

# DOSSIER PROFESSIONNEL (DP)

*Nom de naissance*

▶ MARAIS

*Nom d'usage*

▶ Entrez votre nom d'usage ici.

*Prénom*

▶ CHRISTIAN

*Adresse*

▶ 155 ruelle Larivière  
97440 Saint-André  
La Réunion

## Titre professionnel visé

DEVELOPPEUR WEB ET WEB MOBILE

### MODALITE D'ACCES :

- ☒ Parcours de formation
- ☐ Validation des Acquis de l'Expérience (VAE)

## Présentation du dossier

Le dossier professionnel (DP) constitue un élément du système de validation du titre professionnel. **Ce titre est délivré par le Ministère chargé de l'emploi.**

Le DP appartient au candidat. Il le conserve, l'actualise durant son parcours et le présente **obligatoirement à chaque session d'examen.**

Pour rédiger le DP, le candidat peut être aidé par un formateur ou par un accompagnateur VAE. Il est consulté par le jury au moment de la session d'examen.

### Pour prendre sa décision, le jury dispose :

1. des résultats de la mise en situation professionnelle complétés, éventuellement, du questionnaire professionnel ou de l'entretien professionnel ou de l'entretien technique ou du questionnement à partir de productions.
2. du **Dossier Professionnel** (DP) dans lequel le candidat a consigné les preuves de sa pratique professionnelle
3. des résultats des évaluations passées en cours de formation lorsque le candidat évalué est issu d'un parcours de formation
4. de l'entretien final (dans le cadre de la session titre).

*[Arrêté du 22 décembre 2015, relatif aux conditions de délivrance des titres professionnels du ministère chargé de l'Emploi]*

### Ce dossier comporte :

- ▶ pour chaque activité-type du titre visé, un à trois exemples de pratique professionnelle ;
- ▶ un tableau à renseigner si le candidat souhaite porter à la connaissance du jury la détention d'un titre, d'un diplôme, d'un certificat de qualification professionnelle (CQP) ou des attestations de formation ;
- ▶ une déclaration sur l'honneur à compléter et à signer ;
- ▶ des documents illustrant la pratique professionnelle du candidat (facultatif)
- ▶ des annexes, si nécessaire.

*Pour compléter ce dossier, le candidat dispose d'un site web en accès libre sur le site.*



<http://travail-emploi.gouv.fr/titres-professionnels>

## Sommaire

### Exemples de pratique professionnelle

#### Développer la partie front-end d'une application web ou web mobile en intégrant les recommandations de sécurité p.5

- ▶ Le Pendu avec Bootstrap..... p.6
- ▶ Grand Raid avec Bootstrap et Leaflet ..... p.8
- ▶ Rugby Sochaux Club avec WordPress ..... p.10

#### Développer la partie back-end d'une application web ou web mobile en intégrant les recommandations de sécurité p.11

- ▶ Plugin Wordpress de disclaimer pour un Vapobar ..... p.12
- ▶ Surfwave -Projet MVC - PHP ..... p.14
- ▶ AATI -Système de gestion de stock – MVC-PHP..... p.16

#### Titres, diplômes, CQP, attestations de formation *(facultatif)* p.18

#### Déclaration sur l'honneur p.19

#### Documents illustrant la pratique professionnelle *(facultatif)* p.20

#### Annexes *(Si le RC le prévoit)* p.21

# **EXEMPLES DE PRATIQUE PROFESSIONNELLE**

## Activité-type 1

Développer la partie front-end d'une application web ou web mobile en intégrant les recommandations de sécurité

Exemple n°1 ► *Le pendu (C1,C2,C3)*

### 1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

Les compétences couvertes par ce projet sont :

C-1 - Maquetter une application

C-2 - Réaliser une interface utilisateur web statique et adaptable

C-3 - Développer une interface utilisateur web dynamique

J'ai réalisé au cours d'un exercice durant la formation le maquettage et le moteur JS d'un jeu du pendu. Le jeu affiche dans sur une ligne un mot à deviner avec des lettres manquantes. L'utilisateur doit deviner le mot et proposer les lettres manquantes en cliquant sur un clavier alphabétique.

J'ai réalisé l'application en utilisant le framework Bootstrap et la bibliothèque JQuery.

### 2. Précisez les moyens utilisés :

Support de formation, JQuery, Bootstrap

# DOSSIER PROFESSIONNEL (DP)

## 3. Avec qui avez-vous travaillé ?

En autonomie

## 4. Contexte

Nom de l'entreprise, organisme ou association ► **AFPAR**

Chantier, atelier, service ► **AFPAR**

Période d'exercice ► Du **12/10/2021** au **17/10/2021**

## 5. Informations complémentaires *(facultatif)*

Annexe 1 - captures d'écrans et code javascript

## Activité-type 1

**Développer la partie Front-End d'une application web ou web mobile en intégrant les recommandations de sécurité**

Exemple n°2 ► *Grand Raid avec API-REST, Bootstrap et Leaflet (c1,c2,c3)*

### 1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

Les compétences couvertes par ce projet sont :

C-1 - Maquetter une application

C-2 - Réaliser une interface utilisateur web statique et adaptable

C-3 - Développer une interface utilisateur web dynamique

Dans le cadre d'une Évaluation en Cours de Formation, j'ai réalisé un site fictif pour l'organisation du Raid.

Il avait pour objectif d'afficher sur une carte des informations et des données dynamiques obtenues sous format JSON à partir d'une API REST. L'internaute devait pouvoir consulter sous forme de curseur sur une carte la position en temps réel des coureurs fournie par l'API ainsi que les informations sur les différents points relais du parcours. Le site devait être dynamique et responsive.

Pour cette réalisation, l'utilisation de la charte graphique et d'éléments d'illustrations du Grand Raid a été imposée. Son identité graphique comprenait un logo et des codes couleurs ; les illustrations étaient constituées de photos des courses passées.

### 2. Précisez les moyens utilisés :

Dans le cadre de cette réalisation, il a été imposé l'usage de Bootstrap pour obtenir rapidement et facilement un site responsive et des éléments graphiques et recommandé l'usage de JQuery dans le développement.

Deux urls d'API ont été fournies pour simuler les données. La première contenait les informations sur les coureurs ainsi que leur position en temps réel. La deuxième contenait des informations sur les points relais. La récupération des données se fait via la méthode Xhr.

La fonctionnalité de mapping a été obtenue à partir de la bibliothèque JS Leaflet qui utilise les services de cartographie d'OpenStreetMap.

# DOSSIER PROFESSIONNEL (DP)

## 3. Avec qui avez-vous travaillé ?

En autonomie

## 4. Contexte

Nom de l'entreprise, organisme ou association ► **AFPAR**

Chantier, atelier, service ► **AFPAR**

Période d'exercice ► Du **19/10/2021** au **24/10/2021**

## 5. Informations complémentaires *(facultatif)*

Annexe 2 - Captures d'écran disponible

Annexe 3 – Extraits de code source JS



## Activité-type 1

**Développer la partie front-end d'une application web ou web mobile en intégrant les recommandations de sécurité**

*Exemple n°3 ► Wordpress – Rugby Sochaux (c1,c2,c3,c4)*

### 1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

Les compétences couvertes par ce projet sont :

C-1 - Maquetter une application

C-2 - Réaliser une interface utilisateur web statique et adaptable

C-3 - Développer une interface utilisateur web dynamique

C-4 - Réaliser une interface utilisateur avec une solution de gestion de contenu ou E-commerce

Dans le cadre d'une Évaluation en Cours de Formation, j'ai réalisé un site WordPress pour le compte d'un club sportif (réel ou fictif) que l'on devait choisir. Mon choix s'est porté sur le Rugby Sochaux Club.

Le site devait servir de vitrine à l'association qui souhaitait mettre en avant les événements qu'elle organisait. Il devait proposer une galerie photo des événements passés, deux formulaires de contact, un historique du club et des pages de blog respectant des objectifs SEO.

Le Rugby Sochaux Club dispose d'un conseil d'administration. Certains membres de ce conseil devaient pouvoir publier du contenu sur le site, d'autres devaient simplement pouvoir le consulter. Il fallait ainsi également mettre en place un système de gestion de compte avec des permissions différenciées.

### 2. Précisez les moyens utilisés :

Le CMS Wordpress, différents plug-ins, documentation Wordpress et supports de formations.

# DOSSIER PROFESSIONNEL (DP)

## 3. Avec qui avez-vous travaillé ?

En autonomie

## 4. Contexte

Nom de l'entreprise, organisme ou association ► **AFPAR**

Chantier, atelier, service ► **AFPAR**

Période d'exercice ► Du **15/11/2021** au **21/11/2021**

## 5. Informations complémentaires (facultatif)

Annexe 4 - Captures d'écran et extrait de code source PHP

## Activité-type 2

**Développer la partie back-end d'une application web ou web mobile en intégrant les recommandations de sécurité**

*Exemple n°4 ► Plugin WordPress de disclaimer pour Vapobar (C4,C6,C7,C8)*

### 1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

Les compétences couvertes par ce projet sont :

C-4 – Réaliser une interface utilisateur avec une solution de gestion de contenu ou E-commerce

C-6 – Développer les composants d'accès aux données

C-7 – Développer la partie back-end d'une application web ou web-mobile

C-8 – Elaborer et mettre en œuvre des composants dans une application de gestion de contenu ou e-commerce

Dans une Évaluation en Cours de Formation, j'ai réalisé un plug-in/extension pour un site de Vapobar utilisant le CMS WordPress. L'extension génère un disclaimer sous forme de pop-up et vérifie l'âge des internautes avant tout accès au site. Ce disclaimer affiche un message personnalisé et redirige le mineur en dehors du site vers une page qui aura été renseignée.

### 2. Précisez les moyens utilisés

Documentation WordPress, supports de formations, tutoriels

# DOSSIER PROFESSIONNEL (DP)

## 3. Avec qui avez-vous travaillé ?

En autonomie

## 4. Contexte

Nom de l'entreprise, organisme ou association ► **AFPAR**

Chantier, atelier, service ► **AFPAR**

Période d'exercice ► Du **13/12/2021** au **17/12/2021**

## 5. Informations complémentaires *(facultatif)*

Annexe 5 - Capture d'écran et extraits de code source PHP.

## Activité-type 2

**Développer la partie back-end d'une application web ou web mobile en intégrant les recommandations de sécurité**

Exemple n°5 ► *Surfwave - MVC*

### 1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

Les compétences couvertes par ce projet sont :

C-1 - Maquetter une application

C-2 - Réaliser une interface utilisateur web statique et adaptable

C-3 - Développer une interface utilisateur web dynamique

C-5 – Créer une base de données

C-6 – Développer les composants d'accès aux données

C-7 – Développer la partie back-end d'une application web ou web mobile

Dans le cadre d'une Évaluation en Cours de Formation, j'ai réalisé un site en PHP 7.4 selon le modèle architectural Model-View-Controller.

Dans le cahier des charges du projet, il s'agissait de rendre utilisable une maquette de site qui nous est fournie. J'ai eu pour mission de concevoir et compléter le modèle physique de données, de coder un module de connexion, une interface d'administration et ses opérations de traitements, de créer des CRUD sur les produits, abonnements et équipiers affichés sur le site.

### 2. Précisez les moyens utilisés :

Documentation PHP, supports de cours, tutoriels

# DOSSIER PROFESSIONNEL (DP)

## 3. Avec qui avez-vous travaillé ?

En autonomie

## 4. Contexte

Nom de l'entreprise, organisme ou association ► **AFPAR**

Chantier, atelier, service ► **AFPAR**

Période d'exercice ► Du **01/01/2022** au **10/01/2022**

## 5. Informations complémentaires (facultatif)

Annexe 6 - Capture et le code source PHP.

## Activité-type 2

**Développer la partie back-end d'une application web ou web mobile en intégrant les recommandations de sécurité**

*Exemple n°3 ► Application de gestion de stock PHP*

### 1. Décrivez les tâches ou opérations que vous avez effectuées, et dans quelles conditions :

Les compétences couvertes par ce projet sont :

C-1 - Maquetter une application

C-2 - Réaliser une interface utilisateur web statique et adaptable

C-3 - Développer une interface utilisateur web dynamique

C-5 – Créer une base de données

C-6 – Développer les composants d'accès aux données

C-7 – Développer la partie Backend d'une application web ou web mobile

Dans le cadre de la Période en Entreprise, j'ai réalisé une application de gestion des stocks en PHP 7.4 utilisant apache2 et MySQL et selon le modèle architectural Model-View-Controller.

J'ai eu pour mission de concevoir la BDD, maquetter l'application, coder un module de connexion, un système de gestion de permission, une interface d'administration et ses opérations de traitement, des CRUD sur des articles, des utilisateurs, des formations, des stocks.

Je devais aussi coder ou intégrer à l'application des solutions permettant de proposer une fonctionnalité de mailing et de génération de pdf pour le suivi des stocks et une fonctionnalité d'affichage graphique pour les statistiques.

J'ai eu pour dernière mission d'installer et de configurer les logiciels nécessaires au fonctionnement de l'application sur le serveur linux de l'entreprise.

### 2. Précisez les moyens utilisés :

Documentations : PHP, sur la gestion des stocks, documentation sur la sécurité, forums, tutoriels, documentations sur l'installation et la configuration d'apache2, MySQL, PHP 7.4.

Bibliothèques : FPDF, PHPMAILER

Framework : Bootstrap 4.6

Logiciels : Putty, Bash, Proftpd, VSCode, Filezilla, Opera, Firefox

# DOSSIER PROFESSIONNEL (DP)

## 3. Avec qui avez-vous travaillé ?

En autonomie. De la documentation était fourni par mon tuteur sur la partie linux

## 4. Contexte

Nom de l'entreprise, organisme ou association ► **AATI**

Chantier, atelier, service ► Service informatique

Période d'exercice ► Du 14/02/2022 au 22/04/2022

## 5. Informations complémentaires (facultatif)

Annexe 7 – Captures d'écran et extraits de code PHP



# DOSSIER PROFESSIONNEL (DP)

## Titres, diplômes, CQP, attestations de formation

(facultatif)

Intitulé	Autorité ou organisme	Date
Attestation de réussite de SecNum Académie en annexe 8	Agence Nationale de la Sécurité des Systèmes d'Informations	18/02/2022
Certificat de réussite – Apprenez à programmer avec Javascript en annexe 9	OpenClassroom	02/05/2022
Certificat de réussite – Concevez votre site web avec PHP et MySQL en annexe 10	OpenClassroom	02/05/2022
Certificat de réussite – Simplifiez-vous le CSS avec Sass en annexe 11	OpenClassroom	29/09/2021
Certificat de réussite – Apprenez à créer votre site web avec HTML5 et CSS3	OpenClassroom	08/12/2019
Cliquez ici.	Cliquez ici pour taper du texte.	Cliquez ici pour sélectionner une date.
Cliquez ici.	Cliquez ici pour taper du texte.	Cliquez ici pour sélectionner une date.
Cliquez ici.	Cliquez ici pour taper du texte.	Cliquez ici pour sélectionner une date.
Cliquez ici.	Cliquez ici pour taper du texte.	Cliquez ici pour sélectionner une date.

---

## DOSSIER PROFESSIONNEL (DP)

---

---

## DOSSIER PROFESSIONNEL (DP)

### Déclaration sur l'honneur

Je soussigné(e) [prénom et nom] *Christian MARAIS* ,  
déclare sur l'honneur que les renseignements fournis dans ce dossier sont exacts et que je  
suis l'auteur(e) des réalisations jointes.

Fait à *Saint-André* le *26/04/2022*  
pour faire valoir ce que de droit.

Signature :

# DOSSIER PROFESSIONNEL (DP)

## Documents illustrant la pratique professionnelle

*(facultatif)*

Intitulé
Annexe 1 – Le Pendu – Captures d’écran et extraits de code
Annexe 2 – Grand Raid – Captures d’écran
Annexe 3 – Grand Raid - Code Javascript
Annexe 4 – Rugby Sochaux Club avec thème enfant
Annexe 5 – Plugin WordPress
Annexe 6 - Surfwave
Annexe 7 – Système de gestion de stock – AATI PHP
Annexe 8 – Attestation de réussite de SecNum Académie
Annexes – Certificats de réussite OpenClassroom

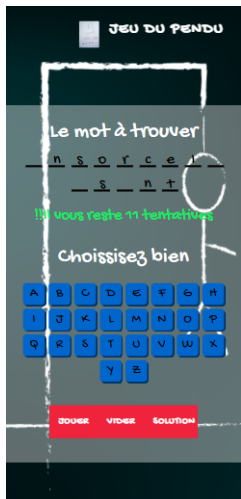
## DOSSIER PROFESSIONNEL (DP)

### ANNEXES

*(Si le RC le prévoit)*

## Annexe 1 – Le Pendu -Js

Captures d'écran version web et web mobile.



Extraits des deux méthodes principales du moteur. Une qui place le mot dans la grille et une autre qui vérifie la correspondance avec le mot cherché.

```
function placerMot(){// procédure qui va placer une partie du mot dans la grille
  reinitialiser();

  bouton2.addEventListener("click",verifierMot,false);

  let aleatoire=Math.random();
  indiceMot=Math.floor(aleatoire*(tableau.length));
  for(i=0;i<(tableau[indiceMot]).length;i++){
    input[i+1].value="";
    aleatoire=Math.random();
    if(aleatoire>difficulte){
      input[i+1].value=tableau[indiceMot][i];
    }
  }
}
```

```
function verifierMot(){//procédure qui va verifier la correspondance entre la lettre tapée et les lettres du
mots

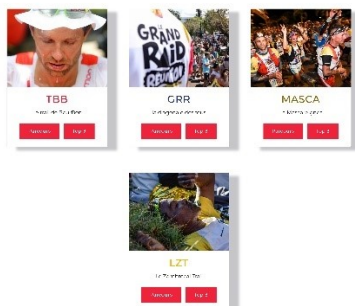
  if(continuer!=0){
    let poursuivre=0;
    for(let a=0;a<tableau[indiceMot].length;a++){
      if(input[a+1].value!=tableau[indiceMot][a]){
        input[a+1].style.color="yellow";
      }else{
        poursuivre++;
      }
    }
    if(poursuivre===tableau[indiceMot].length){
      continuer=0;
      document.querySelector("#compteur").innerHTML="<p>Bravo!! J'y crois pas! Vous avez réussi :(</p>";
      document.querySelector(".boite").style.background="url(images/oui.gif) no-repeat";
    }
  }
}
```

## Annexe 2 – Grand Raid

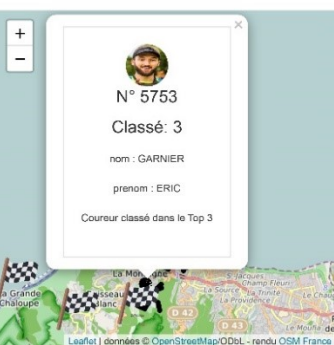
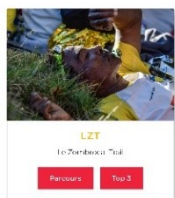
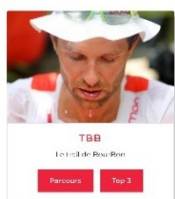
### Aperçu du site - versions Web et mobile



Découvrez les 4 courses mythiques



Découvrez les 4 courses mythiques



## Annexe 3– Grand Raid- source

On récupère les données de l'API via la méthode XHR.

```
/* *****methodes***** */
/**
 * M Elle récupère les données du JSON et accueille les méthodes qui vont les utiliser avec leur
 * parametre respectif; respecter l'ordre d'entree de la méthode pour l'ordre d'entréedu paramètre
 * O Rien
 * I String, tableau,tableau
 * @param {String} url
 * @param {[]} methodes
 * @param {[]} donneesDeLaMethode
 */
function recupererDonnee(url,methodes,donneesDesMethodes){//récupère les données au format JSON à
partir de l'url et passe son resultat à des méthodes qui reçoivent d'autres donneesDesMethodes
// "https://polenumerique.re/dl/dwwwm2021/ws/m1s3/";
// "https://polenumerique.re/dl/dwwwm2021/ws/m1s4/?q=pc";
var xhr= new XMLHttpRequest();

    xhr.onreadystatechange= function (){
        if(this.readyState!=4 || this.status!=200){//etat pour en cours de chargement qui se charge de
l'afficher sur la page avec des spinners
            $("#demo").html('<td class="text-center"><div class="spinner-border" role="status"><span
class="sr-only">Loading...</span></div></td><td class="text-center"><div class="spinner-border"
role="status"><span class="sr-only">Loading...</span></div></td><td class="text-center"><div
class="spinner-border" role="status"><span class="sr-only">Loading...</span></div></td>');
        }
        setTimeout(() => {
            if(this.readyState!=4 || this.status!=200){//etat pour echec de chargement avec message
d'erreur sur la page
                $("#demo").html("<p style='color:orangered'>Le temps de chargement a été trop long.
Veuillez recharger la page</p>");
            }
        }, 10000);

        if(this.readyState==4 && this.status==200){//surveille les changements d'états et attends un
succès
            elementJson=xhr.response;//transmet la valeur reçu en cas de succès
            for(let i=0;i<methodes.length;i++){ //passe aux méthodes leurs donneesDesMethodes respectif
qui a été fournie sous forme de tableau
                methodes[i](elementJson,donneesDesMethodes[i]);
            }
        }
    }
    xhr.open("GET",url,true);
    xhr.responseType="json";
    xhr.send();
};
```



Extrait de code de la méthode `insererElement`. On manipule le DOM pour créer un tableau listant les coureurs et les points d'étapes du parcours.

```
function insererElement(formatJson,donneesDeLaMethode){//procedure qui insère l'ensemble des données
dans la page
    let cles=donneesDeLaMethode[0];//clés des valeurs du JSON ex: lat, lon...
    let containerId=donneesDeLaMethode[1];//id du container ou on va insérer les td
    let baliseParente=donneesDeLaMethode[2];//balise th contenant les td
    let balise=donneesDeLaMethode[3];//balise du td
    let rajoutElements=donneesDeLaMethode[4];// boolean qui dit si on rajoute ou non des éléments
    let container=document.getElementById(containerId);// on selectionne le conteneur des nouveaux
éléments html
    container.innerHTML=""; // on le vide en début de méthode
    let newElementHtml= []; // tableau elements html destinés à contenir les tr créés
    let newSousElementHtml=[]; //tableau element html -idem- td créés

    //-----personnalisation du tHeader du tableau, on rajoute les colonnes classement et
photo-----
    if(!rajoutElements){ //version normale
        $("#titre").html('<th scope="col">'+cles[0]+'</th><th scope="col">Latitude</th><th
scope="col">Longitude</th><th scope="col">Altitude</th>')
    }else{
        let texte="";
        for(cle of cles){
            texte+='<th scope="col">'+cle+'</th>';
        }
        $("#titre").html('<th scope="col">#</th><th scope="col"></th>'+texte);
    }

    let p=1;
    for (let objetDuJson in formatJson){ //Crée les lignes du tableau

        newElementHtml[objetDuJson]=document.createElement(baliseParente); //on crée la ligne qui va
accueillir les cellules contenant les infos
        container.appendChild(newElementHtml[objetDuJson]); //on insere la ligne dans le corps du
tableau

        let i=0;
        //-----rajout de cellule dans une ligne. On fait un rajout de code personnalisé à
cette demande étant donné que les elements ne sont pas des elements du fichier Json de base-----
        if(rajoutElements){
            //----- rajout du numéro de classement -----
            newSousElementHtml[i]=document.createElement(balise); //on ajoute la cellule supplémentaire
contenant la photo en premier
            newElementHtml[objetDuJson].appendChild(newSousElementHtml[i]); // on l'affecte à la ligne
tr

            newSousElementHtml[i].innerHTML='<span>'+p+'</span>';
            //----- rajout de la photo du coureur-----
            newSousElementHtml[i]=document.createElement(balise); //on ajoute la cellule supplémentaire
contenant la photo en premier
            newElementHtml[objetDuJson].appendChild(newSousElementHtml[i]); // on l'affecte à la ligne
tr

            newSousElementHtml[i].innerHTML='';
            //-----
        }
        //-----on créé les HTML td du tableau automatiquement à partir du JSON-----
        for(let cle of cles){ //idem crée une ligne du tableau avec les cellules contenant les éléments
            newSousElementHtml[i]=document.createElement(balise);
            newElementHtml[objetDuJson].appendChild(newSousElementHtml[i]);
        }
    }
}
```

```
// -----on en profite pour créer des boutons dans le tableau qui vont permettre
de centrer le coureur ou l'étape sur la map-----
if (cle==cles[0]){// on crée le lien sur le numéro de dossard clé[0]
    newSousElementHtml[i].innerHTML='<a class=" badge bg-tertiaire"type="button"
href="#maCarte" id="centrer'+i+' ">'+formatJson[objetDuJson][cle]+'</a>';
    newSousElementHtml[i].addEventListener("//on ajoute un écouteur d'évènements
        "click",
        function(){

macarte.setView([formatJson[objetDuJson]["lat"],formatJson[objetDuJson]["lon"],20]);// Méthode qui
centre sur la map

        },
        false
    );

}
}else{
    newSousElementHtml[i].innerText=formatJson[objetDuJson][cle];
}
i++;
}
p++;//p s'inscrimente à chaque fois qu'une ligne est créé,c'est à dire qu'on passe à
l'objetDuJson suivant
}

    newSousElementHtml[i].innerHTML='<a class=" badge bg-tertiaire"type="button"
href="#maCarte" id="centrer'+i+' ">'+formatJson[objetDuJson][cle]+'</a>';
    newSousElementHtml[i].addEventListener("//on ajoute un écouteur d'évènements
        "click",
        function(){

macarte.setView([formatJson[objetDuJson]["lat"],formatJson[objetDuJson]["lon"],20]);// Méthode qui
centre sur la map

        },
        false
    );

}
}else{
    newSousElementHtml[i].innerText=formatJson[objetDuJson][cle];
}
i++;
}
}
```

On initialise la map.

```
/**
 * M génère la map et l'insere dans le container ayant l'ID qui lui a été passé
 * O la variable macarte
 * I 3=4 donneesDelaMethode ci-dessous
 * @param {string} mapId
 * @param {double} lat
 * @param {double} lon
 * @param {int} zoom
 */
function initialiserMap(mapId,lat,lon,zoom){// elle a besoin de l'id de la div, d'une latitude et
longitude pour s'initialiser
    // Créer l'objetDuJson "macarte" et l'insérer dans l'élément HTML qui a l'ID "map"
    macarte = L.map(mapId).setView([lat, lon], zoom);
    // Leaflet ne récupère pas les cartes (tiles) sur un serveur par défaut. Nous devons lui préciser
où nous souhaitons les récupérer. Ici, openstreetmap.fr
    L.tileLayer('https://{s}.tile.openstreetmap.fr/osmfr/{z}/{x}/{y}.png', {
        // Il est toujours bien de laisser le lien vers la source des données
        attribution: 'données © <a href="//osm.org/copyright">OpenStreetMap</a>ODbL - rendu <a
href="//openstreetmap.fr">OSM France</a>',
        minZoom: 1,
        maxZoom: 20
    }).addTo(macarte);
}
```

## Annexe 4 – Wordpress – Rugby Sochaux

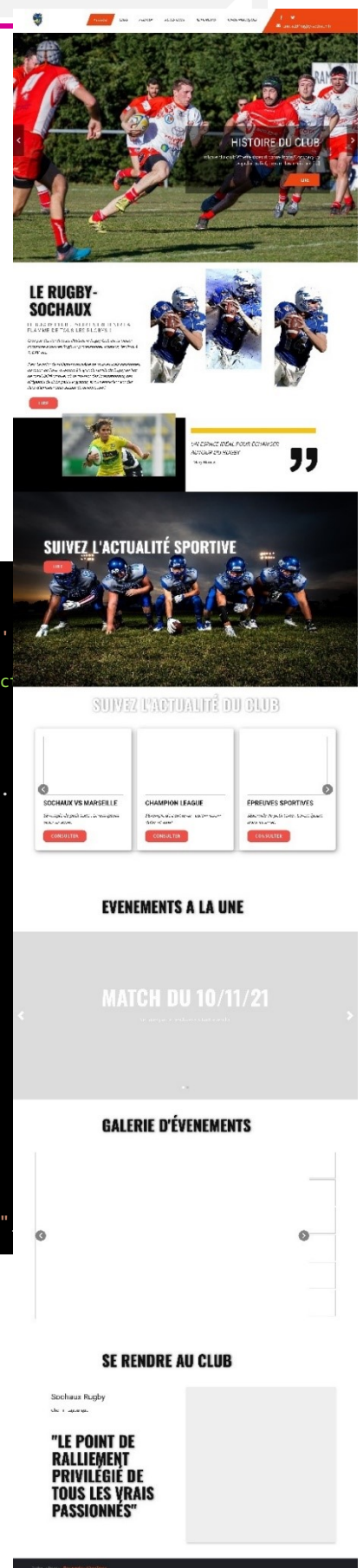
Contenu de notre fichier functions.php de notre thème enfant.

Wordpress au travers des hooks (crochets) d'actions qui permettent d'utiliser des fonctions personnalisés à un moment précis de l'exécution de Wordpress ou des hooks de filtres qui permettent de modifier des données retournés d'autres fonctions, nous permet de personnaliser son comportement.

```
<?php
/** On indique qu'il faut charger notre feuille de style du thème enfant */
function my_enqueue_styles(){
    wp_enqueue_style( 'parent-style', get_template_directory_uri() . '/style.css' );
}
add_action( 'wp_enqueue_scripts', 'my_enqueue_styles' ); // on accroche notre fonction à la fonction wp_enqueue_scripts
/* Ajout du fichier css personnalisé */

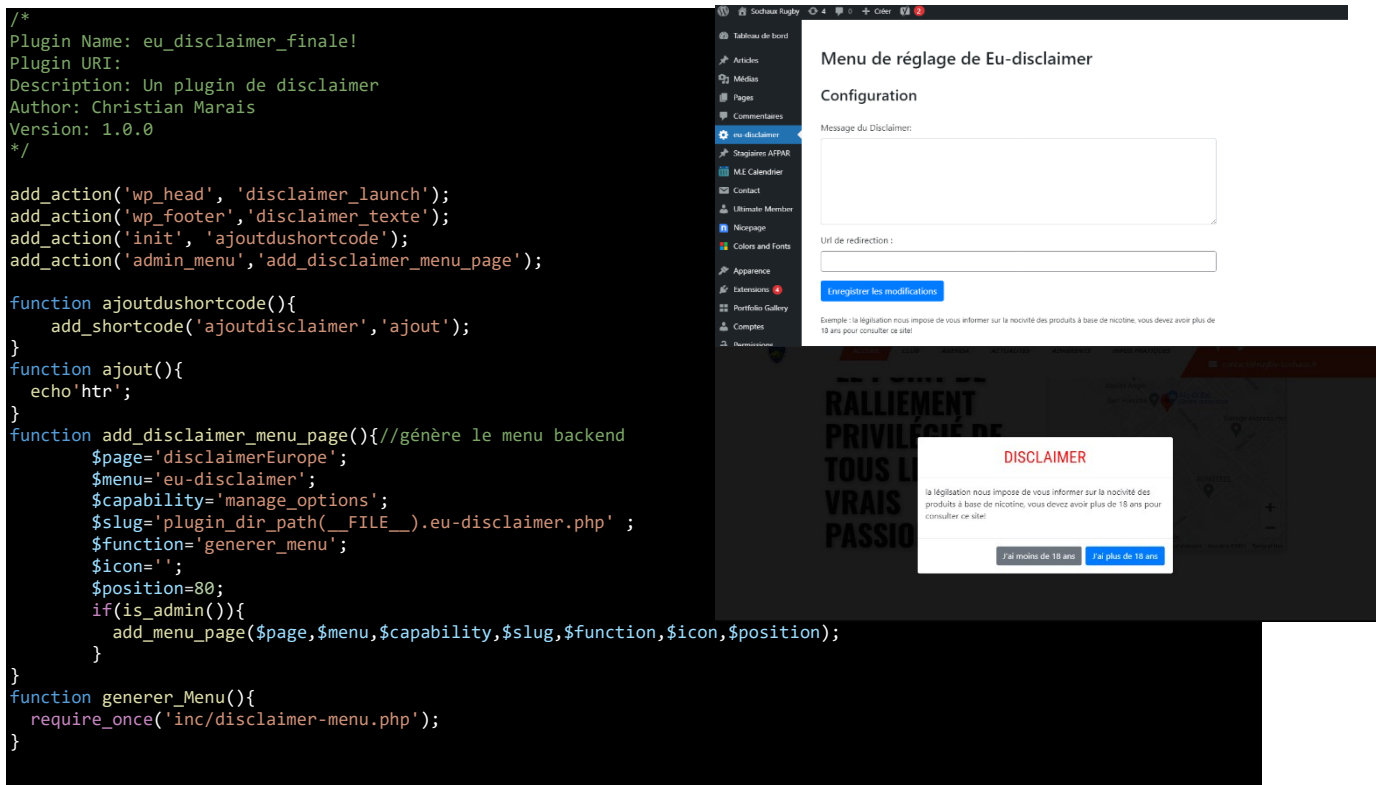
function custom_login(){
    echo '<link rel="stylesheet" href="' . get_bloginfo('stylesheet_directory') . 'style.css' type="text/css" media="all" />';
}
add_action( 'login_head', 'custom_login' );
/* Personnalisation du lien de la page de connexion */
function my_login_logo_url() {
    return get_bloginfo( 'url' );
}
add_filter( 'login_headerurl', 'my_login_logo_url' );
function my_login_logo_url_title() {
    return 'Ti Web Compétences';
}
add_filter( 'login_headertitle', 'my_login_logo_url_title' );
/* Pré-cocher la case "Se souvenir de moi" */
function login_checked_remember_me() {
    add_filter( 'login_footer', 'rememberme_checked' );
}
add_action( 'init', 'login_checked_remember_me' );

function rememberme_checked() {
    echo "<script>document.getElementById('rememberme').checked = true;</script>";
}
```



## Annexe 5 – Plugin WordPress – Vapobar

Grâce aux Hooks d'action on ajoute nos méthodes au processus d'exécution de WordPress pour générer le disclaimer en Frontend et son menu d'administration en backend.



The image displays the plugin's PHP code on the left and its WordPress admin interface on the right. The code defines the plugin's metadata, hooks for the front-end (wp\_head, wp\_footer, init) and back-end (admin\_menu), and functions for adding a shortcode, generating the disclaimer message, and creating an admin menu item. The admin interface, titled 'Menu de réglage de Eu-disclaimer', includes a configuration page with a text area for the disclaimer message and a redirect URL field. Below the configuration, a preview of the disclaimer as it appears on the website is shown, featuring a red 'DISCLAIMER' header and a warning about nicotine products for users under 18.

```
/*
Plugin Name: eu_disclaimer_finale!
Plugin URI:
Description: Un plugin de disclaimer
Author: Christian Marais
Version: 1.0.0
*/

add_action('wp_head', 'disclaimer_launch');
add_action('wp_footer', 'disclaimer_texte');
add_action('init', 'ajoutdushortcode');
add_action('admin_menu', 'add_disclaimer_menu_page');

function ajoutdushortcode(){
    add_shortcode('ajoutdisclaimer', 'ajout');
}

function ajout(){
    echo'htr';
}

function add_disclaimer_menu_page(){//génère le menu backend
    $page='disclaimerEurope';
    $menu='eu-disclaimer';
    $capability='manage_options';
    $slug='plugin_dir_path(__FILE__).eu-disclaimer.php' ;
    $function='generer_menu';
    $icon='';
    $position=80;
    if(is_admin()){
        add_menu_page($page,$menu,$capability,$slug,$function,$icon,$position);
    }
}

function generer_Menu(){
    require_once('inc/disclaimer-menu.php');
}
```

A l'activation du plugin on crée les tables utilisées par notre plugin et à sa désactivation on les supprime.

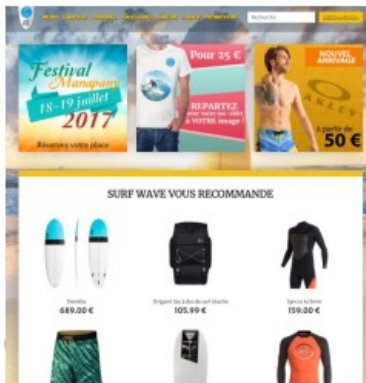
```
/**
 * fonction qui va créer la table disclaimer_options à l'activation
 * et la supprimer à la désactivation
 */
$generTable=function (){
    if (class_exists('DisclaimerGestionTable')){
        $gener_table = new DisclaimerGestionTable();
    }
    if (isset($gener_table)){
        register_activation_hook(__FILE__, array($gener_table, 'creerTable'));
        register_deactivation_hook(__FILE__, array($gener_table, 'supprimerTable'));
    }
};
$generTable();
```

La méthode insererDansTable nous permet de faire un update sql dans sur notre table dans la BDD de wordpress.

```
function insererDansTable($contenu,$url){
    global $wpdb;
    $table_disclaimer=$wpdb->prefix.'disclaimer_options';
    $sql=$wpdb->prepare("
    UPDATE $table_disclaimer
    SET message_disclaimer = '%s',redirection_ko='%s' WHERE id_disclaimer=%s",$contenu,$url,1
    );
    $wpdb->query($sql);
}
```

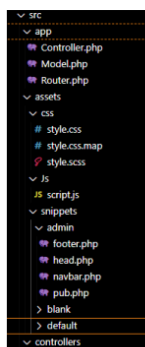
## Annexe 6 – Surfwave – MVC

Captures du site.

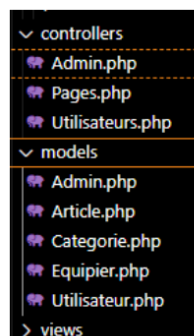


Arborescence du site

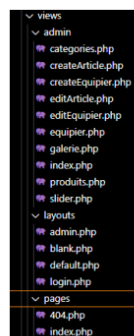
Arborescence des assets



Arborescence des contrôleurs



arborescence des views



Description du fonctionnement Model – View – Controller à travers les classes principales Model, Controller et Router.

**Model :**

- o La méthode **getConnection()** qui initialise la connexion à la base de données.

```
protected function getConnection(){//initialise la connexion par la classe PDO
    $this->connexion=null;
    try{
        $this->connexion = new PDO('mysql:host='.$this->host.'; dbname='.$this->dbname, $this->username,$this->password);
        $this->connexion->exec('set names utf8');
    }catch(Exception $e){
        echo 'Erreur : '.$e->getMessage();
    }
}
```

Elle est appelée dans la méthode constructeur des classes filles models pour être instancié à chaque nouvelle instance de la classe fille.

```
class Article extends Model{
    public function __construct(){
        $this->table='tarif';
        $this->getConnection();
    }
}
```

Le modèle article possède ses méthodes personnalisés pour effectuer ses requêtes à la BDD.

```
15
16     public function getAlaune(){
17         $sql='SELECT * FROM produits WHERE alaune = 1';
18         $req=$this->connexion->prepare($sql);
19         $req->execute();
20         return $req;
21     }
22
```

2. **Contrôleur** : Classe mère qui sert de base aux contrôleurs fils comme par ex. : **admin**. Elle contient les méthodes qui seront utilisées systématiquement par les classes filles.

- Méthode **loadModel()** :

```
protected function loadModel(string $model){//créé une instance de model pour pouvoir faire appel à ses méthodes
//sert uniquement à récupérer de la donnée
require_once(ROOT.'src/models/'.$model.'.php');
$this->$model = new $model();
}
```

Méthode qui fait les « require » des models chargés récupérés les données et les instancie. Il prend en input le nom du model avec en uppercase la première lettre.

- Méthode **render()** :

```
protected function render(string $fichier,array $data = [], $class=null){// sert uniquement à l'affichage en faisant
extract($data);//extraire les données d'un tableau associatif sous forme de variables. On accède ainsi aux données
if(!empty($_SESSION['login'])){
    $this->theme='admin';
}
if($class==null){
    $class=get_class($this);
}
controller::loadComponent();//charge les composants ou éléments html headers... dans des constantes pour pouvoir
ob_start();

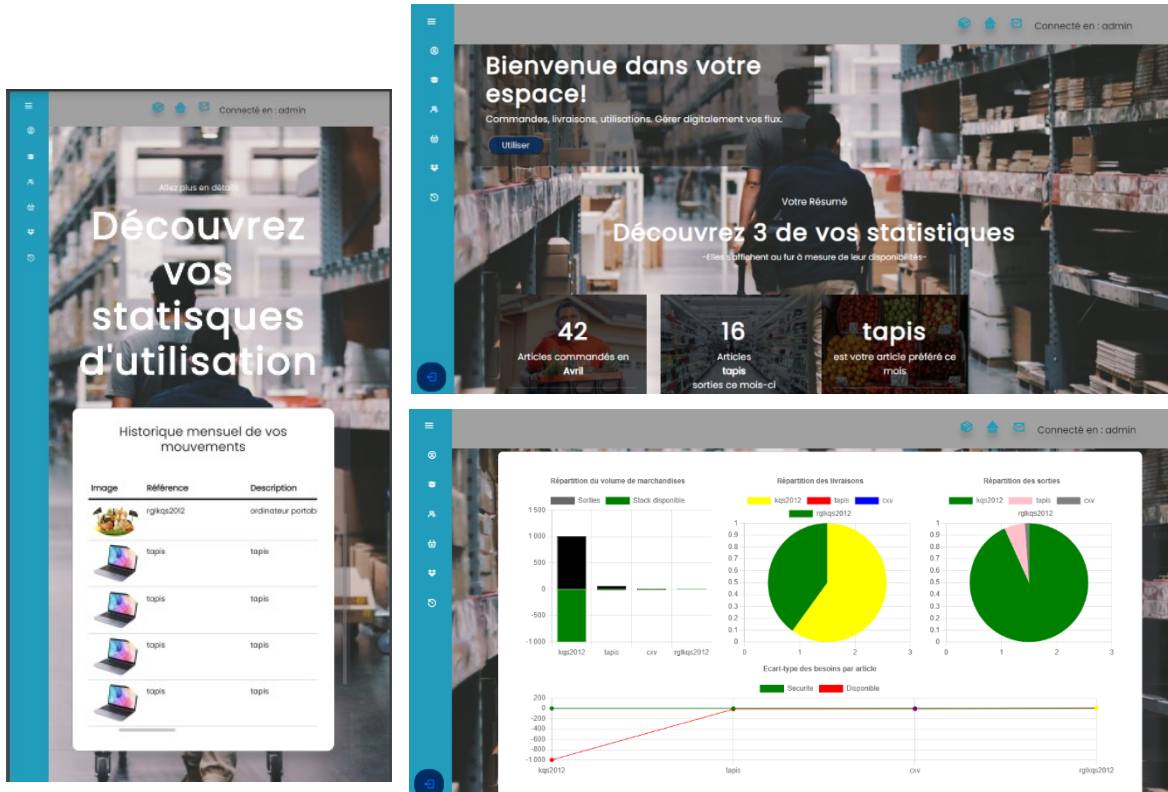
require_once(ROOT.'src/views/'.$strtolower($class).'/'.$fichier.'.php');
$content=ob_get_clean();//les echos et require de la vue ont été placés dans une variable qui est dans le layout
require_once(ROOT.'src/views/layouts/'.$this->layout.'.php');//une fois la variable déclarée on importe le layout
}
```

Il fait le require des view à partir des paramètres fournis :

1. Le string du nom de la VIEW
2. Le tableau de données utilisé dans les variables de la VIEW
3. En facultatif, le nom du dossier dans lequel se trouve la view. Par défaut le nom de la classe en cours obtenu avec la méthode : **get\_class(\$this)**.



## Annexe 7 – Système de gestion de stock – MVC AATI



Extrait de code sur l'url rewriting dans le HTACCESS. Il redirige les requêtes url vers l'unique point d'entrée le fichier index.php

RewriteEngine On

#requete de rewrite sauf les ###.## comme.css

RewriteRule ^([a-zA-Z0-9\-\\_\.\~\\*\.\:]\*)\$ index.php?p=\$1 [L]

Le point d'entrée charge la classe Autoload qui charge la classe router qui utilise la méthode setRedirection qui est chargé des redirections vers le bon contrôleur et la bonne méthode.

```
private static function setRedirection(){//on met en place les redirections qui vont faire le require du controleur et l'appel à la méthode demandée

    $params=PARAMS;// on récupère l'url placé en get dans le paramètre p par le htaccess. PARAMS est une constante définie par la classe Security
    // on vérifie le controleur
    if($params[0]!=""){//si nous avons un uri
        $controller = ucfirst(Security::classAlias($params[0]));// A partir du paramètre 0 de l'url on définit le controleur à appeler
        $action= isset($params[1]) ? $params[1] : 'index';// on definit la méthode à appeler et met une méthode par défaut
        $controller_path= Sources::path();//on récupère le chemin complet vers le controleur
        //$controller_path= ROOT.'src/controllers/'; Décommenter pour ne pas utiliser la méthode Sources::path
        if(!file_exists($controller_path.$controller.'.php')){//si le controleur n'est pas trouvé
```

```
        header('Location: '.BASE_URI.'pages/erreur404'); // on redirige vers une page 404
    }
} else { // si nous n'avons pas d'URI, on redirige à l'accueil
    $controller="Pages"; // on définit le contrôleur par défaut
    $action="index"; // on définit la méthode par défaut
}
// on instancie le contrôleur (chargement automatique du contrôleur via l'autoloader)
$controller=Autoload::classToNamespace($controller);
$controller = new $controller();

// on vérifie la méthode
if(method_exists($controller,$action)){ // si la méthode existe
    unset ($params[0],$params[1]); // On prépare le tableau des paramètres à envoyer aux méthodes.
    // On retire le contrôleur et la méthode des paramètres pour garder que ceux qu'on doit envoyer aux méthodes
    call_user_func_array([$controller,$action],$params); // fonction qui fait appel à la méthode
    // d'une classe en lui envoyant un paramètre
} else { // si la méthode n'existe pas
    header('Location: '.BASE_URI.'pages/erreur404'); // on redirige vers une page 404
}
}
```

Le contrôleur fait un appel aux classes modèles pour interagir avec la BDD. LoadModel() est la méthode utilisée par les contrôleurs pour instancier un modèle.

```
/**
 * M permet d'initialiser les classes modèles choisies qui devront communiquer avec la bdd
 * O rien
 * I le string du nom du modèle ex: le model ;
 */
protected function loadModel(string ...$model){ // crée une instance de modèles transmis pour pouvoir
    // faire appel à leurs méthodes

    foreach($model as $item){ // sert uniquement à récupérer de la donnée
        $item=ucfirst($item); // On commence le nom de classe par une majuscule pour respecter le
        // nom de fichier qui correspond à celui utilisé pour l'appel de classe donc impérativement en majuscule
        $itemNamespace=Autoload::classToNamespace($item); // on récupère le namespace correspondant
        $this->$item = new $itemNamespace(); // on instancie le modèle
    }
}
```

Méthode de la classe MODEL appelée par les classes modèles filles pour faire une requête de type select.

```
/**
 * M récupère une occurrence selon la valeur d'un champ
 * O un tableau de données
 * I string du nom de colonne et string de sa valeur
 * @param string champ nom du champ
 * @param string valeur valeur du champ
 */
protected function getBy($champ,$valeur){
    $data=array($champ,$valeur); // on passe l'ensemble des données dans un array
    $this->secure($data); // pour les sécuriser par la méthode secure
    extract($data); // on récupère les variables champ et valeur
    $sql="SELECT * FROM ".$this->table." WHERE ".$champ." = ?"; // on définit la requête de base
    $this->testSql($sql,$valeur); // on passe le sql et les données pour le suivi du debug
    $query=$this->connexion->prepare($sql); // on prépare la requête
    $this->setDataNull($valeur); // on met les valeurs à null si vide pour sécuriser la requête
    $query->execute(array($valeur)); // on envoie les valeurs à pdo
    return $query->fetch(); // on récupère les valeurs
}
```



# DOSSIER PROFESSIONNEL (DP)

Méthode utilisée par le modèle statistique avec des appels de procédures.

```
public function fluxComCurrYearByUserGrpByArticle($idUser){
    $sql= 'call flux_commande_current_year_by_user_by_article(?)';
    $req=$this->connexion->prepare($sql);
    $req->execute(array($idUser));
    return $req->fetchAll();
}
```

Extrait de code principal intervenant dans la gestion des permissions en vérifiant les accès users.

```
/**Description : fonction qui gère la vérification des permissions des users par page
 * @param string nomPage string du nom de page
 * @param array permString de string de nom des permissions nécessaires ex :
 * ['LECTURE','CREATION', 'MODIFICATION', 'SUPPRESSION']
 * @param boolean egaliteStrict facultatif si on veut définir des permissions strictes par exemple
seul création est autorisé.Si une personne a un score supérieure elle sera quand meme refusée.
 * peut servir a affiner les permissions sur des pages ou l'affichage ou non de blocs html
 * @return boolean true si le score de permission de l'utilisateur est supérieure à celle demandée
 */
protected function checkUserContenuPermission($nomPage,$permString,$egaliteStrict=false){
    $allowedScore=0;// on met un score autorisé de 0
    foreach($permString as $perm){// pour chaque string de permission entré
        if($this->translatePermStringToValue($perm)>=0){//on traduit le string en score et s'il
est bon et >=0
            $allowedScore=$allowedScore+$this->translatePermStringToValue($perm);// on le rajoute
au score autorisé
        }
    }
    $idUser=$this->translatePermStringToValue(strtoupper($nomPage),Security::PERMISSION_CONTENU);//on récupère l'id de page
à partir du nom

    $results=false;
    $roleFromDb = new Role();// on initie une instance de role
    if(!empty($_SESSION['role'])){//si la session n'est pas vide
        $roles= explode(',',$_SESSION['role']);// on obtient tous les roles enrgistrés dans la
session

        foreach($roles as $role){// pour chaque role
            $score=0;// on définit un score de départ de 0
            $userRole=$roleFromDb->getOneRole('code_role',$role);// puis on récupère les infos du
role

            $userPermission=@json_decode($userRole["permission"]);//on récupère plus précisemnt
l'ensemble des permissions du role sous forme d'objet
            if(gettype($userPermission)=='array'){
                foreach($userPermission as $permission){//pour chaque permission de page

                    ($permission->id == $idUser && $permission->permissions
>=$score)?$score=$permission->permissions:'';// si l'id de page correspond et si son score de permission
est supérieure au score en cours on l'enregistre et obtient le niveau de permission accordé le plus
élevé

                }
            }
        }
        if($egaliteStrict==true){// si on souhaite une valeur de permission égale
            $results = $score==$allowedScore;// si le score de l'utilisateur est strictement égale
à celui demandé on retourne true
        }else{// si on ne souhaite pas d'égalité stricte

            $results = ($score>=$allowedScore);// si le score de l'utilisateur est suffisant on
retourne true
        }
    }
    return $results;
}
```

Extrait du code de la view layout qui int gre les autres views appel s par les contr leurs.

```
<!DOCTYPE html>
<html lang="en">
  <head>

    <?=HEAD?>

    <link rel="stylesheet" href="<?=BASE_URI?>src/assets/css/backend-style.css">
  </head>

  <body>

    <?=(isset($_POST['mail']) && !empty(MAIL))? MAIL:""?>

    <header>
      <?=NAVBAR?>
    </header>
    <?=SIDEBAR?>
    <main>
      <section>
        <div class="home_content">
          <div class="text top_title">
            <h1>
              Page des <?=PAGE?>
            </h1>
          </div>
          <div class="content">
            <div class="container">
              <?=$content?>
            </div>
          </div>
        </div>
      </section>
    </main>
    <?=(!empty(MODAL) && !empty($_SESSION['message']))?MODAL:"";?>
    <?=(isset($_POST['upload_file']) && !empty(FILE))? FILE:""?>
    <script src="<?=BASE_URI?>node_modules/chart.js/dist/chart.min.js"></script>
    <script src="<?=BASE_URI?>src/assets/Js/script.js"></script>
    <?=FOOTER?>
  </body>
</html>
```

Configuration d'apache 2

```
sudo apt-get -y install wget apache2*****installer apache2

sudo nano /etc/apache2/apache2.conf*****fichier de config d'apache

sudo systemctl restart apache2 *****redemarrer apache

sudo service apache2 restart *****redemarrer apache

++++++il faut donner les permissions
d'acc s au dossier  l'utilisateur d'apache+++++

sudo chown www-data /home/christian/app -Rf *****changer le propri taire d'un dossier
```

▼ autoload	●
🐘 Autoload.php	1
> css	
> images	
> js	
> node_modules	
▼ src	●
▼ app	●
🐘 Controller.php	5
🐘 Model.php	8
🐘 Router.php	1
> assets	
> config	●
▼ controllers	●
🐘 Admin.php	9+
🐘 Api.php	
🐘 Auth.php	
🐘 Pages.php	
🐘 Stock.php	
> lib	
▼ models	●
🐘 Article.php	
🐘 Categorie.php	
🐘 Formation.php	
🐘 Fournisseur.php	
🐘 Groupe.php	
🐘 Role.php	
🐘 Statistique.php	
🐘 Stocks.php	3
🐘 Utilisateur.php	
▼ views	●
> admin	
> layouts	●
> pages	
> stock	
> temp	
⚙ .htaccess	
📄 aati_bdd_v1 copy.sql	
🐘 index.php	

Arborescence du projet

### Annexe 9 – Certificat de réussite OpenClassroom -JS



### Annexe 10 – Certificat de réussite OpenClassroom PHP-SQL



### Annexe 11 – Certificat de réussite OpenClassroom CSS avec Sass



### Annexe 12 – Certificat OpenClassroom HTML5 – CSS3

