Tableau -->  
11 <table id="tableForm">  
12 <form id="formulaire" method="post" action="../../../BD/insertTicket.php">  
13 <tr>  
14 <td class="TdCol_1"><label for="Demandeur">Demandeur : </label></td>  
15 <td class="TdCol_2"><input required type="text" name="Demandeur" id="Demandeur" placeholder="Ex : Dylan Guiducci"/></td>  
16 <td class="TdCol_3"><label for="TlDemandeur">+Téléphone Demandeur : </label></td>  
17 <td class="TdCol_4"><input required type="text" name="TlDemandeur" id="TlDemandeur" placeholder="+41 21 544 38 48"/></td>  
18 </tr>  
19 <tr>  
20 <td class="TdCol_1 Separation-border-bottom"><label for="Departement">*Département : </label></td>  
21 <td class="TdCol_2 Separation-border-bottom"><select required name="Departement" id="Departement">  
22 <?php import_Departements(); ?>  
23 </select>  
24 </td>  
25 <td class="TdCol_3 Separation-border-bottom"><label for="Localisation">*Localisation : </label></td>  
26 <td class="TdCol_4 Separation-border-bottom"><input required type="text" name="Localisation" id="Localisation" placeholder="Ex : Funky Claude's Bar"/></td>  
27 </tr>  
28 <tr>  
29 <td class="TdCol_1 Separation-border-top"><label for="NoMateriel">*N° du matériel : </label></td>  
30 <td class="TdCol_2 Separation-border-top"><input required type="text" name="NoMateriel" id="NoMateriel" placeholder="Ex : PC160065"/></td>  
31 <td class="TdCol_3 Separation-border-top"><label for="DateProbleme">*Date du problème : </label></td>  
32 <td class="TdCol_4 Separation-border-top"><input required type="datetime-local" name="DateProbleme" id="DateProbleme"/></td>  
33 </tr>  
34 <?php  
35 // Recuperer l'adresse IP du matériel qui a fait la requête  
36 $ip = $_SERVER['REMOTE_ADDR'];  
37 $short = gethostbyaddr($ip);  
38 ?>  
39 <tr>  
40 <td class="TdCol_1"><label for="NomPcDemande">Nom du matériel (Demande) : </label></td>  
41 <td class="TdCol_2"><input disabled type="text" name="NomPcDemande" id="NomPcDemande" value="<?php echo $short ?>"/></td>  
42 <td class="TdCol_3"><label for="IpMaterielDemande">IP du matériel (Demande) : </label></td>  
43 <td class="TdCol_4"><input disabled type="text" name="IpMaterielDemande" id="IpMaterielDemande" value="<?php echo $ip ?>"/></td>  
44 </tr>  
45 <tr>  
46 <td class="TdCol_1"><label for="Impact">*Impact : </label></td>  
47 <td class="TdCol_2"><select required name="Impact" id="Impact">  
48 <?php include 'Impact.php'; ?>  
49 </select></td>
```



Rapport de projet – Groupe 6

Système Ticketing - MPTickets

Skype4Business

Malheureusement, pour le membre Osama, Discord est bloqué par le pare-feu de l'entreprise Montreux Palace pour des questions de sécurité. Donc une solution intervalle à Discord pour les conférences lors du travail, est Skype4Business. **C'est une version professionnelle de Skype. Il permet de faire des partages d'écran mais aussi de prise contrôle à distance du PC.**

The screenshot shows the Skype for Business interface. On the left, a chat window with Dylan Guiducci is displayed, showing messages like "Il fonctionne ton GitHub ?" and "GitHub est mort". On the right, a list of missed calls is shown, with entries from Dylan Guiducci, Kumar, Akshay, and others, all dated 23.04.2020. The interface includes standard Skype controls for video, audio, and screen sharing at the bottom.

Discord

L'utilisation du logiciel Discord permet de faire des conférences pour le suivi du projet. Discord propose des options comme partage d'écran et des conférences de plusieurs personnes.

Faires des conférences entre les membres du projet et le professeur permettent d'avoir des retours, de l'aide sur notre projet.

The screenshot shows the Discord application interface. On the left, a sidebar lists various servers and users. The main area shows a group chat for "Dylan Guiducci, RogeiroA" with a message from Dylan Guiducci adding RogeiroA to the group. The interface includes a search bar at the top and standard Discord controls at the bottom.



Rapport de projet – Groupe 6

Système Ticketing - MPTickets

WhatsApp

Pour la communication des informations concernant le projet comme les accès, les liens, nous utilisons un groupe WhatsApp. C'est vachement pratique vu que nous pouvons accéder puis l'ordinateur et le téléphone.

The screenshot shows a WhatsApp group named "Projet - MPTickets". The messages include:

- Dylan Gi: "tu veux pas genre l'afficher dans une création mais on dit que le gars ne peut pas le changer" (10:22)
- Dylan Gi: "Tu as pris quelle genre de drogue 😂😂" (10:22)
- Dylan Gi: "il prend automatiquement la date de mnt" (10:22)
- Aurélien: "mec je bosse sur ve truc depuis 8 heure" (10:23)
- Aurélien: "en fait du coup on change le type des champs fk à varchar ? pour bénéficiaire et demandeur ?" (10:23)
- Aurélien: "voilà à quoi genre ressemble le dashboard" (10:24) - Attached a screenshot of the MPTickets application dashboard.
- Dylan Gi: "Il y a le champ date du problème mais après c'est vrai qu'on peut mettre date de la création du ticket" (10:24)

The right sidebar shows group information and links to various system access points like phpMyAdmin and MariaDB.

Trello

On utilise Trello comme solution pour distribuer et lister les tâches à faire concernant l'avancement du projet. C'est une solution gratuite et efficace pour la gestion de projet et qui permet de savoir les tâches restantes avant la fin du projet.

The Trello board has several lists:

- A faire:
 - Nassi-Schnedermann - Arborisation de fichiers
 - Nassi-Schnedermann - fonctionnement de l'application Web
 - Rapport - Insérer le code des pages dans le rapport (chaque code chaque page)
 - HTML, CSS PHP - Faire le remember me sur la page Administration login
 - Rapport - Partie sql base de données (explication de chaque fichier)
 - HTML, CSS PHP - Créer un reCAPTCHA sur la page Login Administration
 - HTML, CSS PHP - Créer un reCAPTCHA sur la page Crédit de ticket
- Travail de la semaine | jeudi - mercredi:
 - Rapport - Modifier la partie base de données dans le rapport
 - GANTT
 - HTML, CSS PHP - Développer la page ManageTicketPage.php
 - HTML, CSS PHP - Modifier la page "Suivi d'un ticket" après la modification sur la page avec Dylan
 - HTML, CSS PHP - Développer la page ViewTicket.php
- En travail:
 - Ajouter des commentaires dans le code
 - SQL - Codage pour faire en sorte de créer un ticket dans la BD via WEB sql
- A mettre dans le Rapport:
 - Script de sauvegarde - NAS
 - Ajouter une autre carte
- Vérifier:
 - Footer
 - Header
 - A vérifier plus tard
 - MCD et MLD - Modifier la table T_Tickets et la relation entre T_Users
 - Rapport - Partie Software sur Synology Drive Client
 - Rapport - Faire la partie Partage de fichiers
 - Modifier le formulaire de la page création d'un ticket
 - Modifier dans le formulaire de création d'un ticket, dans l'impact, il faudra mettre l'utilisateur, Utilisateur, Entreprise, Désarçlement
- Terminer:
 - MCD
 - MLD
 - Modification GANTT
 - Faire Trello
 - Créer Design page Dashboard
 - Ajouter le projet dans un serveur local MPTickets
 - Connexion User
 - Corriger Rapport - base de données
 - Test connexion avec la base de données - Log in Administration
 - Remplir table base de données
 - Connexion BD - Suivi de ticket
 - Connecter le site avec la base de données



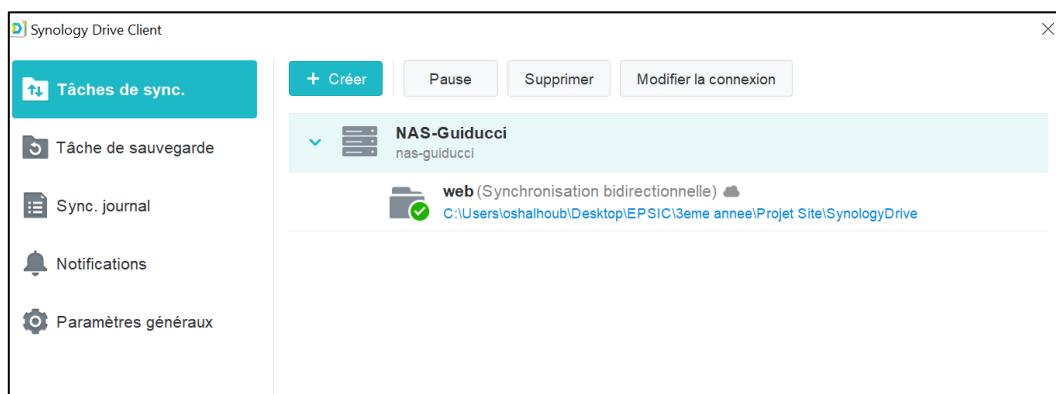
Rapport de projet – Groupe 6

Système Ticketing - MPTickets

Synology Drive Client

Nous utilisons un NAS Synology pour stocker le site WEB MPTickets. Pour faire la synchronisation avec le NAS, nous utilisons le logiciel Synology Drive Client. Il crée un dossier sur le PC où les fichiers du site WEB sont stockés. Pour pouvoir travailler, sur les fichiers nous avons qu'à simplement double-cliquer et il va télécharger le fichier. Ce système permet d'économiser de l'espace sur le poste en cas de gros fichiers. Les synchronisations se font automatiquement dès qu'un fichier a été modifié. Cela permet d'éviter d'oublier des faire une synchronisation une fois le travail effectué. De plus cela est pratique lors de la modification des fichiers PHP, le site WEB change quasiment instantanément.

Nous avons créé des sauvegardes automatiques grâce au logiciel de sauvegarde créé par Synology. Cela permet d'éviter les oubli lors des sauvegardes manuelles.



Serveur Local XAMP

Le serveur Local utilisé par chaque membre est **XAMP Version 3**. Le serveur Local XAMP va permettre de faire la connexion entre l'Application Web et la base de données.

Cela permettra aussi aux membres du projet de faire leur test en local. Mais cette solution, n'a été utilisée qu'une semaine lors de la connexion à la base de données car nous avions eu un problème avec le NAS. Le détail du problème se trouve dans les problèmes rencontrés.

The screenshot shows two windows side-by-side. On the left is a web browser window titled 'MP.Tickets' showing the homepage. On the right is the 'XAMPP Control Panel v3.2.4' window, which lists the following services:

Service	Module	PID(s)	Port(s)	Actions
Apache		5316 13580	80, 443	Stop Admin Config Logs
MySQL		9344	3306	Stop Admin Config Logs
FileZilla				Start Admin Config Logs
Mercury				Start Admin Config Logs
Tomcat				Start Admin Config Logs

Below the table, there is a log of messages from the Apache service:

```
19:56:27 [main] Initializing Control Panel
19:56:27 [main] Windows Version: Enterprise 64-bit
19:56:27 [main] XAMPP Version: 7.4.2
19:56:27 [main] Control Panel Version: 3.2.4 [ Compiled: Jun 5th 2019 ]
19:56:27 [main] Running with Administrator rights - good!
19:56:27 [main] XAMPP Installation Directory: "c:\xampp\"
19:56:27 [main] Checking for prerequisites
19:56:27 [main] All prerequisites found
19:56:27 [main] Initializing Modules
19:56:27 [Apache] XAMPP Apache is already running on port 80
19:56:27 [Apache] XAMPP Apache is already running on port 443
19:56:27 [mysql] XAMPP MySQL is already running on port 3306
19:56:27 [main] Enabling autostart for module "Apache"
19:56:27 [main] Enabling autostart for module "MySQL"
19:56:27 [main] Starting Check-Timer
19:56:27 [main] Control Panel Ready
19:56:28 [Apache] Autostart aborted: Apache is already running
19:56:28 [mysql] Autostart aborted: MySQL is already running
```



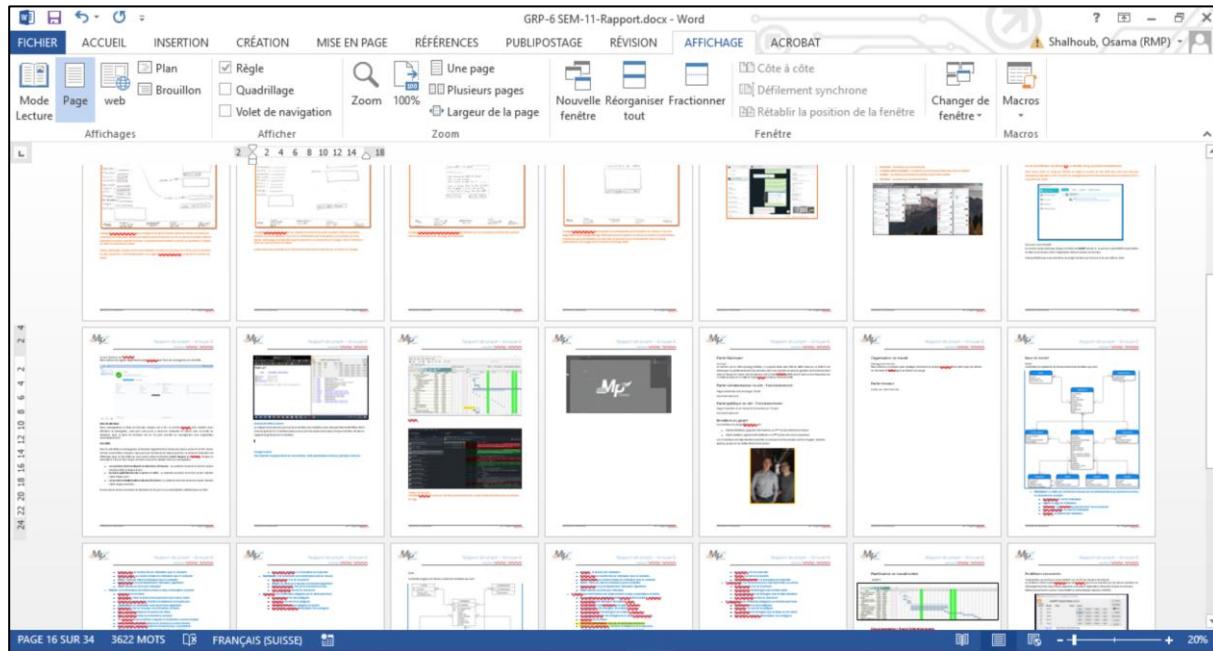
Rapport de projet – Groupe 6

Système Ticketing - MPTickets

Microsoft Office Word

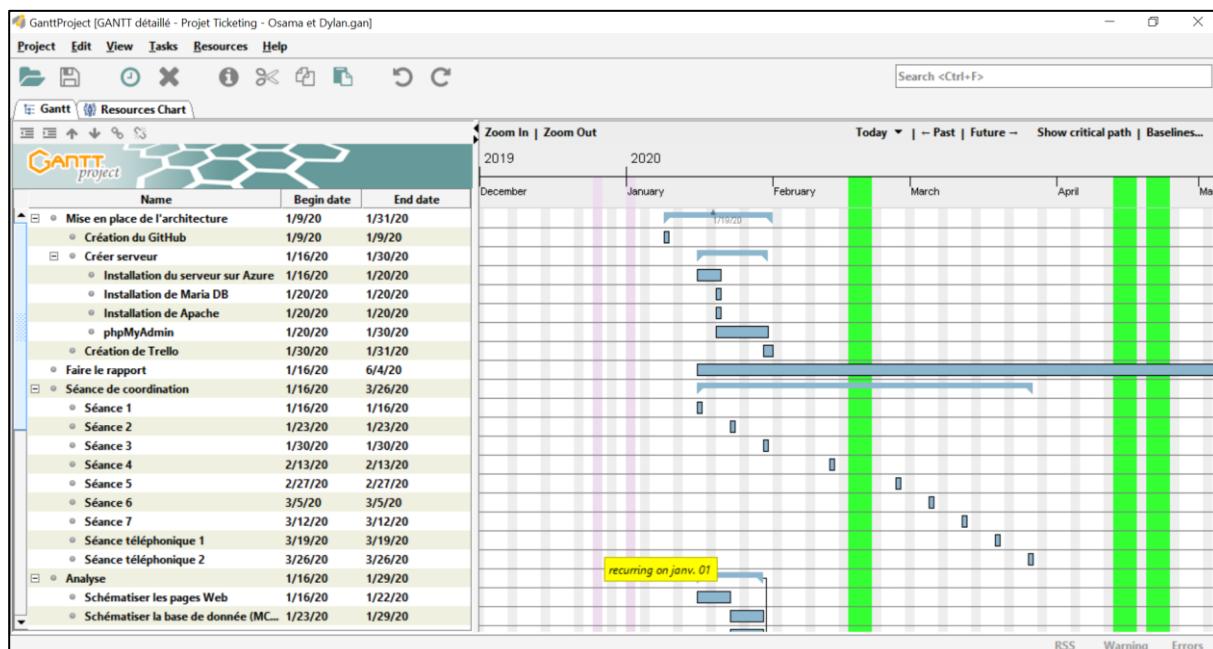
Le rapport de projet et le journal de la semaine des membres sont créés par Microsoft Office Word.

Le journal est personnel donc chaque membre s'occupe de son fichier Puis il permet de lister les tâches faites par l'utilisateur et le temps qu'il a passé sur celle-ci.



Gantt

Pour planifier le projet durant les 21 semaines. Cette planification est tenue par Dylan Guiducci en utilisant le logiciel Gantt. Il est super pratique pour tout ce qui est planification des projets sur des longs durées, comme ce qui est le cas pour MPTickets.





Rapport de projet – Groupe 6

Système Ticketing - MPTickets

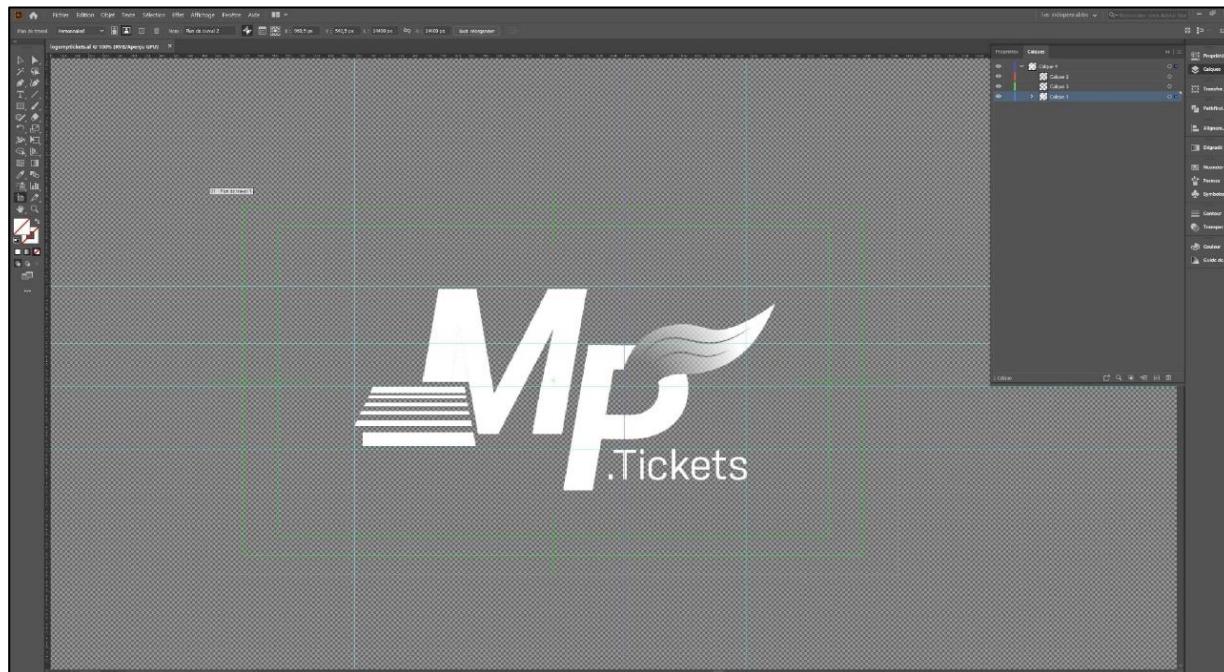
GitHub Desktop

Pour partager les fichiers entre les membres, GitHub Desktop est la meilleure solution. Les fichiers sont synchronisés sur un dossier spécifique où les deux membres ont accès aux fichiers. Il suffit juste de faire Fetch Origin afin de voir s'il y a des documents à télécharger, Push pour uploader les documents et Pull pour télécharger.

The screenshot shows the GitHub Desktop application interface. On the left, the repository 'Projet_Ticketing' is selected. The main area displays a merge commit from 'HeroChan7' (4da80b3) on 'origin/master' into the local 'master' branch. The commit message is 'Merge branch 'master' of https://github.com/DydouGui/Projet_Ticketing'. The commit details show changes to three files: 'Base de donn...\\MLD_MPTicket.drawio', 'Base de données\\MLD_MPTicket.jpg', and 'Base de données\\MLD_MPTicket.pdf'. The code diff view shows the XML content of the drawio file, comparing the local version (green) with the fetched version (red). The diff highlights differences in node positions and attributes.

Adobe Illustrator

Aurélien Kollbrunner est un ami au membre Osama Shalhoub. Nous l'avons contacté pour la création du logo MPTickets. Il a utilisé Adobe Illustrator pour le créer.





Rapport de projet – Groupe 6

Système Ticketing - MPTickets

PhpMyAdmin

Nous utilisons l'application Web phpMyAdmin pour la gestion de base de données MySQL. Cela permet d'avoir une vue sur les tables et les données.

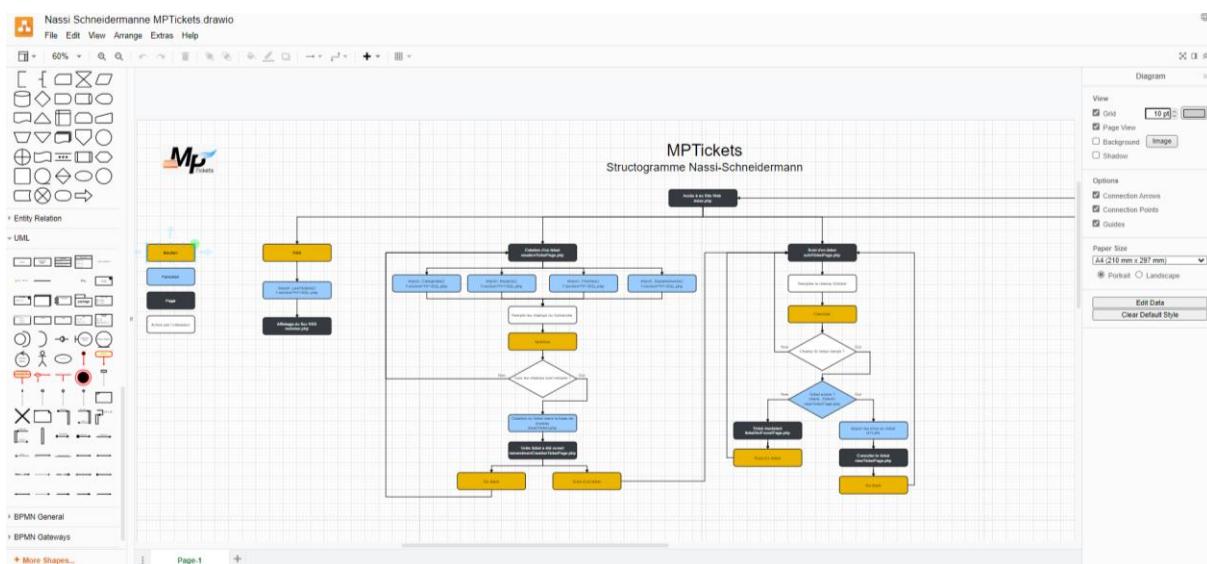
C'est super pratique parce qu'on peut facilement ajouter/modifier/supprimer des données ou même des tables.

The screenshot shows the PhpMyAdmin interface for the 'MPTickets' database. The left sidebar lists the database schema with various tables such as T_Categories, T_Commentaires, T_Documents, T_Droits, T_Impacts, T_Priorites, T_Status, T_Tickets, and T_Utilisateurs. The main area displays a table of all 10 tables, showing columns for Table, Action, Rows, Type, Collation, Size, and Overhead. Below this, there are tabs for Structure, SQL, Search, Query, Import, Operations, Privileges, Routines, Events, Triggers, Tracking, Designer, and Central columns. A 'Create table' form is also visible at the bottom.

Draw.io

Afin de réaliser l'arborescence des fichiers du site Web et le structogramme de Nassi-Schneidermann, nous avons utilisé l'application web Draw.io.

C'est une application gratuite en ligne qui permet de dessiner des diagrammes ou des organigrammes.



Logiciels utilisé pour l'hébergement **Web** sur le NAS

Plusieurs logiciels sont installés et utilisés pour l'hébergement du site Web sur le NAS.

On trouve :



Web Station : Solution pour l'hébergement des sites Web. Son rôle est de gérer les éléments.



Maria DB : Système de gestion de base de données. Fait le rôle du Serveur de base de données dans le



PHP 7.3 : Langage de programmation pour les sites dynamiques. Langage installée sur le serveur (NAS).

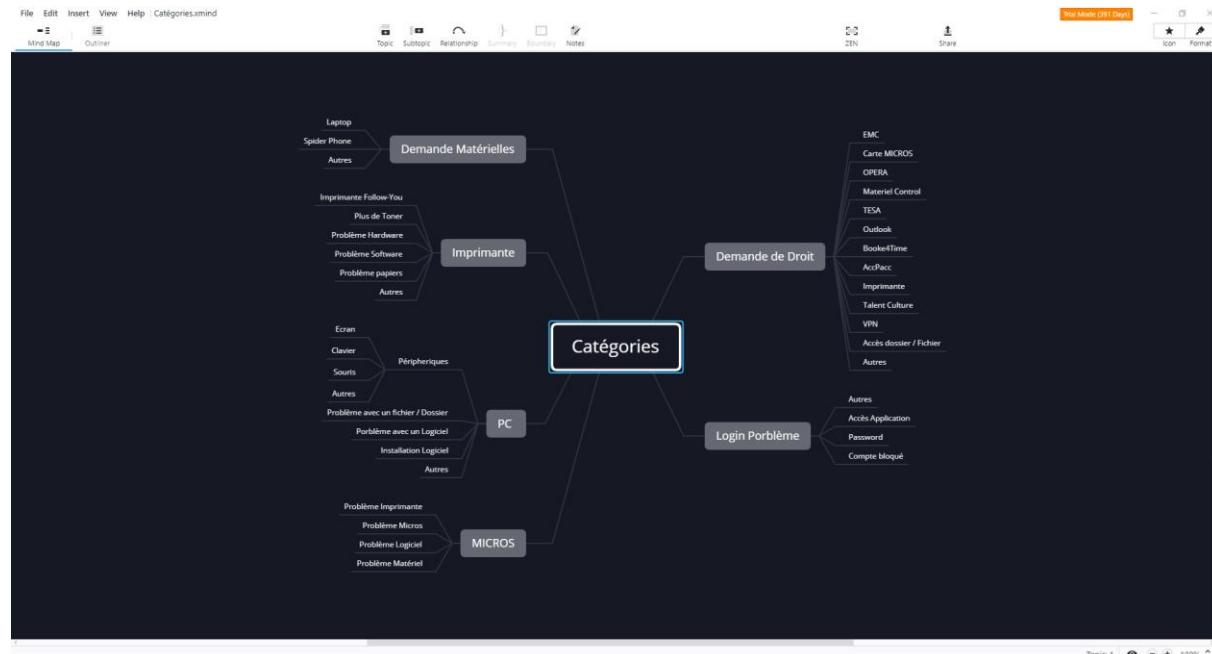


Apache 2 : Serveur http et son rôle dans le projet est le serveur Web.

XMind ZEN

XMind ZEN est un logiciel aussi de Min Mappage comme draw.io mais en mode logiciel. Nous avons utilisé ce logiciel uniquement pour la création des catégories. C'est un logiciel gratuit mais lors de l'exportation du Map en PDF, il y a le Filigrane de XMind, vu qu'il est gratuit.

Nous conseillons plutôt l'application Web Draw.io





Rapport de projet – Groupe 6

Système Ticketing - MPTickets

Hyper Backup de Synology

Nous utilisons le logiciel Hyper Backup de Synology pour faire les sauvegardes du site WEB.

Base de données

Nous sauvegardons la base de données chaque soir à 3h. Le service MariaDB doit s'arrêter pour effectuer la sauvegarde. C'est pour cela qu'on a choisi de l'exécuter en pleine nuit. La durée de rétention pour la base de données est de 11 jours ensuite les sauvegardes sont supprimées automatiquement.

Site WEB

Pour le site WEB, les sauvegardes se font plus régulièrement, toutes les heures entre 7h et 17h. Aucun service n'est arrêté, c'est pour cela que nous les faisons en pleine journée. La durée de rétention est différente pour le site WEB car nous avons utilisé la fonction **Smart Recycle** de Synology. D'après la description c'est un bon moyen de faire une bonne rotation dans les sauvegardes.

- **Les versions horaires depuis les dernières 24 heures** : Le système conserve la version la plus récente créée à chaque heure.
- **Versions quotidiennes de la veille à 1 mois** : Le système conserve la version la plus récente créée chaque jour.
- **Les versions hebdomadaires de plus d'un mois** : Le système conserve la version la plus récente créée chaque semaine.

Et nous avons choisi une durée de rétention de 19 jours ce qui devrait être suffisant pour du DEV.

The screenshot shows the Synology Hyper Backup interface. At the top, there's a green checkmark icon and the word "Succès". Below it, the message "Dernière sauvegarde réussie: 2020-03-31 11:00" and "Heure de la prochaine sauvegarde programmée: 2020-03-31 12:00". On the left, a sidebar lists backup jobs: "Backup_MPTickets... BD_job" and "Backup_MPTickets... WEB_job", both with green checkmarks. The main panel has two sections: "Cible" (Target) and "Paramètres de tâche" (Task Settings). In the "Cible" section, it shows "Dossier partagé: Backup_MPTickets", "Répertoire: Administrateur", "Taille: 0,34 MB", and "Vérification de l'intégrité: Pas encore effectué". In the "Paramètres de tâche" section, it shows "Volume: web", "Dossier partagé: Administrateur", "Filtres de fichier: Aucune", "Application: Aucune", and "Calendrier de sauvegarde: Heure: 07:00 ~ 17:00 tous les 1 heure(s) Intervalle: Quotidienne".

Hardware

Serveur

[Le NAS du membre Dylan est utilisé comme serveur Web pour héberger le projet.](#) C'est un NAS



Synology DS918+, il a quatre baies avec 4TB de HDD chacune. Le RAID 5 est utilisé pour le partitionnement des données, cela nous permet de pouvoir garantir le fonctionnement avec un disque en moins. Son processeur est un Intel Celeron J3455 avec 4 coeurs à une fréquence de 1.5 GHz de base et 2.3 GHz en mode brust et 4 GB de mémoire RAM.

Membres et aide à la réalisation du projet



Osama Shalhoub

Apprenti informaticien d'entreprise en 3^{ème} année chez Montreux Palace.

Membre du Projet

[LinkedIn](#)



Dylan Guiducci

Apprenti informaticien d'entreprise en 3^{ème} année chez Assura Assurance.

Membre du Projet

[LinkedIn](#)



Angelo Rogeiro

Professeur informatique à l'EPSIC

Superviseur du Projet

[LinkedIn](#)



Aurelien Kollrunner

Ingénieur HES des médias

Aide à la réalisation du projet : Design, Développement pages Web

[Web](#)



Ali Alshweiki

Développeur Web Master HES-SO

Aide à la réalisation du projet : Base de données, Développement pages Web

[LinkedIn](#)

Les membres Osama et Dylan ont déjà travaillé ensemble sur la réalisation de plusieurs projets comme :

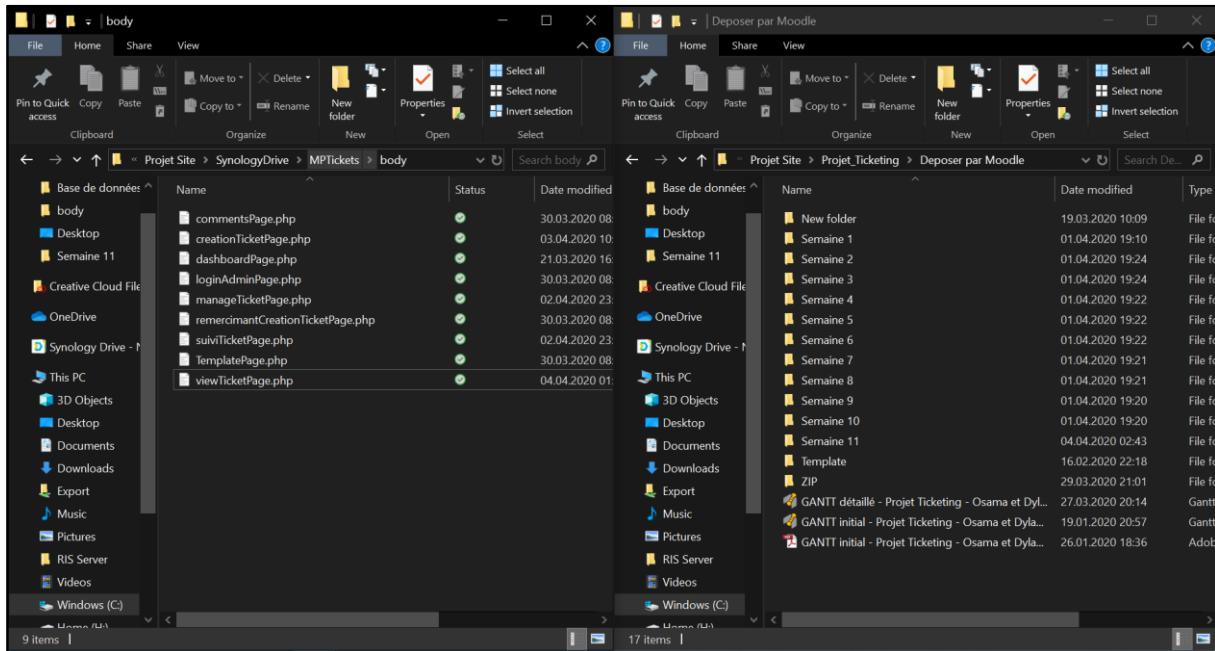
- Système Backup – Module 143
- Câble Ethernet – Module 214
- Gestion de projet d'une entreprise – Module 306
- Comptabilité – Suicide d'un paysan
- Économie – L'environnement politico-égal de l'entreprise
- ECG – Comment expliquer que la musique est aussi importante dans notre quotidien ?
- Anglais – NAS or Cloud
- Anglais – Open question why people Hack
- Science – Liaison sans fil

Organisation du travail – Partage de fichiers

Nous utilisons **deux** solutions pour partager les fichiers du projet. Synology Drive Client pour les fichiers du site Web et GitHub Desktop pour les fichiers du projet.

Synology Drive Client : La solution pour partager les fichiers de source du Site Web.

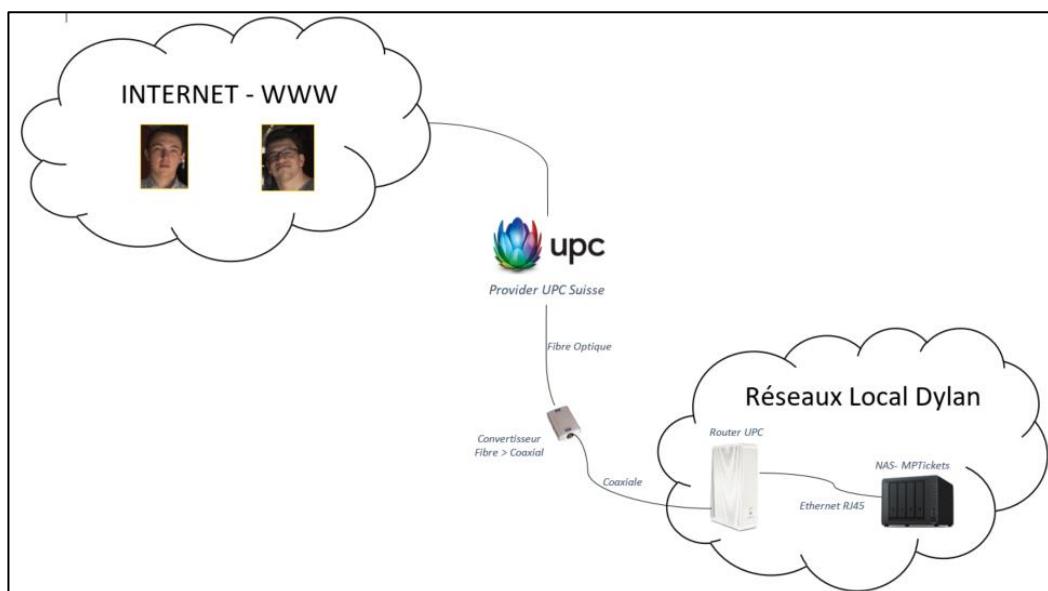
GitHub Desktop : La solution pour partager les fichiers concernant l'organisation du projet comme le journal, le rapport, et autre.



Partie réseaux

Accès au site Internet – Développement du site Web

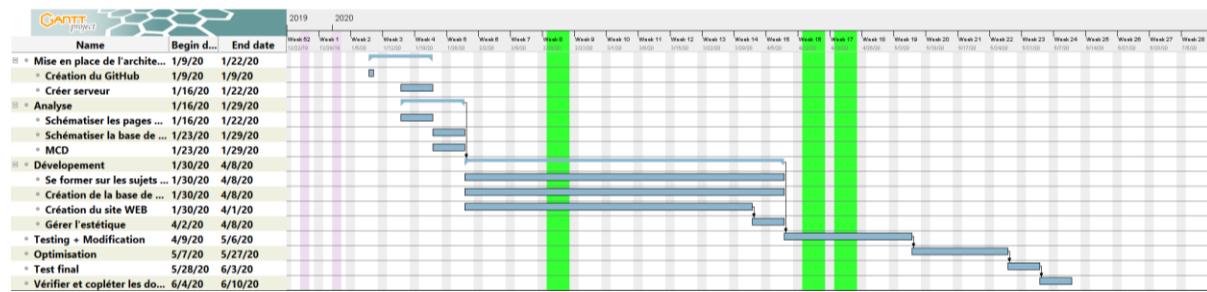
Le serveur Web (NAS) utilise l'adresse IP publique 84.72.82.215 fournit par le provider UPC Suisse Sàrl pour que les membres du projet puissent se connecter de n'importe où et voir le résultat du travail qui a été fait sur les pages Web.



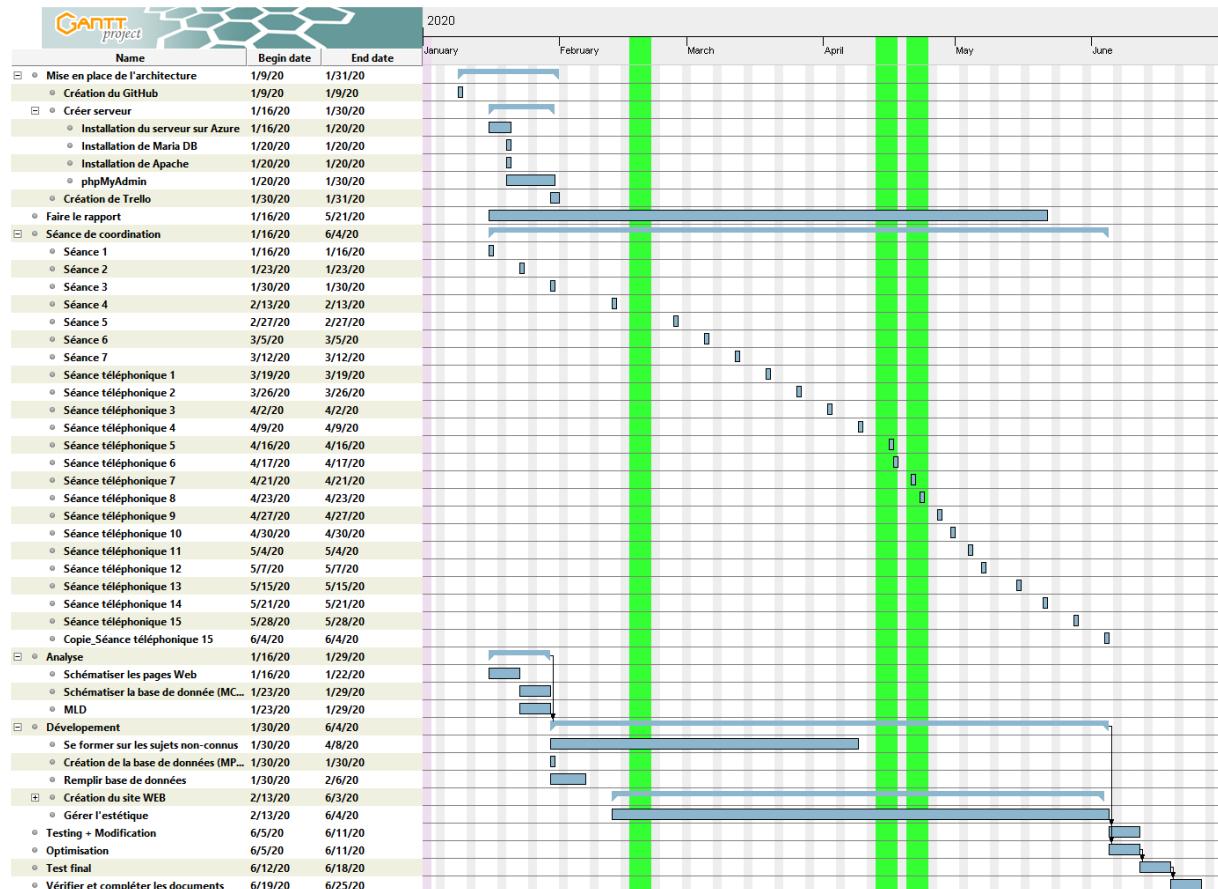
Accès au site Internet – Site Web prête à utiliser

Planification du travail

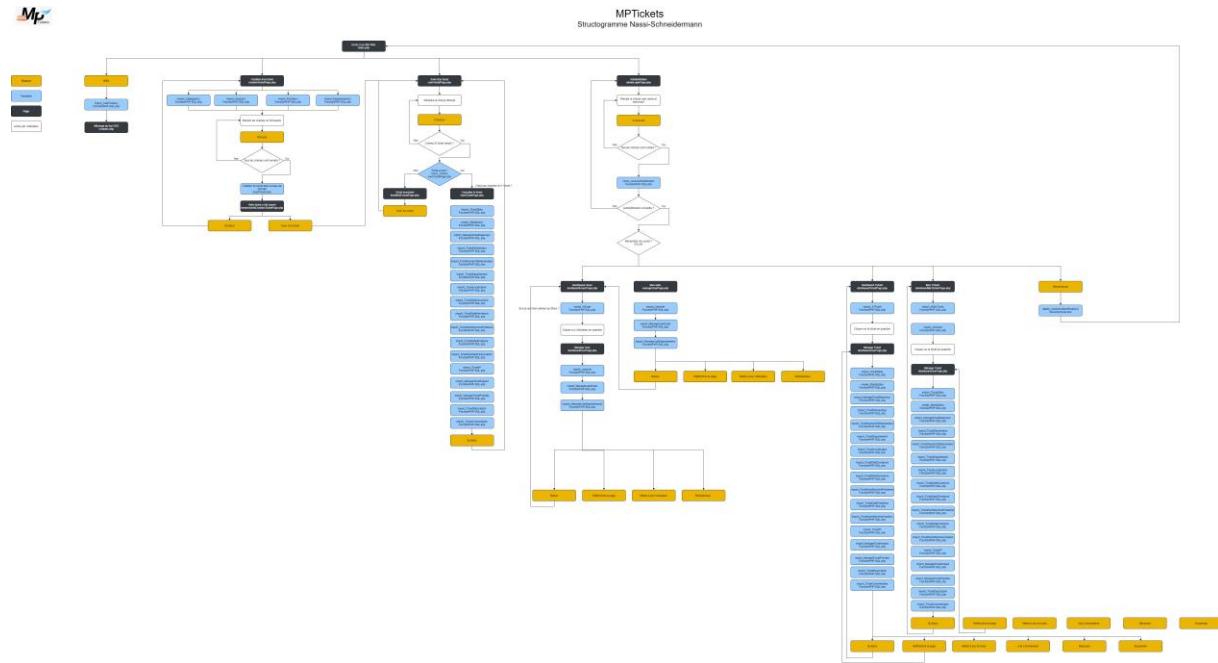
Planification du travail initial



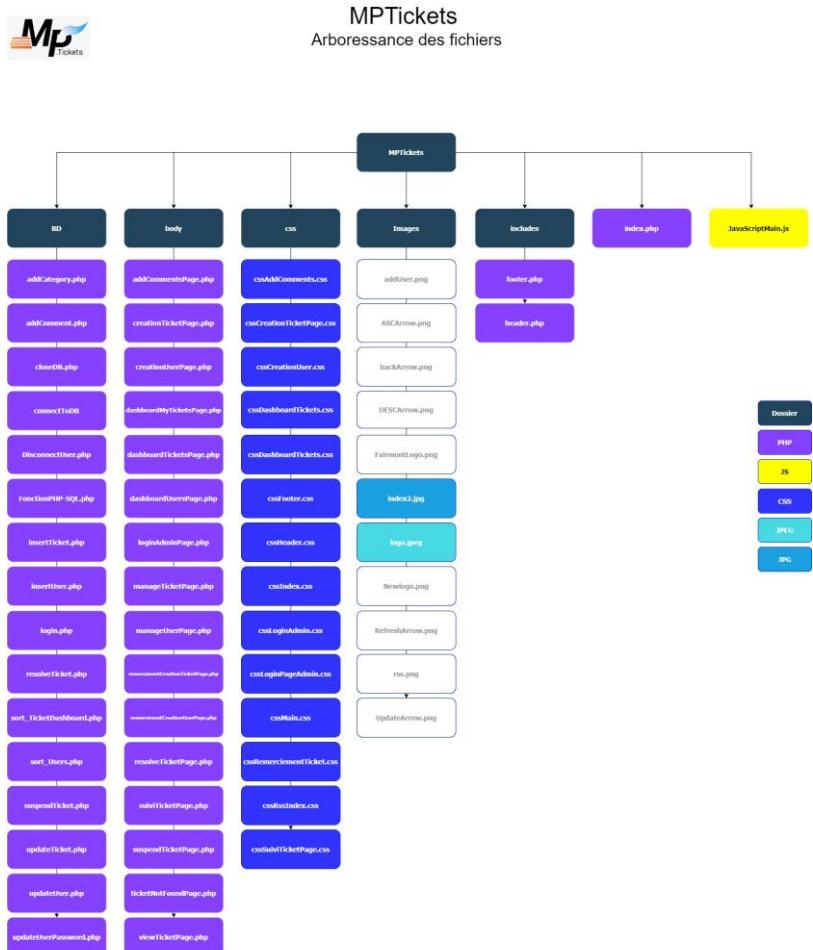
Planification du travail détaillé



Structogramme - Nassi-Schneidermann



Arborescence des fichiers



Sécurité du site

Injection SQL

Les injections SQL ou HTML sont des attaques basiques et connues dans le monde du Web. Pour éviter cela, nous avons ajouté la fonction **htmlspecialchars** de PHP. Cette fonction permet de convertir les caractères spéciaux en entités HTML.

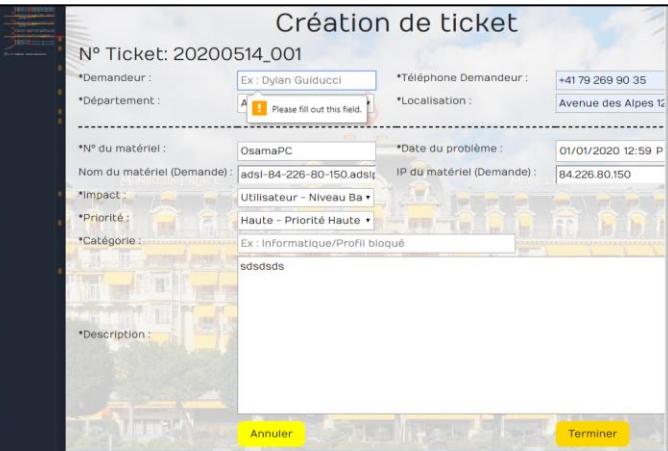
```
$username = htmlspecialchars($_POST['username']); // Récuperer le username en évitant les injections SQL
$password = htmlspecialchars($_POST['password']); // Récuperer le password en évitant les injections SQL
```

Champs non remplis

Pour éviter les champs non remplis lors d'une création d'un ticket, le code source des pages contient l'attribut « **required** » dans les types d'entrées du type **input**.

Required est un attribut du langage HTML. Lorsqu'il est présent, il précise qu'un champ de saisie doit être rempli avant de soumettre le formulaire. Il est présent dans le code de la page `creationTicketPage.php`.

L'utilisateur est obligé de remplir les champs requis du formulaire. Cela permet d'éviter le manque d'information dans les formulaires.



The screenshot shows a web browser displaying a form titled "Création de ticket". The form has several fields:

- *Demandeur: Ex: Dylan Guiducci (highlighted in red)
- *Département: A [input field] Please fill out this field. (highlighted in red)
- *Localisation: Avenue des Alpes 12
- *N° du matériel: OsamaPC
- *Date du problème: 01/01/2020 12:59 P
- Nom du matériel (Demande): adsl-84-226-80-150.adsl
- IP du matériel (Demande): 84.226.80.150
- *Impact: Utilisateur - Niveau Ba
- *Priorité: Haute - Priorité Haute
- *Catégorie: Ex : Informatique/Profil bloqué
- sdsdsds
- *Description: [Text area]

At the bottom right of the form are two buttons: "Annuler" and "Terminer".

On the left side of the screenshot, there is a code editor window showing the source code of the "creationTicketPage.php" file. The code includes HTML and PHP, with several lines highlighted in red, corresponding to the fields in the form above.

Accès aux fichiers de source via les liens

Certaines pages ne doivent pas être accessibles aux utilisateurs non identifiés comme la page Dashboard User.

La session est un mécanisme qui permet à PHP de garder « en mémoire » un nombre illimité de valeurs entre plusieurs requêtes d'un même utilisateur. Nous stockons deux valeurs dans la session. Le nom de l'utilisateur et le hash du mot de passe pour vérifier si l'utilisateur est connecté.

Exemple :

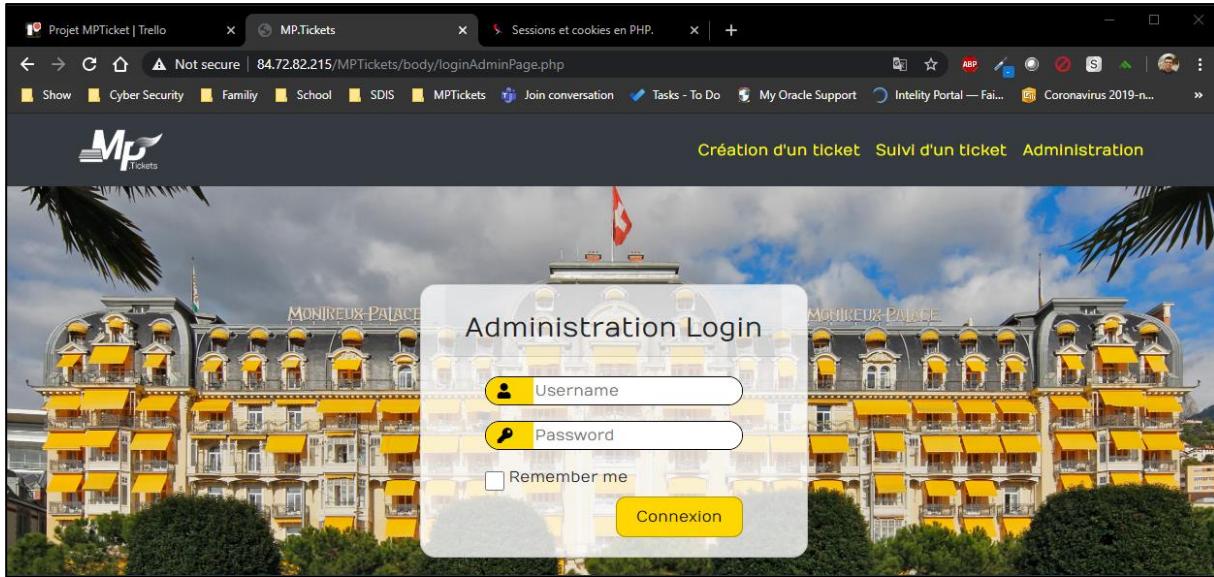
L'utilisateur non identifié tente d'accéder sur la page Dashboard ticket directement depuis la barre de lien juste en ajoutant le nom de la page :

<https://84.72.82.215/MPTickets/body/dashboardTicketsPage.php>



Rapport de projet – Groupe 6

Système Ticketing - MPTickets



Cela ne va pas marcher. Car il n'est pas identifié et il sera redirigé sur la page de connexion.
Fonctionnement : Au début de chaque page qui n'a pas besoin d'authentification, il y a la variable \$NoAuthenticationPage qui est égal à vrai.

```
// Ajout la variable pour enlever l'authentification  
// Doit être mis avant importation du header  
$NoAuthenticationPage = true;
```

Les pages ont besoin d'une authentification sauf si la variable \$NoAuthenticationPage sur la page est égal à vrai.

- Si la variable n'existe pas ---> Authentication
- Si la variable existe ---> No AuthenticationCette

```
if(empty($NoAuthenticationPage))  
{  
    // Verifie l'autentification  
    check_UserRedirection();  
}
```

Si une page requière une authentification et que l'utilisateur n'est pas authentifié, il sera redirigé sur la page login.

```
// Fonction qui redirige le User sur la page de Login s'il n'est pas connecté  
function check_UserRedirection()  
{  
    $UserAuthentication = check_UserAuthentication();  
    // Vérifie si l'utilisateur n'est pas connecté  
    if ($UserAuthentication === false)  
    {  
        // Si l'Utilisateur n'est pas authentifié, il est redirigé vers la page de Log  
        in  
        header('Location: /MPTickets/body/loginAdminPage.php');  
    }  
}
```



Rapport de projet – Groupe 6

Système Ticketing - MPTickets

Salt

Nous utilisons des salts pour augmenter la sécurité [des mots de passe](#). Le salt s'ajoute au mot de passe ce qui [le](#) rend plus fort.

Le salt est généré [aléatoirement lors de](#) la création du compte ainsi qu'à chaque changement de mot de passe. Pour le créer nous générerons une suite de 8 caractères qu'on va ensuite chiffrer en sha-256 pour avoir plus de caractères. Nous sauvegardons le salt dans la base de données.

Login	salt_user
Lohri	411e98ee060fee5a78d1fa32abcd8c7cbee52db42dd0ef5...
Didier	484e6a1619ba2feab100dcc6f01336ee4560f3f7cc61dcf3f4...
test	44260cd158020859327762a7d98dba2106caf1585edfb7799...
osama	b24887e05a00c9c09f4f13ad72cf4b3ce616d1dce640d93eb6...
dylan	50ef6ba722bfccaceacd8d7fa4ca25a5e49648fa18f301e7d5...
Angelo	848190bddf35fe660a797b4f02ed76ec3d1bfe9ee82661f93d...

```
function create_SaltUser()
{
    // Choix des caractères
    $listeCar =
'0123456789abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ<\/$£_-_
.:;.!ééà?%&*@#=+>';
    // Longeur de la chaîne
    $longueur = 8;
    $chaine = '';
    // Compter le nombre de caractères
    $max = mb_strlen($listeCar, '8bit') - 1;

    // Faire une boucle qui génère la chaîne
    for ($i = 0; $i < $longueur; ++$i)
    {
        $chaine .= $listeCar[random_int(0, $max)];
    }

    // Hasher le tout en SHA-256
    $chaine = hash("sha256", $chaine);

    // Retourner le salt
    return $chaine;
}
```



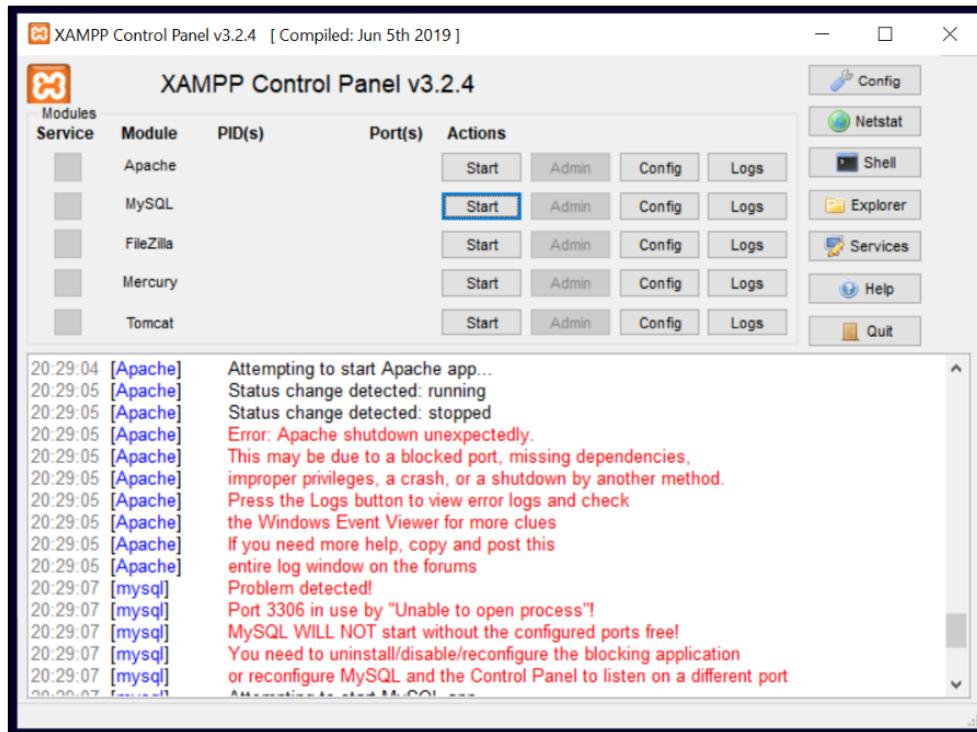
Rapport de projet – Groupe 6

Système Ticketing - MPTickets

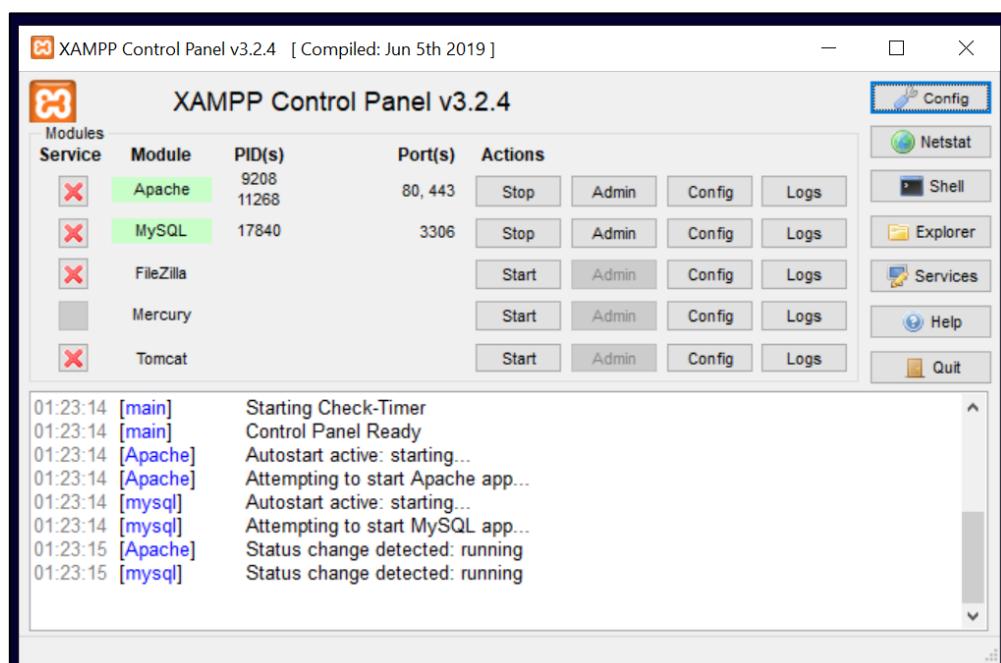
Problèmes rencontrés

Installation du [serveur local](#) XAMP sur le PC de Osama Shalhoub

Le 20 février 2020, Osama Shalhoub et Ali Alshweki (cousin du membre qui est dans le domaine du développement des sites Web) [ont essayé de se connecter](#) sur l'application Web avec la base de données. Malheureusement le serveur [local](#) XAMP ne démarrait pas Apache ni MYSQL.



Solution : La solution de ce problème était grâce à l'aide du professeur Monsieur Rogeiro en désinstallant MAMP et réinstallant XAMP, puis lancer XAMP CONTROLER avec les droits d'administrateur.



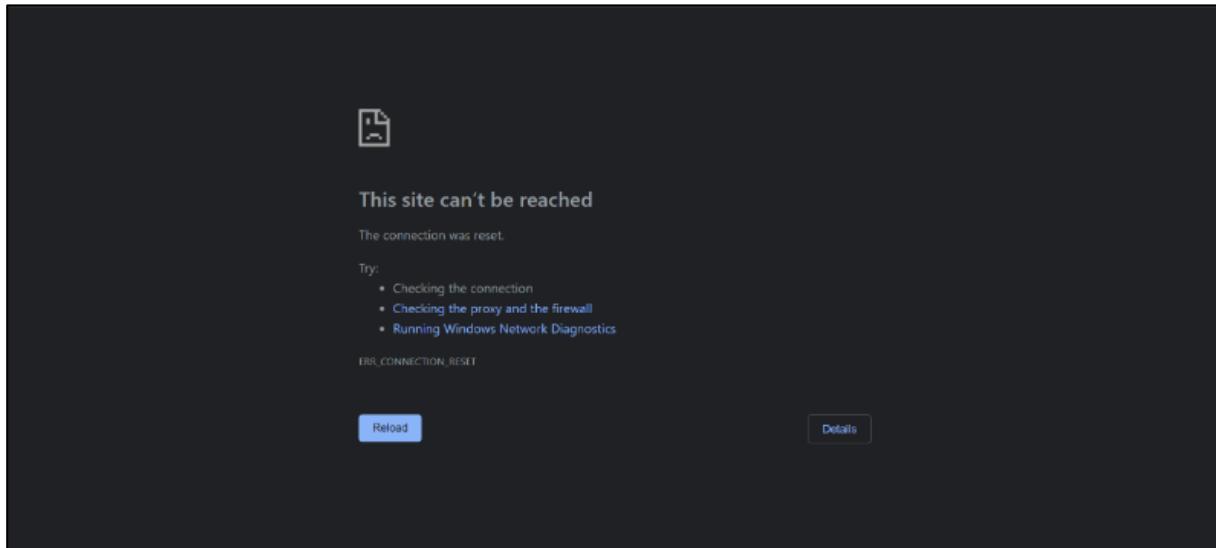


Rapport de projet – Groupe 6

Système Ticketing - MPTickets

Changement d'adresse IP publique du NAS qui héberge le projet

Le NAS qui héberge le projet MPTickets n'était malheureusement **plus** atteignable avec l'adresse IP publique 80.219.100.204.



Solution : La solution de ce problème était que le membre du groupe de projet Dylan Guiducci prenne contact avec UPC pour **leur demander la nouvelle adresse IP publique du routeur**. UPC a donc changé l'adresse IP publique 80.219.100.204 en 84.72.82.215

Connexion à la base de données

La connexion entre le projet MPTickets et la base de données a bien été établie en local où nous avons pu récupérer des données d'une table :

```
C:\xampp\htdocs\MP.Tickets\includes\connectToDB.php - Notepad++  
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window 2  
connectToDB.php connectToDB.php  
1 <?php  
2  
3 $dbname = 'mpticket';  
4 $dbuser = 'MPTicket';  
5 $dbpass = 'root';  
6 $dbhost = 'localhost';  
7  
8 // Connexion au serveur  
9 $link=mysqli_connect($dbhost, $dbuser, $dbpass) or die("Erreur de connexion au serveur base de données, connard");  
10 mysqli_select_db($link,$dbname) or die("Base de données pas trouvée, connard");  
11  
12 ?>
```

Code pour la connexion à la base de données en local

2 Moyen Niveau Moyen : Le problème impacte entre 2 à 10 personnes
3 Haut Niveau Haut : Le problème impacte plus que 10 personnes

Récupération des données d'une table depuis la base de données



Rapport de projet – Groupe 6

Système Ticketing - MPTickets

Malheureusement, la connexion à la base de données via le NAS avait posé un problème. Le membre du projet a tenté de renommer le projet MP.Tickets à MPTickets pour vérifier si ce n'est pas le « . » qui posait des problèmes mais cela n'avait pas résolu le problème.

Warning: mysqli_connect(): php_network_getaddresses: gethostbyname failed. errno=11 in /volume1/web/MPTickets/BD/connectToDB.php on line 9
Warning: mysqli_connect(): (HY000/2002): php_network_getaddresses: gethostbyname failed. errno=11 in /volume1/web/MPTickets/BD/connectToDB.php on line 9 Erreur de connexion au serveur base de données, connard

Erreur lors de la connexion à la base de données via le NAS

```
1 <?php
2
3 $dbname='MPTicket';
4 $dbuser='MPTicket';
5 $dbpass='R0gelR0-151';
6 $dbhost=$_SERVER['DOCUMENT_ROOT'];
7
8 // Connexion au serveur
9 $link=mysqli_connect($dbhost, $dbuser, $dbpass) or die("Erreur de connexion au serveur base de données, connard");
10 mysqli_select_db($link, $dbname) or die("Base de données pas trouvée, connard");
11
12 ?>
```

Code pour la connexion à la base de données via le NAS

Nom	Valeur
mysqli.default_pw	
mysqli.default_socket	/run/mysqld/mysqld10.sock
mysqli.default_user	
mysqli.max_links	-1
mysqli.max_persistent	-1
mysqli.reconnect	0
mysqli.rollback_on_cached_plink	0
mysqlnd.collect_memory_statistics	0
mysqlnd.collect_statistics	1
mysqlnd.debug	
mysqlnd.fetch_data_copy	0
mysqlnd.log_mask	0
mysqlnd.mempool_default_size	16000
mysqlnd.net_cmd_buffer_size	4096
mysqlnd.net_read_buffer_size	32768
mysqlnd.net_read_timeout	86400
mysqlnd.sha256_server_public_key	
mysqlnd.trace_alloc	
pdo_mysql.default_socket	/run/mysqld/mysqld10.sock

Solution : Le membre du groupe Dylan a comparé les paramètres avancés de PHP par rapport à celui du professeur Rogeiro.

Nous avons trouvé deux différences. **Mysqli.default_socket** et **pdo_mysql.default_socket** qui était faux. Les paramétrages pour MariaDB5 sont installés de base et pas MariaDB10 qui est installé sur le NAS. Ils ont décidé de modifier le répertoire de travail lors de la migration. Et nous devons le modifier nous-même.

Manque de connaissance en CSS

Les membres du groupe Osama et Dylan ont de la peine avec la mise en page CSS. Le module 101 qui a été fait en cours inter-entreprises remonte à trop loin et ils ont beaucoup de trous de mémoire.

Solutions : Pour remédier à cela, ils essayent de se former avec différents sites comme OpenClasseRoom et W3School. Et surtout faire des tests sur le site. Mais ils perdent beaucoup de temps.

Liaison entre la table T_Tickets et T_Utilisateurs

L'idée d'un des membres de projet, était de faire une liste déroulante pour afficher les utilisateurs actuels lors de la création d'un ticket et si la personne n'avait pas de compte, il devra créer un pour créer un ticket. Le problème, ça serait de créer une liste déroulante pour afficher les utilisateurs avec une relation avec la table utilisateurs, créer une page pour créer des utilisateurs en utilisant de l'Ajax.

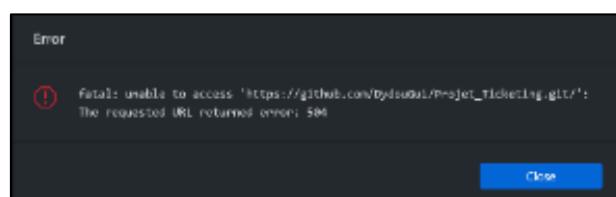
Malheureusement c'est une solution très difficile car nous n'avons pas de connaissance sur l'utilisation de l'Ajax.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	id_ticket	varchar(15)	utf8_general_ci		No	None	id de la table Ticket - Format (YYYYMMDD_XXX)		
2	fk_id_utilisateur_demandeur	int(16)			No	None	(Demandeur - Crée par) Clé étrangère de la table Utilisateur		
3	fk_id_utilisateur_beneficiaire	int(16)			No	None	(Bénéficiaire - Crée pour) Clé étrangère de la table Utilisateur		
4	fk_id_utilisateur_depanneur	int(16)			Yes	NULL	(Dépanneur) Clé étrangère de la table Utilisateur		
5	fk_id_status	int(4)			No	None	Status du ticket - Ouvert, Fermé, Résolu - Clé étrangère de la table Status		
6	fk_id_impact	int(4)			No	None	Impacte du ticket - Nombre de personnes affectée - Clé étrangère de la table Impacte		
7	fk_id_priorite	int(4)			No	None	Priorité du ticket - 1,2,3 - Clé étrangère de la table Priorité		
8	fk_id_categorie	int(12)			No	None	Catégorie du ticket - Clé étrangère de la table Catégorie		
9	Date_ouverture	datetime			No	None	Date d'ouverture du ticket		
10	Date_fermeture	datetime			Yes	NULL	Date fermeture ticket		
11	IP	varchar(15)	utf8_general_ci		No	None	IP de la machine ayant fait le ticket		
12	Nom_machine_creation	varchar(30)	utf8_general_ci		No	None	Nom de la machine ayant fait le ticket (PC-190043)		
13	Nom_machine_probleme	varchar(30)	utf8_general_ci		No	None	Nom de la machine ayant le problème (PC-190043)		
14	Description_ticket	text	utf8_general_ci		No		Description du ticket fait par l'utilisateur		
15	Résolution	text	utf8_general_ci		Yes		Résolution du ticket fait par le dépanneur		

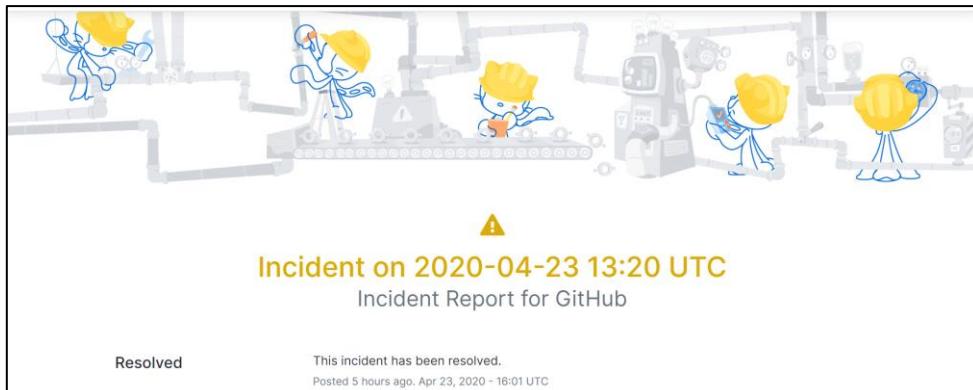
Solution : Nous allons faire en sorte que la personne qui créé le ticket sera stocké dans la table T_Tickets et la table T_Utilisateurs va concerner que les administrateurs. Et lors de la création d'un ticket, la personne mettra son nom et il n'y aura pas de liste déroulante.

GitHub problème - 2020-04-23 13:20 UTC

Le 23 Avril 2020, la solution GitHub est tombée en panne et cela nous a causé un problème de mettre à jours les fichiers.



Solution : L'équipe de GitHub a résolu le problème.



Lecture Flux RSS sur Chrome

Depuis un certain temps, les navigateurs comme Chrome, Edge et Firefox ont retiré la fonction de lecture de Flux RSS. Le seul moyen de pouvoir lire le FLUX RSS du site MPTickets est d'utiliser le navigateur Internet Explorer ou ajouter l'extension « Extension Abonnement RSS (par Google) » dans les navigateurs qui n'ont pas de lecteur RSS incorporé.

```
<?xml version="1.0" encoding="UTF-8" ?> <!-- Laisse à la première ligne sinon le fil ne reconnaît pas comme fichier XML -->

<rss version="2.0">
    <channel>
        <title>MPTickets</title>
        <description>Flux RSS de MPTickets - Derniers tickets résolus</description>
        <item>
            <title>Booke4Time</title>
            <description>wwwqqqqqq/nccccccc...</description>
            <pubDate>Thu, 04 Jun 2020 11:37:51 +0200</pubDate>
        </item>
        <item>
            <title>Outlook</title>
            <description>4/nj...</description>
            <pubDate>Thu, 04 Jun 2020 11:37:21 +0200</pubDate>
        </item>
        <item>
            <title>Problème fichier / Dossier</title>
            <description>444/nggggg...</description>
            <pubDate>Thu, 04 Jun 2020 11:36:47 +0200</pubDate>
        </item>
        <item>
            <title>Demande Matériels</title>
            <description>Test de la requête SQL INSERT INTO/test...</description>
            <pubDate>Wed, 27 May 2020 12:46:33 +0200</pubDate>
        </item>
        <item>
            <title>Demande de droit</title>
            <description>J'aurais besoin de droit sur ma nouvelle machine/nLes droits ont été ajoutés depuis l'Active Directory. Dylan...</description>
            <pubDate>Tue, 26 May 2020 13:49:56 +0200</pubDate>
        </item>
    </channel>
</rss>
```

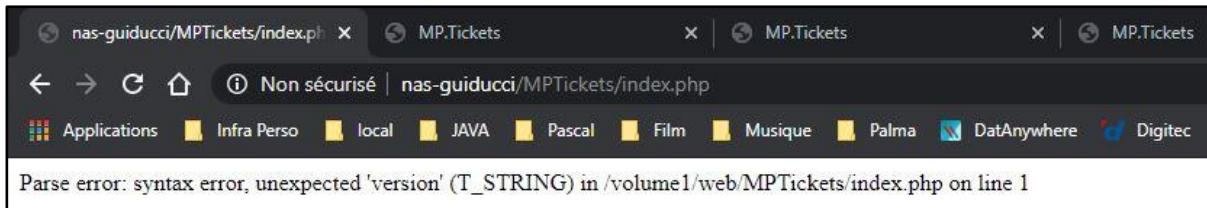
Titre	Description	Date
Booke4Time	wwwqqqqqq/nccccccc...	04/06/2020 11:37:51
Outlook	4/nj...	04/06/2020 11:37:21
Problème fichier / Dossier	444/nggggg...	04/06/2020 11:36:47
Demande Matériels	Test de la requête SQL INSERT INTO/test...	27/05/2020 12:46:33
Demande de droit	J'aurais besoin de droit sur ma nouvelle machine/nLes droits ont été ajoutés depuis l'Active Directory. Dylan...	26/05/2020 13:49:56

Solution : Avec Monsieur Rogeiro, nous avons décidé de créer simplement un bouton dans la page Index qui va diriger l'utilisateur sur la page qui contient le flux RSS.

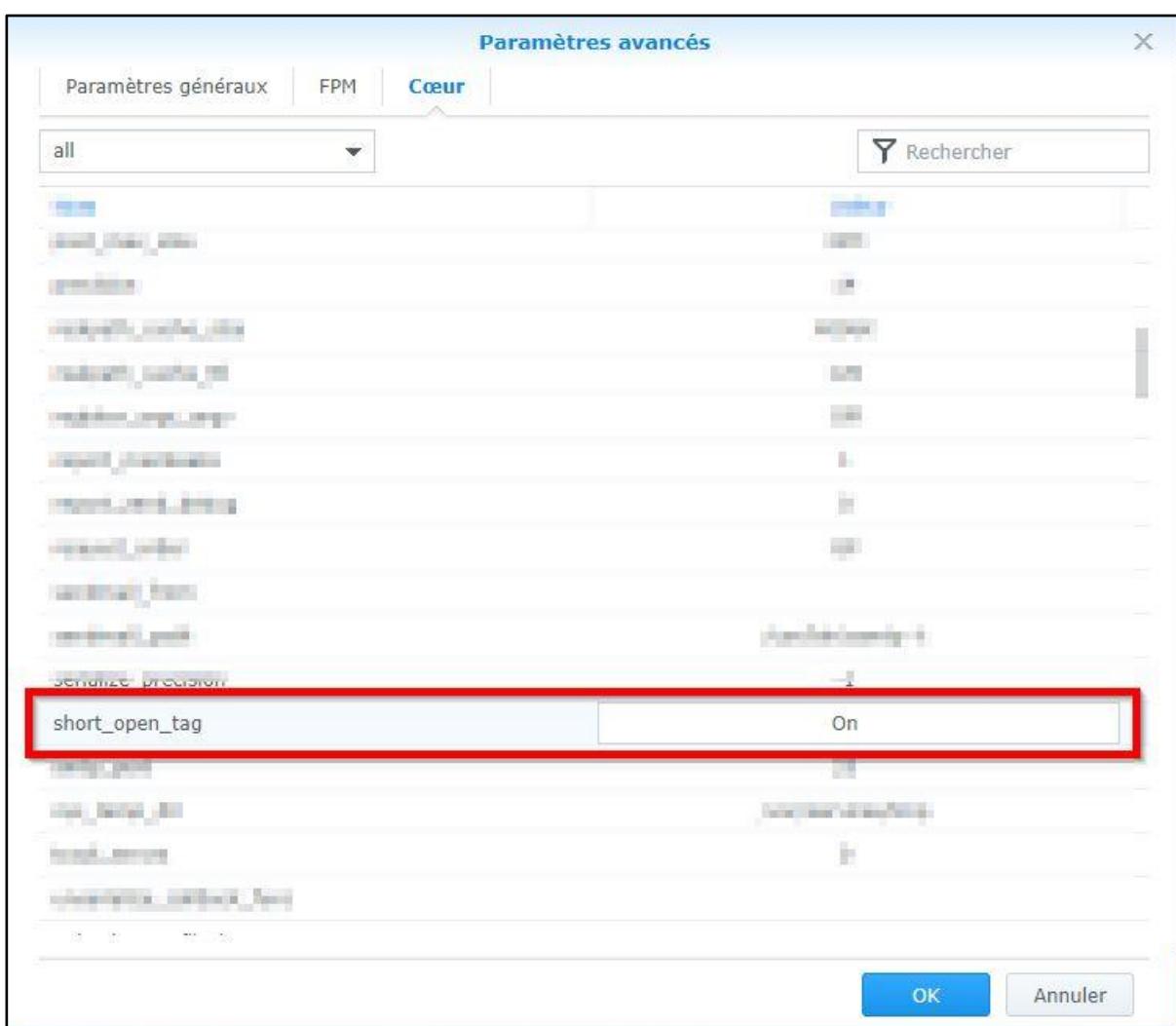
Variable courte activée

Lors de la création du flux RSS, nous avons eu un problème car il ne faisait pas la différence entre les variables XML et PHP.

Solution : Après quelques recherches sur Internet, cela venait des variables courtes qui étaient activées dans le fichier php.ini. Du coup nous avons désactivé cette fonctionnalité.



The screenshot shows a browser window with three tabs: 'nas-guiducci/MPTickets/index.php', 'MP.Tickets', and another 'MP.Tickets'. The address bar indicates the page is 'Non sécurisé'. The main content area displays the error: 'Parse error: syntax error, unexpected 'version' (T_STRING) in /volume1/web/MPTickets/index.php on line 1'.



The screenshot shows the 'Paramètres avancés' dialog with the 'Cœur' tab selected. A search bar at the top right contains the text 'Rechercher'. Below it is a dropdown menu set to 'all'. The main list contains many items, and one specific entry, 'short_open_tag', is highlighted with a red rectangle. To its right, a switch labeled 'On' is also highlighted with a red rectangle. At the bottom right of the dialog are 'OK' and 'Annuler' buttons.

Fermeture de balise

Nous avons eu un problème avec le caractère chevron « > » sur la page « Crédit d'un utilisateur ».

Le caractère s'affichait sur la page et nous n'arrivions pas à détecter d'où il venait. Même en inspectant la page F12, le caractère n'apparaissait pas. Nous avons aussi demandé de l'aide d'un camarade Quentin Krenger mais lui aussi il n'a pas pu trouver d'où il venait.



Rapport de projet – Groupe 6

Système Ticketing - MPTickets

Solution : En relisant l'entier du code de la page, le caractère venait de la partie du password à la fin de la balise.

```
<tr>
    <td class="TdCol_1"></td>
    <td class="TdCol_2"></td>
    <td class="TdCol_3"><label for="Password">*Password : </label>
</td>
    <td class="TdCol_2"><input required type="Password" name="Pass
word" id="Password"/></td>
</tr>
```

Création d'un utilisateur

A chaque fois qu'on essayait de créer un utilisateur, dans le champ « login », on avait Array qui apparaissait.

Utilisateurs					
Prénom	Nom	Login	Email	Département	Droit
Angelo	Rogeiro	Array	angelo.rogeiro@fairmont.com	Willow Stream Spa	Contrôle Total
De Test	User	test	test.deUser@MPTickets.ch	Housekeeping	Contrôle Total
Didier	Lohri	Array	didier.lohri@fairmont.com	Comptabilité	Lecture
Dylan	Guiducci	dylan	dylan.guiducci@fairmont.com	IT - Informatique	Contrôle Total
Osama	Shalhoub	osama	osama.shalhoub@fairmont.com	IT - Informatique	Contrôle Total

Solution : Dans la partie du code importation des données du nouvel utilisateur, il manquait \$_POST

```
// Importation des données
$fk_departement=$_POST['Departement'];
$fk_droit=$_POST['Droit'];
$Login=$_POST['Login'];
$pseudo_user=""; //Création aléatoire
>Password=$_POST['Password'];
$RepeterMotPasse=$_POST['RepeterMotPasse'];
$Nom=$_POST['Nom'];
$Prenom=$_POST['Prenom'];
```



Rapport de projet – Groupe 6

Système Ticketing - MPTickets

Amélioration pour les suites des versions du Site Web

Bouton « Ajouter Documents »

Lors de la création du ticket, un bouton permet l'ajout de documents pour donner plus de détails sur le problème comme une photo du message d'erreur.

Site en Anglais

Comme aussi la plupart des sites Web, il est possible d'avoir le site en anglais, en allemand et en français. Dans les prochaines versions à venir, on pourrait mettre une version du site entièrement en anglais.

Cette option prend du temps, car il faudra traduire les pages, les messages dans les fichiers sources, et un nouveau champ qui indique la langue de l'attribut dans la base de données.

Gestion de droit sur le site

Dès le début de la réalisation du projet, nous avons donné créer les différents options de droit sur le site pour les utilisateurs comme :

- Control total – Contrôle total sur le site comme créer un utilisateur
- Modification – Modification des tickets
- Lecture – S'identifier sur le site mais pouvoir uniquement consulter les Dashboard et les tickets sans modifier

Malheureusement au fur à mesure de la réalisation du projet, nous avons constaté que ce n'était pas si simple que ça et nous n'avons pas assez de temps. Donc cela est planifié pour les prochaines versions.

MPTickets

Osama Shalhoub

Dylan Guiducci