

Documentation Technique

Projet Ligue de Basketball de Lorraine



Réalisé par : Vadier Mathis

Présentation et Objectifs

Nous sommes l'équipe 2, composée de **Artaud Jules** et **Vadier Mathis**. Nous avons choisi la mission 5.

Afin de réaliser notre mission, nous avons 2 tâches à accomplir :

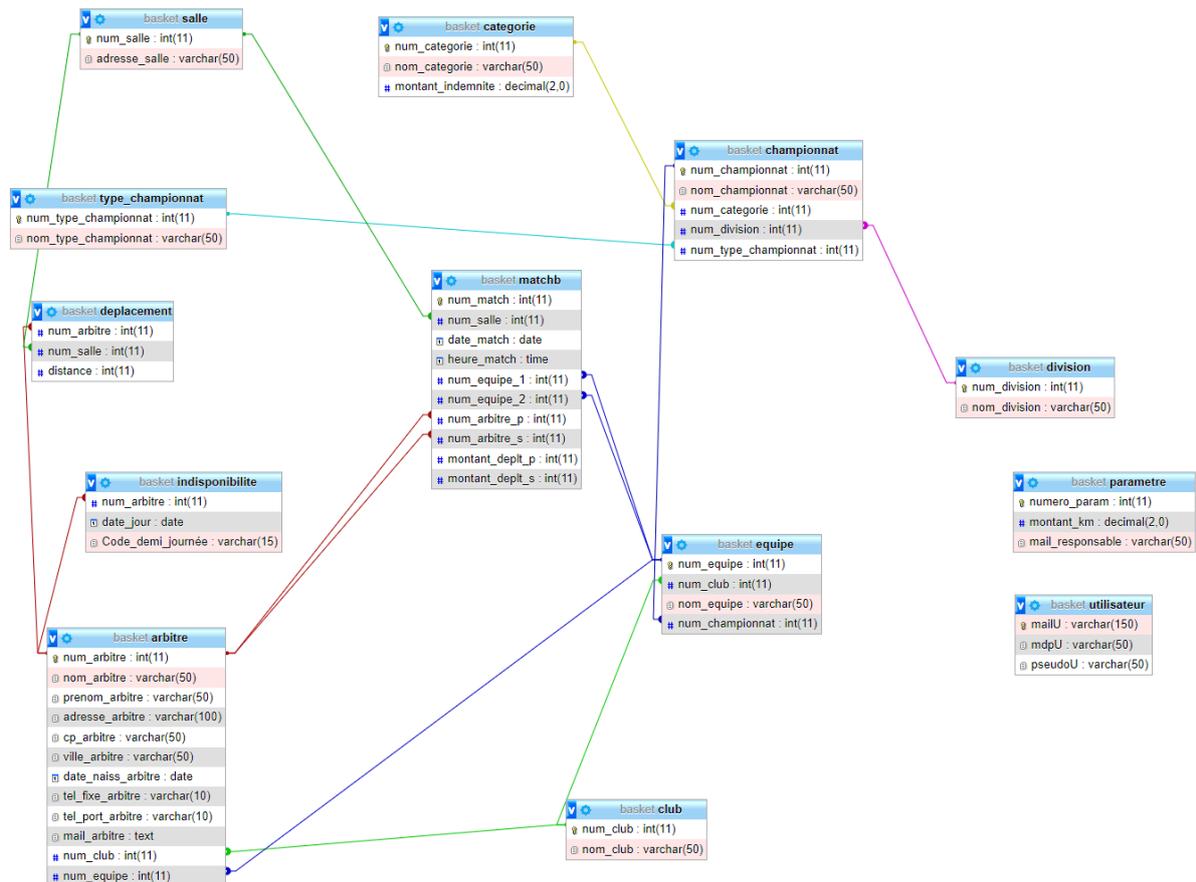
1 - Gestion des arbitres : Cette tâche doit permettre la création, la modification et la suppression d'un arbitre.

2 - Edition des matches prévus pour un jour donné : Le responsable veut pouvoir à tout moment, choisir une date et consulter la liste des matches ayant lieu ce jour-là avec les informations le concernant (date, heure, lieu, équipes et arbitres concernés). Cet état doit être généré au format PDF.

Environnement de travail

Avant de découvrir notre travail, laissez nous vous présenter notre environnement. Nous avons travaillé à l'aide d'une plateforme de développement Web **XAMPP** et d'une base de données **MySQL** que nous avons dû créer pour le projet.

Voici le schéma relationnel :



Par la suite, pour réaliser notre site web nous avons utilisé l'IDE **Visual Studio Code** avec lequel nous avons utilisés les langages de programmation : **HTML**, **CSS** et **PHP** tout en respectant une architecture **MVC** (Modèle / Vue / Contrôle).

Connexion à la BDD

Pour réaliser la connexion à la base de données nous avons dû ajouter une extension nommée PDO qui permet de relier la base de données avec PHP.

La connexion se fait à l'aide de la fonction **connexionPDO()** qui comporte des variables avec les informations sur le serveur, la base de données, nom d'utilisateur et du mot de passe :

```
function connexionPDO() {
    $login = "root";
    $mdp = "";
    $bd = "basket";
    $serveur = "localhost";

    try {
        $conn = new PDO("mysql:host=$serveur;dbname=$bd", $login, $mdp,
            array(PDO::MYSQL_ATTR_INIT_COMMAND => 'SET NAMES \'UTF8\''));
        $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
        return $conn;
    } catch (PDOException $e) {
        print "Erreur de connexion PDO ";
        die();
    }
}
```

Détails de notre travail

- Connexion et déconnexion au site
- Ajouter un arbitre
- Supprimer ou modifier un arbitre
- Afficher les matches

Connexion et déconnexion au site

Connexion

Afin de gérer la connexion au site, nous avons créé une table utilisateur dans la base de données, nous avons par la suite créé plusieurs fonctions afin de récupérer les informations de la table.

Cette fonction récupère les données sur les utilisateurs tels que les adresses mails et les mots de passe.

```
function getUtilisateurs() {
    $resultat = array();

    try {
        $cnx = connexionPDO();
        $req = $cnx->prepare("select * from utilisateur");
        $req->execute();

        $resultat = $req->fetchAll(PDO::FETCH_ASSOC);

    } catch (PDOException $e) {
        print "Erreur !: " . $e->getMessage();
        die();
    }
    return $resultat;
}
```

Cette fonction récupère les données d'un utilisateur à partir de son mail de connexion.

```
function getUtilisateurByMailU($mailU) {
    $resultat = array();

    try {
        $cnx = connexionPDO();
        $req = $cnx->prepare("select * from utilisateur where mailU=:mailU");
        $req->bindValue(':mailU', $mailU);
        $req->execute();

        $resultat = $req->fetch(PDO::FETCH_ASSOC);
    } catch (PDOException $e) {
        print "Erreur !: " . $e->getMessage();
        die();
    }
    return $resultat;
}
```

Cette fonction permet de se connecter, pour cela elle récupère les identifiants de connexion en vérifiant qu'ils correspondent à la base de données.

```
function login($mailU, $mdpU) {
    if (!isset($_SESSION)) {
        session_start();
    }

    $util = getUtilisateurByMailU($mailU);
    $mdpBD = $util["mdpU"];

    if (trim($mdpBD)==trim($mdpU)) {
        // le mot de passe est celui de l'utilisateur dans la base de données
        $_SESSION["mailU"] = $mailU;
        $_SESSION["mdpU"] = $mdpBD;
        return true;
    }
    else {
        return false;
    }
}
```

Voici le code du contrôleur de la connexion utilisant les fonctions précédentes

```
include "../modele/bd.authentification.php";

if (!isset($_POST["mailU"]) || !isset($_POST["mdpU"])) {
    ?>
    <?php
    include "../vue/vueAuthentification.php";
    } else {
        $mailU = $_POST["mailU"];
        $mdpU = $_POST["mdpU"];
        if (isLoggedIn()) {
            ?>
            <?php
            } else {
                if (login($mailU, $mdpU)==true) {
                    header("Location:/");
                } else {
                    echo " <div id='msgErr' class='alert alert-danger mx-auto' role='alert'>
                    Identifiants incorrect.
                    </div>";
                }
            }
        ?>
    <?php
    }
}

if (isset($_GET['login']) && $_GET['login'] == 'non') {
    echo " <div id='msgErr' class='alert alert-danger mx-auto' role='alert'>
    veuillez vous connecter pour accéder à cette page
    </div>";
}
```

Déconnexion

Afin de se déconnecter du site, nous avons une fonction **logout()** qui permet de vider les variables SESSION.

```
function logout() {  
    if (!isset($_SESSION)) {  
        session_start();  
    }  
    unset($_SESSION["mailU"]);  
    unset($_SESSION["mdpU"]);  
}
```

Ajouter un arbitre

Pour ajouter des arbitres nous avons créé une fonction, qui prend en paramètre toutes les informations saisies par l'utilisateur. Le type de requête utilisé par cette fonction est un insert cela permet d'insérer les données de l'arbitre dans la table Arbitre :

```
function addArbitre($num,$nom,$prenom,$adresse,$cp,$ville,$naiss,$telFix,$telPort,$mail,$club,$equipe){
    try {
        $cnx = connexionPDO();
        $req = $cnx->prepare("insert into arbitre values (null, :nom, :prenom, :adresse, :cp, :ville, :naiss, :telFix, :telPort, :mail, :club, :equipe)");
        $req->bindValue(':nom', $nom );
        $req->bindValue(':prenom', $prenom );
        $req->bindValue(':adresse', $adresse );
        $req->bindValue(':cp', $cp );
        $req->bindValue(':ville', $ville );
        $req->bindValue(':naiss', $naiss);
        $req->bindValue(':telFix', $telFix );
        $req->bindValue(':telPort', $telPort);
        $req->bindValue(':mail', $mail );
        $req->bindValue(':club', $club );
        $req->bindValue(':equipe', $equipe );

        $req->execute();

    } catch (PDOException $e) {
        print "Erreur !: " . $e->getMessage();
        die();
    }
}
```

Voici le code du contrôleur de l'ajout d'arbitre qui utilise la fonction précédente si on clique sur le bouton ajouter ou non.

```
<div id="AjoutArbitre">
    <?php
    include "modele/bd.basket.php";

    if (empty($_SESSION['mailU']) || empty($_SESSION['mdpU'])) {
        header("Location:/?action=connexion&login=non");
    }
    if (isset($_POST['btnAjout'])) {
        if (isset(
            $_POST['txtNom'],
            $_POST['txtPrenom'],
            $_POST['txtAdresse'],
            $_POST['txtCP'],
            $_POST['txtVille'],
            $_POST['txtNaissance'],
            $_POST['txtFixe'],
            $_POST['txtPort'],
            $_POST['txtMail'],
            $_POST['txtClub'],
            $_POST['txtEquipe']
        )) {
            addArbitre(
                NULL,
                $_POST['txtNom'],
                $_POST['txtPrenom'],
                $_POST['txtAdresse'],
                $_POST['txtCP'],
                $_POST['txtVille'],
                $_POST['txtNaissance'],
                $_POST['txtFixe'],
                $_POST['txtPort'],
                $_POST['txtMail'],
                $_POST['txtClub'],
                $_POST['txtEquipe']
            );
        }
    }
}
```

```

    );
    echo " <div id='msgValid' class='alert alert-success' role='alert'>
        Ajout effectué
    </div>";
} else {
    echo " <div id='msgErr' class='alert alert-danger mx-auto' role='alert'>
        Veuillez remplir tous les champs pour ajouter un arbitre
    </div>";
}
}
$listeClub = getClub();
$listeEquipe = getEquipe();
include "vue/vueAddArbitre.php"
?>
</div>

```

Dans le contrôleur nous affichons la vue du formulaire d'ajout ci-dessous

```

<form action="?action=ajout" method="POST" class="lb mb-3">
    <h1 id="lstA">Ajouter un arbitre</h1>
    </br>
    <label class="form-label"> Nom : </label></br>
    <input type="text" name='txtNom'></br>
    </br>
    <label class="form-label"> Prénom : </label></br>
    <input type="text" name='txtPrenom'></br>
    </br>
    <label class="form-label"> Adresse Postale : </label></br>
    <input type="text" name='txtAdresse'></br>
    </br>
    <label class="form-label"> Code Postale : </label></br>
    <input type="text" name="txtCP"></br>
    </br>
    <label class="form-label"> Ville : </label></br>
    <input type="text" name="txtVille"></br>
    </br>
    <label class="form-label"> Date de naissance : </label></br>
    <input type="date" name="txtNaissance"></br>
    </br>
    <label class="form-label"> Téléphone Fixe : </label></br>
    <input type="text" name="txtFixe"></br>
    </br>
    <label class="form-label"> Téléphone Portable : </label></br>
    <input type="text" name="txtPort"></br>
    </br>
    <label class="form-label"> Adresse mail : </label></br>
    <input type="text" name="txtMail"></br>
    </br>
    <select class="form-select" aria-label="Default select example" name="txtClub">
        <option selected>Sélectionnez un club</option>
        <?php
        foreach ($listeClub as $unClub) {
            ?>
            <option value=?php echo $unClub['num_club']; ?><?php echo $unClub['nom_club']; ?></option>
        <?php
        }
        ?>
    </select>
    </br>
    <select class="form-select" aria-label="Default select example" name="txtEquipe">
        <option selected>Sélectionnez une équipe</option>
        <?php
        foreach ($listeEquipe as $unEquipe) {
            ?>
            <option value=?php echo $unEquipe['num_equipe']; ?><?php echo $unEquipe['nom_equipe']; ?></option>
        <?php
        }
        ?>
    </select>
    </br>
    <input type="submit" value="AJOUTER" class="btna" name="btnAjout">
    <input type="submit" value="ANNULER" class="btnc" name="btnCancel">
</form>

```

Supprimer ou modifier un arbitre

Pour réaliser cette page, nous avons afficher les informations de la table arbitre, pour cela nous avons une fonction **getArbitre()** qui permet l'affichage des données.

```
function getArbitres() {
    $resultat = array();

    try {
        $cnx = connexionPDO();
        $req = $cnx->prepare("select num_arbitre,nom_arbitre, prenom_arbitre, adresse_arbitre, cp_arbitre,
ville_arbitre, date_naiss_arbitre, tel_fixe_arbitre,tel_port_arbitre, mail_arbitre, nom_club,
nom_equipe from arbitre a
inner join equipe e on a.num_equipe=e.num_equipe
inner join club c on a.num_club=c.num_club ");
        $req->execute();

        $resultat = $req->fetchAll(PDO::FETCH_ASSOC);

    } catch (PDOException $e) {
        print "Erreur !: " . $e->getMessage();
        die();
    }
    return $resultat;
}
```

Afin de modifier l'arbitre choisit par l'utilisateur, nous avons une seconde fonction qui récupère les données d'un arbitre précis à partir de son numéro.

```
function getArbitreById($id) {
    try {
        $cnx = connexionPDO();
        $req = $cnx->prepare("select * from arbitre a
inner join equipe e on a.num_equipe=e.num_equipe
inner join club c on a.num_club=c.num_club
where num_arbitre=:id");
        $req->bindValue(':id', $id);

        $req->execute();

        $resultat = $req->fetch(PDO::FETCH_ASSOC);

    } catch (PDOException $e) {
        print "Erreur !: " . $e->getMessage();
        die();
    }
    return $resultat;
}
```

Nous pouvons alors procéder à la modification, avec la fonction **updateArbitres(\$num, ... \$equipe)** qui prend en paramètre chaque colonne de la table arbitre. Le type de requête utilisé est UPDATE, cela permet de modifier un enregistrement déjà existant dans la table.

```
function updateArbitres($num,$nom,$prenom,$adresse,$cp,$ville,$naiss,$telFix,$telPort,$mail,$club,$equipe) {  
    try {  
        $cnx = connexionPDO();  
        $prep = $cnx->prepare("UPDATE arbitre SET nom_arbitre=:nomArb, prenom_arbitre=:prenomArb, adresse_arbitre=:adrArb,  
        cp_arbitre=:cpArb, ville_arbitre=:villeArb, date_naiss_arbitre=:naissArb, tel_fixe_arbitre=:telFixArb,  
        tel_port_arbitre=:telPortArb, mail_arbitre=:mailArb, num_club=:numClub, num_equipe=:numEqp  
        WHERE num_arbitre=:numArb");  
  
        $prep->bindValue('nomArb',$nom);  
        $prep->bindValue('prenomArb',$prenom);  
        $prep->bindValue('adrArb',$adresse);  
        $prep->bindValue('cpArb',$cp);  
        $prep->bindValue('villeArb',$ville);  
        $prep->bindValue('naissArb',$naiss);  
        $prep->bindValue('telFixArb',$telFix);  
        $prep->bindValue('telPortArb',$telPort);  
        $prep->bindValue('mailArb',$mail);  
        $prep->bindValue('numClub',$club);  
        $prep->bindValue('numEqp',$equipe);  
        $prep->bindValue('numArb',$num);  
  
        $prep->execute();  
  
    } catch (PDOException $e) {  
        print "Erreur !: " . $e->getMessage();  
        die();  
    }  
}
```

Après la modification, nous avons réalisé la suppression d'un arbitre. Pour cela nous avons utilisé plusieurs requêtes de type DELETE. La 1^{ère} fonction permet de supprimer le déplacement de l'arbitre choisi afin de pouvoir supprimer l'arbitre par la suite.

```
function deleteDeplacement($id){  
    try {  
        $cnx = connexionPDO();  
        $req = $cnx->prepare("delete from deplacement where num_arbitre=:id");  
        $req->bindValue(':id', $id);  
  
        $req->execute();  
  
    } catch (PDOException $e) {  
        print "Erreur !: " . $e->getMessage();  
        die();  
    }  
}
```

Pour finir, nous avons ici la suppression de l'arbitre choisi par l'utilisateur.

```
function deleteArbitre($id){
    try {
        $cnx = connexionPDO();
        $req = $cnx->prepare("delete from arbitre where num_arbitre=:id");
        $req->bindValue(':id', $id);

        $req->execute();

    } catch (PDOException $e) {
        echo " <div style='left:0;' id='msgErr' class='alert alert-danger mx-auto' role='alert'>
            Impossible de supprimer cet arbitre
        </div>";
        die();
    }
}
```

Le contrôleur de la modification et la suppression de l'arbitre utilise les différences vues suivant le choix de l'utilisateur sur les boutons.

```
<div id="ModifArbitre">
    <?php
    include "modele/bd.basket.php";
    $listeArbitres = getArbitres();
    $listeEquipe = getEquipe();
    $listeClub = getClub();
    include "vue/vueArbitre.php";
    if (empty($_SESSION['mailu']) || empty($_SESSION['mdpU'])) {
        header("Location:/?action=authentification&login=non");
    }
    if (isset($_GET['arbitre'])) {
        $id = $_GET['arbitre'];
        $unArbitreId = getArbitreById($id);
        include "vue/vueModif.php";
    }
    if (isset($_POST['btnEnregistrer'])) {
        updateArbitres(
            $_POST['txtNum'],
            $_POST['txtNom'],
            $_POST['txtPrenom'],
            $_POST['txtAdresse'],
            $_POST['txtCP'],
            $_POST['txtVille'],
            $_POST['txtNaissance'],
            $_POST['txtFixe'],
            $_POST['txtPort'],
            $_POST['txtMail'],
            $_POST['txtClub'],
            $_POST['txtEquipe']
        );
    }
    if (isset($_GET['idSuppr'])) {
        $idSuppr = $_GET['idSuppr'];
        deleteArbitre($idSuppr);
        deleteDeplacement($idSuppr);
    }
    ?>
</div>
```

Voici les vues de la page modification

```
<select class="form-select" aria-label="Default select example" name="txtClub">
  <option value="<?php echo $unArbitreId["num_club"]; ?>" selected<?php echo $unArbitreId["nom_club"]; ?></option>
  <?php
  foreach ($listeClub as $unClub) {
    ?>
    <option value="<?php echo $unClub['num_club']; ?>><?php echo $unClub['nom_club']; ?></option>
  }
  <?php
  ?>
</select>
</td>
<td>
  <select class="form-select" aria-label="Default select example" name="txtEquipe">
    <option value="<?php echo $unArbitreId["num_equipe"]; ?>" selected<?php echo $unArbitreId["nom_equipe"]; ?></option>
    <?php
    foreach ($listeEquipe as $unEquipe) {
      ?>
      <option value="<?php echo $unEquipe['num_equipe']; ?>><?php echo $unEquipe['nom_equipe']; ?></option>
    }
    <?php
    ?>
  </select>
</td>
<td><input type="submit" name="btnEnregistrer" value="ENREGISTRER" class="record"></td>
</tr>
</table>
</form>
```

```
<table class="table table-dark">
  <tr class="table-active">
    <td>Nom</td>
    <td>Prenom</td>
    <td>Adresse</td>
    <td>Code Postal</td>
    <td>Ville</td>
    <td>Date de naissance</td>
    <td>Telephone fixe</td>
    <td>Telephone portable</td>
    <td>Mail</td>
    <td>n° Club</td>
    <td>n° Equipe</td>
  </tr>
  <tr>
    <td><form action="?action=modification" method="POST">
      <input class="txt" type="hidden" name="txtNum" value="<?php echo $unArbitreId["num_arbitre"]; ?>" />
      <td><input class="txt" type="text" name="txtNom" value="<?php echo $unArbitreId["nom_arbitre"]; ?>" />
      <td><input class="txt" type="text" name="txtPrenom" value="<?php echo $unArbitreId["prenom_arbitre"]; ?>" />
      <td><input class="txt" type="text" name="txtAdresse" value="<?php echo $unArbitreId["adresse_arbitre"]; ?>" />
      <td><input class="txt" type="text" name="txtCP" value="<?php echo $unArbitreId["cp_arbitre"]; ?>" />
      <td><input class="txt" type="text" name="txtville" value="<?php echo $unArbitreId["ville_arbitre"]; ?>" />
      <td><input class="txt" type="text" name="txtNaissance" value="<?php echo $unArbitreId["date_naiss_arbitre"]; ?>" />
      <td><input class="txt" type="text" name="txtFixe" value="<?php echo $unArbitreId["tel_fixe_arbitre"]; ?>" />
      <td><input class="txt" type="text" name="txtPort" value="<?php echo $unArbitreId["tel_port_arbitre"]; ?>" />
      <td><input class="txt" type="text" name="txtMail" value="<?php echo $unArbitreId["mail_arbitre"]; ?>" />
    </td>
  </tr>
```

```
<h1 id="listArb">Liste des arbitres</h1>
<table class="table table-dark tableauArbitre">
  <tr class="table-active">
    <td>Nom</td>
    <td>Prenom</td>
    <td>Adresse</td>
    <td>Code Postal</td>
    <td>Ville</td>
    <td>Date de naissance</td>
    <td>Telephone fixe</td>
    <td>Telephone portable</td>
    <td>Mail</td>
    <td>Nom du Club</td>
    <td>Nom de l'équipe</td>
  </tr>
  <tr>
    <td><?php
    foreach ($listeArbitres as $unArbitre) {
      ?>
      <td><?php echo $unArbitre["nom_arbitre"]; ?></td>
      <td><?php echo $unArbitre["prenom_arbitre"]; ?></td>
      <td><?php echo $unArbitre["adresse_arbitre"]; ?></td>
      <td><?php echo $unArbitre["cp_arbitre"]; ?></td>
      <td><?php echo $unArbitre["ville_arbitre"]; ?></td>
      <td><?php echo $unArbitre["date_naiss_arbitre"]; ?></td>
      <td><?php echo $unArbitre["tel_fixe_arbitre"]; ?></td>
      <td><?php echo $unArbitre["tel_port_arbitre"]; ?></td>
      <td><?php echo $unArbitre["mail_arbitre"]; ?></td>
      <td><?php echo $unArbitre["nom_club"]; ?></td>
      <td><?php echo $unArbitre["nom_equipe"]; ?></td>
      <td><a href="index.php?action=modification&arbitre=<?php echo $unArbitre['num_arbitre']; ?>" class="btnm">MODIFIER</a></td>
      <td><a href="index.php?action=modification&idsuppr=<?php echo $unArbitre['num_arbitre']; ?>" class="btns">SUPPRIMER</a></td>
    }
  </tr>
</table>
```

Afficher les matches

Pour afficher les informations de tous les matches à partir d'un jour précis, nous utilisons une fonction **getMatches(\$date)**, elle récupère les enregistrements de matches qui ont lieu la date demandée.

Nous récupérons la date directement lorsque l'utilisateur la choisie dans le calendrier à l'aide d'une fonction JavaScript **submit()**.

```
function getMatches($date){
    $resultat = array();

    try {
        $cnx = connexionPDO();
        $req = $cnx->prepare("select num_match, adresse_salle, date_match, heure_match,
e1.nom_equipe as equipe1, e2.nom_equipe as equipe2, a1.nom_arbitre as arbitreP,
a2.nom_arbitre as arbitreS, montant_deplt_p, montant_deplt_s from matchb
inner join arbitre a1 on matchb.num_arbitre_p=a1.num_arbitre
inner join arbitre a2 on matchb.num_arbitre_s=a2.num_arbitre
inner join equipe e1 on matchb.num_equipe_1=e1.num_equipe
inner join equipe e2 on matchb.num_equipe_2=e2.num_equipe
inner join salle on matchb.num_salle=salle.num_salle where date_match=:date");
        $req->bindValue(':date', $date );

        $req->execute();

        $resultat = $req->fetchAll(PDO::FETCH_ASSOC);
    } catch (PDOException $e) {
        print "Erreur !: " . $e->getMessage();
        die();
    }
    return $resultat;
}
```

L'objectif est de récupérer les informations d'un match choisi et de créer un PDF des informations du match, pour cela nous avons réalisé un bouton télécharger qui récupère les données d'un seul match à partir de son numéro avec la fonction **getMatchById(\$id)**

```
function getMatchesById($id){  
  
    try {  
        $cnx = connexionPDO();  
  
        $req = $cnx->prepare("select num_match, adresse_salle, date_match, heure_match,  
e1.nom_equipe as equipe1, e2.nom_equipe as equipe2, a1.nom_arbitre as arbitreP,  
a2.nom_arbitre as arbitreS, montant_deplt_p, montant_deplt_s from matchb  
inner join arbitre a1 on matchb.num_arbitre_p=a1.num_arbitre  
inner join arbitre a2 on matchb.num_arbitre_s=a2.num_arbitre  
inner join equipe e1 on matchb.num_equipe_1=e1.num_equipe inner join equipe e2 on matchb.num_equipe_2=e2.num_equipe  
inner join salle on matchb.num_salle=salle.num_salle where num_match=:id");  
        $req->bindValue(':id', $id);  
  
        $req->execute();  
  
        $resultat = $req->fetch(PDO::FETCH_ASSOC);  
  
    } catch (PDOException $e) {  
        print "Erreur !: " . $e->getMessage();  
        die();  
    }  
    return $resultat;  
}
```

Le contrôleur utilisant les fonctions pour les matchs est ci-dessous

```
<div id="InfoMatch">  
    <?php  
    if (empty($_SESSION['mailU']) || empty($_SESSION['mdpU'])) {  
        header("Location:/?action=authentification&login=non");  
    }  
    include "modele/bd.basket.php";  
    $date = "";  
    include "vue/vueMatch.php";  
    if (isset($_POST['datePicker'])) {  
        $date = $_POST["datePicker"];  
        $listeMatches = getMatches($date);  
    }  
    ?>  
    <?php  
    $listeDates = getDateMatch();  
    foreach ($listeDates as $suneDate) {  
        if (isset($_POST['datePicker'])) {  
            $suneDate == $_POST['datePicker'];  
        }  
    }  
    if (isset($_POST['datePicker'])) {  
        if ($_POST['datePicker'] == $suneDate) {  
            include "vue/vueInformation.php";  
        }  
        else {  
            echo " <div id='msgErr' class='alert alert-danger mx-auto' role='alert'>  
                Aucun match n'existe à cette date  
            </div>";  
        }  
    }  
    ?>  
    <?php  
    }  
    ?>  
</div>
```

Les vues présentes dans le contrôleur sont séparées avec le formulaire comprenant la date et ensuite la seconde vue qui affiche le tableau

```
<h1 id="listArb">Liste des matchs</h1>
<form id="formDate"action=""?action=info" method="POST">

    <label style="color: aliceblue;">Choisissez votre date : </label>
    <input type="date" class="datepicker-input" name="datePicker" onChange="submit()" value=<?php echo $date ?>>
</form>
```

```
<div id="TableauInformation">
    <table class="table table-dark">
        <tr class="table-active">
            <td>Numéro de la salle</td>
            <td>Date du match</td>
            <td>Heure du match</td>
            <td>Nom de l'équipe 1</td>
            <td>Nom de l'équipe 2</td>
            <td>Nom de l'arbitre primaire</td>
            <td>Nom de l'arbitre secondaire</td>
            <td>Montant du déplacement de l'arbitre primaire</td>
            <td>Montant du déplacement de l'arbitre secondaire</td>
        </tr>
        <tr>
            <?php
            foreach ($listeMatches as $unMatch) {
                ?>
                <input class="txt" type="hidden" name="txtNum" value="<?=$unMatch["num_match"]; ?>">
                <td><?php echo $unMatch["adresse_salle"]; ?></td>
                <td><?php echo $unMatch["date_match"]; ?></td>
                <td><?php echo $unMatch["heure_match"]; ?></td>
                <td><?php echo $unMatch["equipe1"]; ?></td>
                <td><?php echo $unMatch["equipe2"]; ?></td>
                <td><?php echo $unMatch["arbitreP"]; ?></td>
                <td><?php echo $unMatch["arbitreS"]; ?></td>
                <td><?php echo $unMatch["montant_depl_t_p"]; ?></td>
                <td><?php echo $unMatch["montant_depl_t_s"]; ?></td>
                <td><a href="index.php?action=PDF&id=<?php echo $unMatch['num_match']; ?>" class="record">TELECHARGER</a></td>
            </tr></br>
            <?php
            }
        ?></table>
</div>
```

Nous affichons ensuite ces données dans un document PDF, pour cela nous avons importé la classe **fpdf.php** ainsi que paramétrer notre fichier PDF

```
<?php
require("pdf/tfpdf.php");
// // Activation de la classe
$pdf = new TFPDF();
// // Nouvelle page A4 (incluant ici logo, titre et pied de page)
$pdf->AddPage();
// // Polices par défaut : Helvetica taille 9
$pdf->SetFont('Arial','',9);
// // Couleur par défaut : noir
// Centre le titre
$pdf->Ln(10);
$pdf->SetX((190-40)/2);
// Couleur du texte et du fond du titre
$pdf->SetDrawColor(0, 0, 0);
$pdf->SetFillColor(255, 255, 255);
$pdf->SetTextColor(0, 0, 0);
// Taille du contour
$pdf->SetLineWidth(0,8);
// Titre
// Cellule (Longueur, Hauteur, Contenu, Bordure, nextline parameters, C=CENTER , Remplissage)
$title = 'Match numéro '.$unMatch['num_match'].' du '.$unMatch['date_match'];
$pdf->SetTitle($title);
$pdf->Cell(60, 19, $title, 0, 0, 'C', true);
// Saut de ligne
$pdf->Ln(15);
// Tableau Info
```

```
// Ligne 1
$pdf->Ln(10);
$pdf->SetTextColor(0, 0, 0);
$pdf->SetX((150-60)/2);
$pdf->Cell(120, 9, 'Informations generales', 1, 1, 'C');
$pdf->SetX((150-60)/2);
//Ligne 2
$pdf->Cell(60, 10, 'Numero du match :', 1, 0, 'C');
$pdf->Cell(60, 10, $unMatch['num_match'], 1, 1, 'C');
$pdf->SetX((150-60)/2);
//Ligne 3
$pdf->Cell(60, 10, 'Date du match :', 1, 0, 'C');
$pdf->Cell(60, 10, $unMatch['date_match'].' à '.$unMatch['heure_match'], 1, 1, 'C');
$pdf->SetX((150-60)/2);
//Ligne 4
$pdf->Cell(60, 10, 'Nom des équipes :', 1, 0, 'C');
$pdf->Cell(60, 10, $unMatch['equipe1'].' vs '.$unMatch['equipe2'], 1, 1, 'C');
$pdf->SetX((150-60)/2);
//Ligne 5
$pdf->Cell(60, 10, 'Nom des arbitres :', 1, 0, 'C');
$pdf->Cell(60, 10, $unMatch['arbitreP'].' et '.$unMatch['arbitres'], 1, 1, 'C');
$pdf->SetX((150-60)/2);
//Ligne 6
$pdf->Cell(60, 10, 'Montant des deplacements des arbitres :', 1, 0, 'C');
$pdf->Cell(60, 10, $unMatch['arbitreP'].' : '.$unMatch['montant_deplt_p'].' '.$unMatch['arbitres'].' : '
.$unMatch['montant_deplt_s'], 1, 1, 'C');
ob_end_clean();
$pdf->Output('I', 'Match.pdf');
```

?>