

**Labussière
Guillaume**

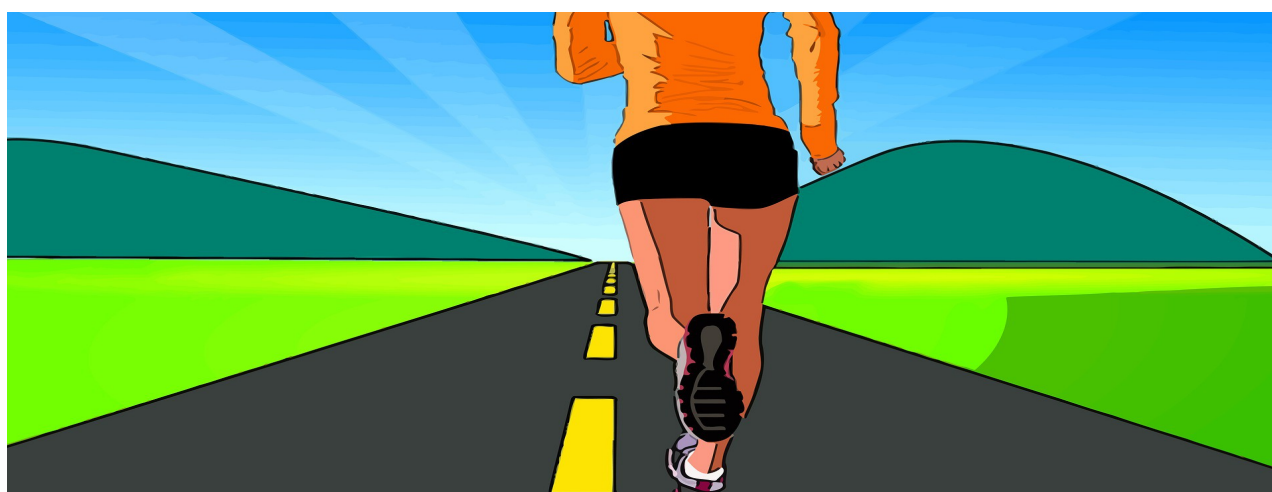


Dossier de Projet

Titre professionnel

**"Développeur web et web mobile" -
RNCP 31114**

SportVolt



sportvolt.lescadev.fr

Dates de présentation: Le 28/08/2020 à 8h30

Sommaire

Liste des compétences.....	p.2
Résumé du projet.....	p.3
Cahier des charges.....	p.4
INTRODUCTION.....	p.14
Front.....	p.15
I-Bootstrap.....	p.15
1.Qu'est-ce que Bootstrap ?	
2.Comment mettre en place du reponsive avec bootstrap ?	
II-SASS.....	p.16
1.Compléter les templates avec du SASS	
2.Qu'est-ce que SASS ?	
3.Comment mettre en place du SASS ?	
III-Balise html pour le référencement.....	p.17
1.Une balise title	
2.Une méta-description	
3.Les Balises Hn (pour n allant de 1 à 6)	
4.Les attribut Alt	
5.Sitemap	
IV-Une interface web dynamique.....	p.18
1.Définition	
2.JavaScript	
3.Exemple de code JavaScript	
Back	p.19
I-La Bdd.....	p.19
1.Qu'est-ce qu'une base de donnée?	
2.Qu'est-ce qu'une clés étrangères?	
3.Présentation de la base de donnée de Sportvolt	
II-Les SQL (Model).....	p.20/21
1.Qu'est-ce que le langage SQL?	
2.L'utilisation de Modèle dans un MVC	
3.Exemples de requête SQL	
III-Le PHP (Controllers).....	p.22/23
1.Qu'est-ce que le PHP?	
2.L'utilisation de Controllers dans un MVC	
3.Exemples de code PHP	
IV-Le Back office.....	p.24
1.Qu'est-ce qu'un back office?	
2.Comment le mettre en place	
Le principe de test unitaire.....	p.25
Sécurité.....	p.25
Documentation anglaise.....	p.26
Traduction.....	p.27
Conclusion.....	p.28
Annexes.....	p.29/30

Liste des compétences

FRONT-END	BACK-END
<p>Maquetter une application et créer un cahier des charges :</p> <ul style="list-style-type: none"> -La maquette prend en compte les spécificités fonctionnelles décrites dans les cas d'utilisation ou les scénarios utilisateur -La maquette est conforme à l'expérience utilisateur et à l'équipement ciblé -La maquette respecte les principes de sécurisation d'une interface utilisateur -Le contenu de la maquette, pour la partie visible, est rédigé, en français ou en anglais, de façon adaptée à l'interlocuteur et sans faute . 	<p>Créer une base de données :</p> <ul style="list-style-type: none"> -La base de données est conforme au schéma physique -Les scripts de création de bases de données et d'insertion des données de test s'exécutent sans erreurs -Les besoins de sécurité du SGBD sont exprimés selon l'état de l'art et les exigences de sécurité identifiées -La démarche de recherche permet de résoudre un problème technique ou de mettre en œuvre une nouvelle fonctionnalité
<p>Réaliser une interface utilisateur web statique et adaptable :</p> <ul style="list-style-type: none"> -L'interface est conforme à la maquette de l'application -Les bonnes pratiques de structuration sont respectées y compris pour le web mobile -Les pages web s'adaptent à la taille de l'écran -Le site respecte les règles de référencement naturel -La démarche de recherche permet de résoudre un problème technique ou de mettre en œuvre une nouvelle fonctionnalité 	<p>Développer les composants d'accès aux données :</p> <ul style="list-style-type: none"> -Les traitements relatifs aux manipulations des données répondent aux fonctionnalités décrites dans le dossier de conception technique -Les composants d'accès à la base de données suivent les règles de sécurisation reconnues -La démarche de recherche permet de résoudre un problème technique ou de mettre en œuvre une nouvelle fonctionnalité -La veille sur les vulnérabilités connues permet d'identifier des failles potentielles
<p>Développer une interface utilisateur web dynamique :</p> <ul style="list-style-type: none"> -Les pages web sont conformes à l'expérience utilisateur y compris pour l'expérience mobile -L'architecture de l'application répond aux bonnes pratiques de développement et de sécurisation d'application web -L'application web est optimisée pour les équipements mobiles -La démarche de recherche permet de résoudre un problème technique ou de mettre en œuvre une nouvelle fonctionnalité -La veille sur les vulnérabilités connues permet d'identifier des failles potentielles 	<p>Développer la partie back-end d'une application web ou web mobile :</p> <ul style="list-style-type: none"> -Les bonnes pratiques de développement sont respectées -Les composants serveur contribuent à la sécurité de l'application -Le code source des composants est documenté ou auto-documenté -La démarche de recherche permet de résoudre un problème technique ou de mettre en œuvre une nouvelle fonctionnalité -La veille sur les vulnérabilités connues permet d'identifier des failles potentielles
<p>Réaliser une interface utilisateur avec une solution de gestion de contenu :</p> <ul style="list-style-type: none"> -Les comptes utilisateurs sont créés avec leurs droits et rôles dans le respect des règles de sécurité -La structure du site est conforme au besoin client -Le site respecte les règles de référencement naturel -La démarche de recherche permet de résoudre un problème technique ou de mettre en œuvre une nouvelle fonctionnalité -La veille sur les vulnérabilités connues permet d'identifier des failles potentielles 	<p>Elaborer et mettre en œuvre des composants dans une application de gestion de contenu :</p> <ul style="list-style-type: none"> -Les composants complémentaires ou réalisés s'intègrent dans l'environnement de l'application -Les composants serveur contribuent à la sécurité de l'application -La démarche de recherche permet de résoudre un problème technique ou de mettre en œuvre une nouvelle fonctionnalité -La veille sur les vulnérabilités connues permet d'identifier des failles potentielles

Résumé du projet

Le présent projet, SportVolt, représente ma volonté de mettre en place un site permettant de raconter et d'illustrer les événements sportifs que l'on a vécu. En effet, étant passionné par le sport et ayant vécu des moments qui pour moi son mémorable , c'est avec ce thème que j'ai décidé d'exprimer mes compétences en développement web.

La première étape pour moi fut de rédiger un cahier des charges afin d'avoir une vision globale de ce que je voulais. Le cahier des charges a pour but de lister les objectifs du site . Dans un second temps, j'ai réalisé des wireframes (format téléphone , tablette et ordinateur) afin de visualiser la forme que possédera mon site sur toutes les plateformes.

Puis, il est venu le moment de créer la base de données, ce qui est le plus important pour créer un site internet. En effet, une base de donnée claire permet à elle seule de savoir qu'elle sera l'objectif du site.

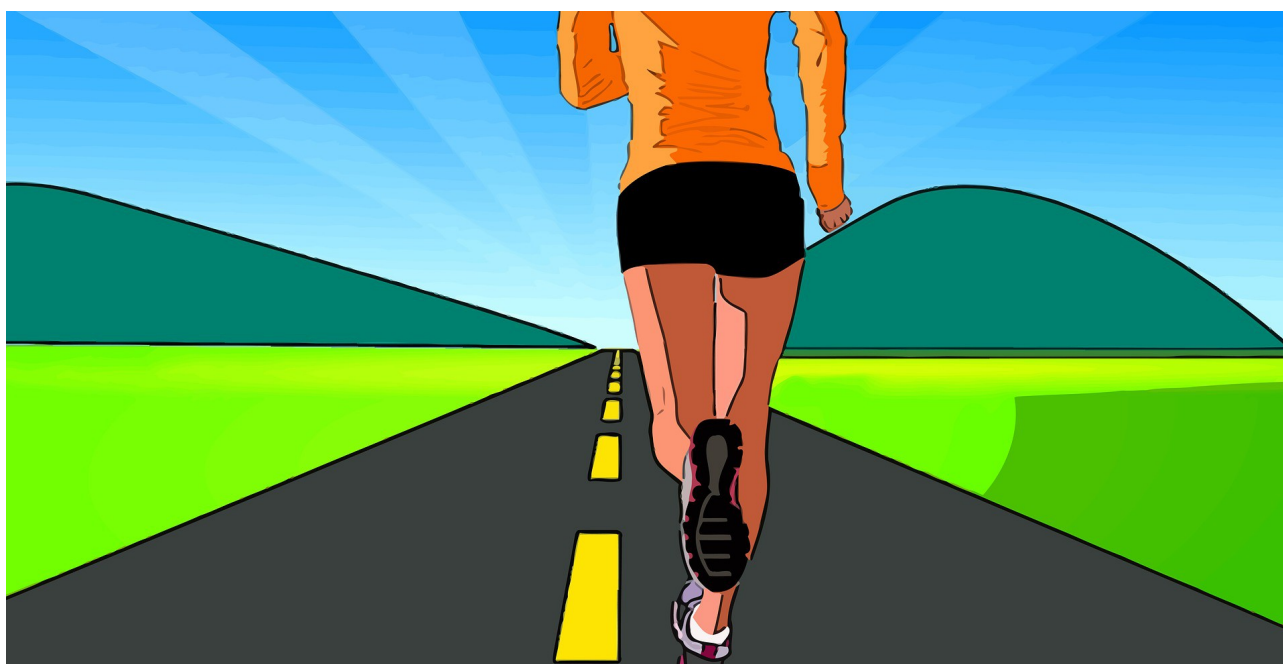
Suite à cela j'ai commencé par créer mon arborescence de fichiers pour mon MVC (model vue controller). Puis j'ai commencé par créer mon routeur ainsi que mon htaccess . Pour cette création, je l'ai effectuer dans un ordre logique. Dans un premier temps, j'ai fait les pages permettant d'alimenter la base de données (ex : ajout d'une histoire). Dans un second temps, j'ai construit les pages permettant d'afficher le contenu de la base de données(ex :la page qui affiche chaque histoire) puis pour finir j'ai codé les pages qui permettent de modifier le contenu de ma base de données (back office) . Pour la mise en ligne, j'ai créé un sous domaine (sportvolt.lescadev.fr) sur le serveur o2switch de la formation et j'ai utilisé filezilla pour transférer les fichiers.

**Labussière
Guillaume**



Cahier des charges

SportVolt



sportvolt.lescadev.fr

Sommaire

1. Présentation d'ensemble du projet

- a. Présentation de l'entreprise**
- b. Les objectifs du site**
- c. La cible adressée par le site**
- d. Objectifs quantitatifs**
- e. Périmètre du projet**

2. Description graphique et ergonomique

- a. Charte graphique**
- b. Design**
- c. Wireframe**

3. Description fonctionnelle et technique

- a. Arborescence du site**
- b. Description fonctionnelle**
- c. Informations relatives aux contenus**
- d. Contraintes techniques**

4. Prestations attendues et modalités de sélection des prestataires

- a. Prestations attendues**
- b. Planning**
- c. Méthodologie de suivi**

1. Présentation d'ensemble du projet

a. Présentation de l'entreprise :

Ici le Projet est réalisé selon mes besoins personnels. Le but n'étant pas de le proposer à une entreprise .

b. Les objectifs du site :

Le présent projet, SportVolt, représente ma volonté de mettre en place un site permettant de raconter et d'illustrer les événements sportifs que l'on a vécu. En effet, étant passionné par le sport et ayant vécu des moments qui pour moi son mémorable , je souhaite pouvoir mettre en place un moyen de raconter nos histoires ainsi que de découvrir celle des autres. Il y aura aussi un moyen de noter les histoires dès lors que l'on est connecté au site.

Ces histoires pourront être choisies par via une liste sur la page d'accueil ou via ou bouton qui sélectionnera une histoire aléatoire .

c. La cible adressée par le site :

Ce site vise les sportifs ainsi que les fans de sport afin de pouvoir partager des histoires de sport.

d. Objectifs quantitatifs :

Le site a pour but une dizaine de visites par mois, 5 nouveaux comptes , 3 nouvelles histoires et 2 notes par utilisateur connecté.

e. Périmètre du projet :

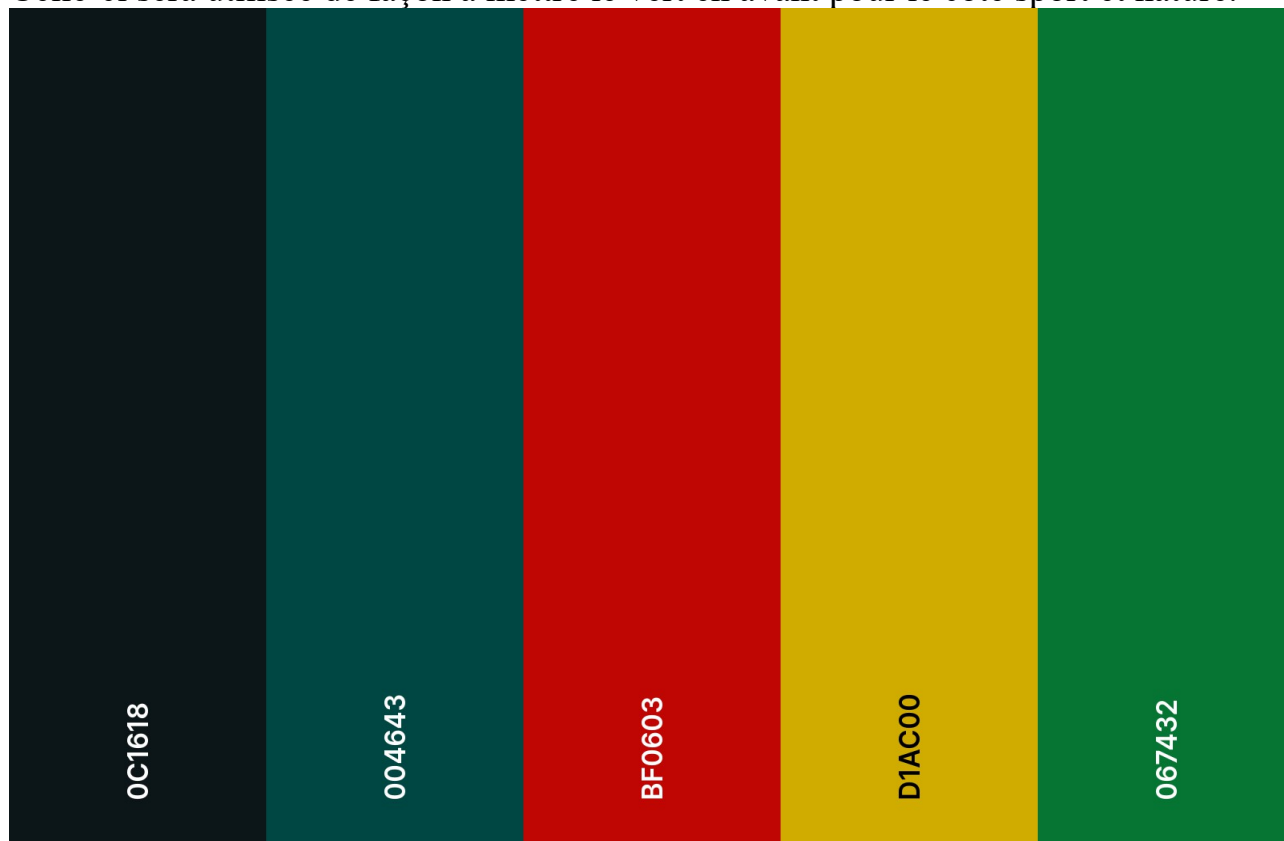
Le site sera uniquement en français dans un premier temps. Le site sera aussi responsive (format tablette et téléphone adapté).

2. Description graphique et ergonomique

a. Charte graphique :

Voici la palette de couleur qui sera utilisée sur le site :

Celle-ci sera utilisée de façon à mettre le vert en avant pour le côté sport et nature.



couleurprojetpro

coolors

Voici le favicon qui sera utilisé sur le site :



Voici l'image qui sera utilisée dans le header :



Voici les images qui seront utilisées dans le footer :



La police sera l'Arial.

b. Design :

Le design devra être très simple afin de le rendre accessible à tous types d'utilisateurs.

c. Wireframe :

Voici certains wireframes qui serviront pour le site :

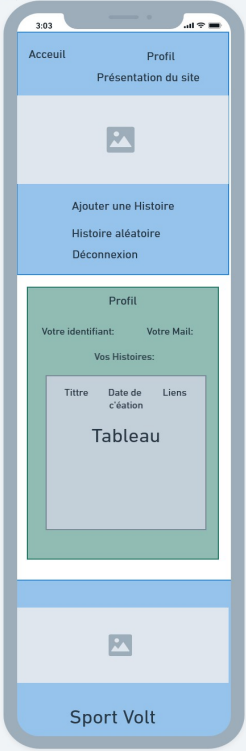
Tablet



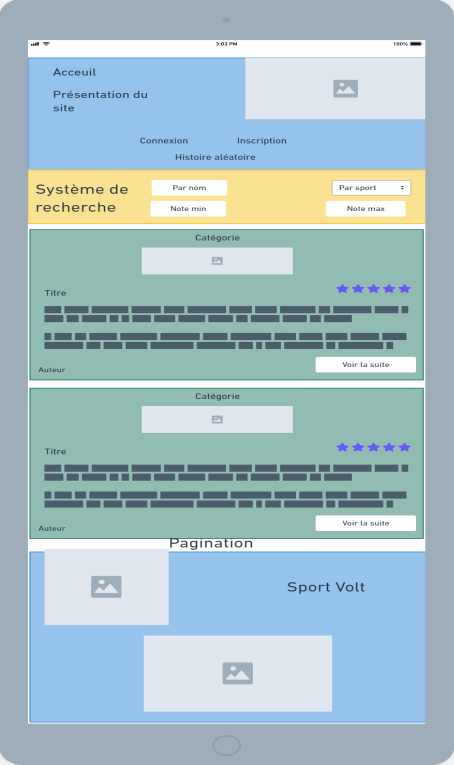
Profil



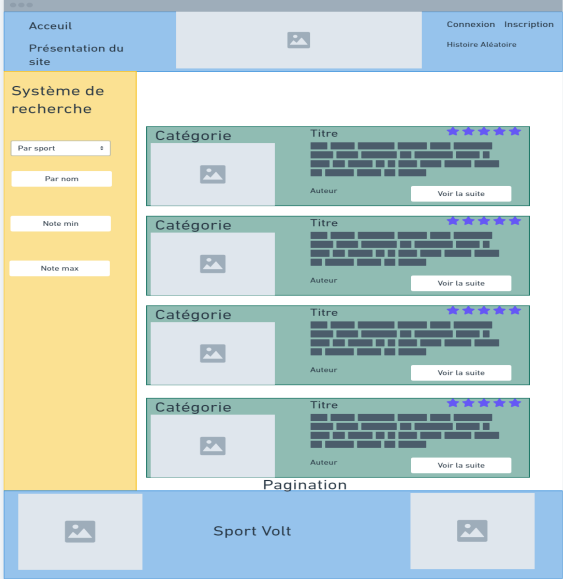
Tabletphone



Accueil tablette



Accueil PC

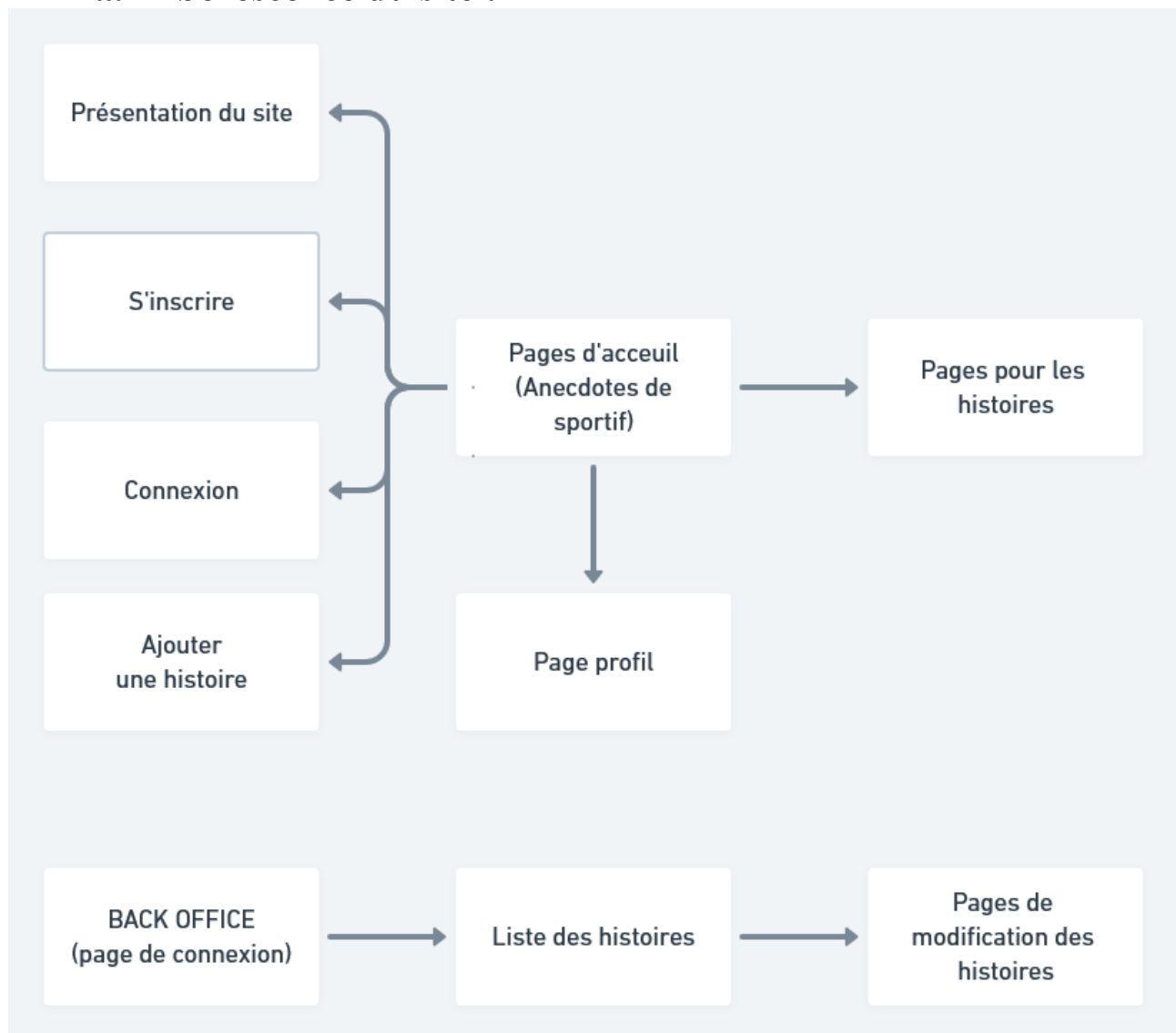


Accueil téléphone



3. Description fonctionnelle et technique

a. Arborescence du site :

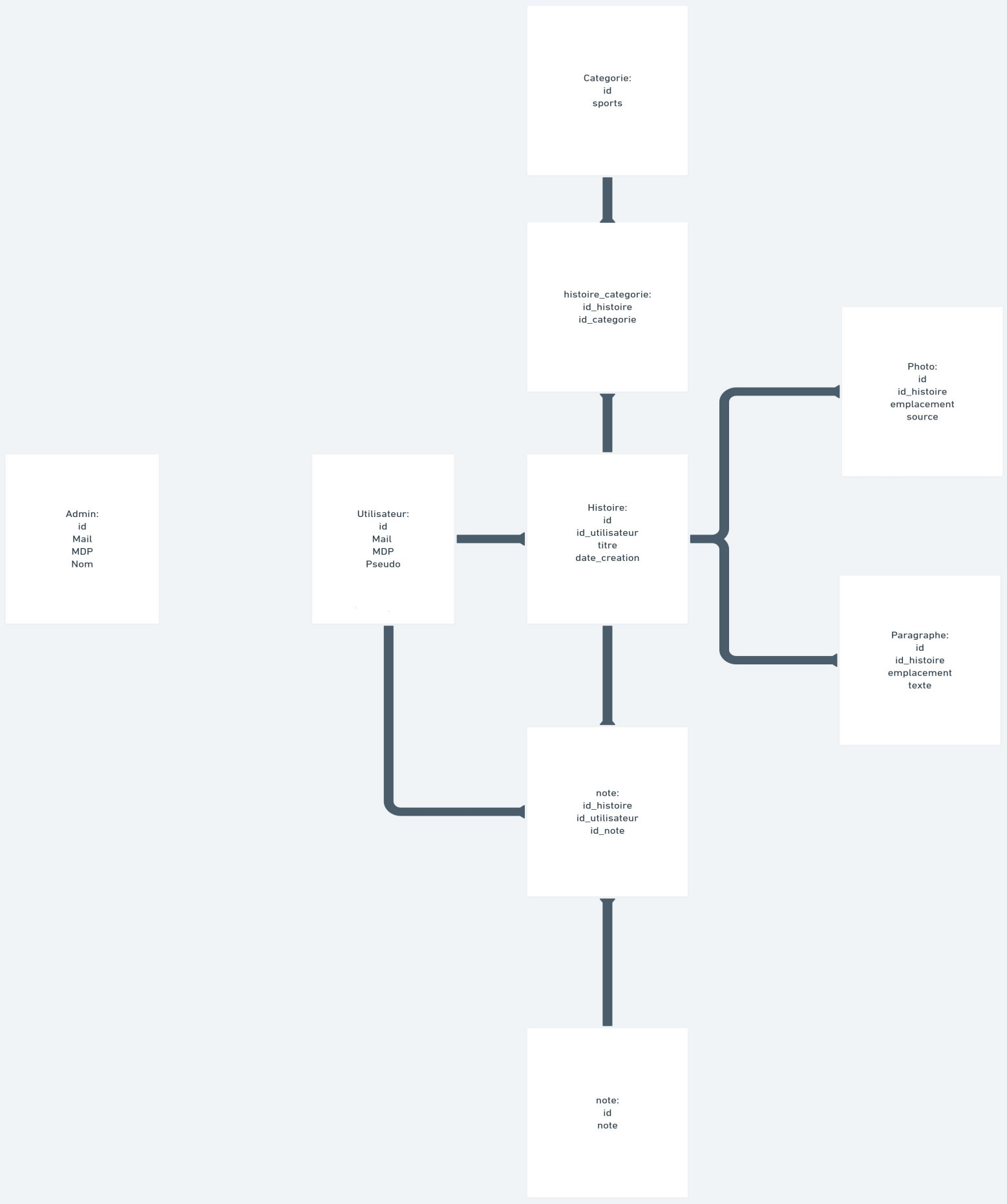


b. Description fonctionnelle :

Le site possédera un espace client sécurisé par un mot de passe. Il possédera aussi un back office pour modérer le contenu (seul endroit où le contenu sera modérable pour le MVP).

Un système de recherche devra être rajouté plus tard sur la page d'accueil.

c. Informations relatives aux contenus :



Le contenu principal sera les histoires créées par les utilisateurs . Le contenu sera uniquement modifiable par l'administrateur dans le back office.

d. Contraintes techniques :

Le site utilisera bootstrap pour la création du visuel.

Le site se devra d'être compatible avec la totalité des navigateurs.

L'hébergement se fera sur un sous domaine du serveur o2switch de la formation.

4. Prestations attendues et modalités de sélection des prestataires

a. Prestations attendues :

- développement
- design
- intégration
- migration de base de données
- maintenance et mises à jour

b. Planning :

Le MVP doit être rendu avant le 25/08/2020

c. Méthodologie de suivi :

Étant donné que je suis à la fois le client et le développeur, le suivi du site se fait au quotidien.

Introduction du projet:

Le site que je vais vous proposer est en MVC (Modèle-Vue-Contrôleur) en PHP, j'ai choisi ceci car le modèle MVC est un principe que l'on rencontre dans beaucoup de frameworks (symfony , angular ...). Nos deux besoins avant de pouvoir commencer à créer les pages du projet. Il faut créer un routeur et un htaccess afin de correctement effectuer la redirection des pages.

Suite à ce cahier des charges je vais vous présenter les outils pour la création de ce site en MVC avec dans un premier temps les outils pour concevoir un front puis dans un second temps on verra les outils pour développer le back. Puis suite à cela on verra le principe de test unitaire, et quelles sont les sites pour faire de veille sur la sécurité informatique. On terminera par une documentation anglaise sur les variables de session avant de conclure ce projet par ce qui m'a apporté dans mon expérience développeur.

Front :

I-Bootstrap

1.Qu'est-ce que Bootstrap ?

Bootstrap est une librairie CSS. En d'autres termes, Bootstrap est un ensemble de fichiers CSS et JavaScript fonctionnant ensemble et qu'on va pouvoir utiliser pour créer des designs complexes de manière relativement simple.

La librairie Bootstrap est donc un ensemble de fichiers CSS et JavaScript qui contiennent des règles prédéfinies et qui définissent des composants. De plus, Bootstrap utilise également des fichiers JavaScript. Bootstrap possède un avantage qui est pour moi notable .Un gain de temps de développement conséquent pour le responsive.

En plus de cela, utiliser Bootstrap assure d'avoir une certaine cohérence et une certaine robustesse dans l'ensemble du design.

2.Comment mettre en place du responsive avec bootstrap ?

a.Flexbox

Flexbox est une propriété qui va permettre d'agir sur les enfants d'un conteneur parents . En effectuant ceci il va être possible de ranger les conteneurs en ligne ou en colonne selon nos besoins. Tout conteneur qui se verra dépasser de la div , retournera à la ligne suivante.

b.Grid

Ce système est celui que j'ai le plus utiliser pour concevoir ce site web. Grid est un module CSS qui permet de créer des mises en page en divisant l'espace d'affichage en régions utilisables par une application ou en définissant des relations de taille et de position entre les différents éléments HTML.

Bootstrap optimise ceci avec un système à 12 colonnes(voir annexe 1). Pour le responsive , bootstrap permet de réutiliser les 12 colonnes selon le format demander . Ainsi pour une div , on peut lui accorder 12 colonnes en format téléphone ,4 en format tablette et 2 en format ordinateur . Pour utiliser cela il faut utiliser les classes suivantes :

	<576px	≥576px	≥768px	≥992px	≥1200px
Préfixe de classe	col-	col-sm-	col-md-	col-lg-	col-xl-

Si jamais on a besoin de décaler notre div de x colonnes, on utilisera la classe offset (par exemple « col-6 offset-3 »).Cela fonctionne aussi avec le responsive comme expliqué ci dessus (par exemple « col-sm-6 offset-sm-3 »).

II-SASS

1.Compléter les templates avec du SASS

Afin de faire correspondre notre site à nos wireframe on va compléter nos templates avec du SCSS

2.Qu'est-ce que SASS ?

SASS est un préprocesseurs CSS .C'est à l'aide d'un langage de feuille de style, qui est une sorte d'extension du CSS sous-jacent que l'on va pouvoir grandement simplifier le travail de conception du front.

3.Comment mettre en place du SASS ?

Un préprocesseur est un méta-langage qui nécessite d'être compilé en CSS "normal" pour être interprété par un navigateur. Il est donc nécessaire d'introduire une étape de compilation, où une moulinette va se charger de la transformation Sass en CSS (qui pour moi sera Live Sass Compiler , une extension de VS code).

SASS offre des options que le CSS seul ne propose pas comme par exemple, des variables , des boucles, des fonctions ou encore la notion d'héritage.

SASS permet aussi d'importer des fichiers. Ainsi on peut séparer le code en plusieurs dossiers puis tout importer dans un styles.scss.

Par exemple :

```
@import 'variable/variable';
@import 'page/header';
@import 'page/footer';
@import 'page/inscription';
@import 'page/ajouterunehistoire';
@import 'page/profil';
@import 'page/accueil';
@import 'page/presentation';
@import 'page/pagehistoire';
@import 'page/backoffice';
```

Ceci donnera un styles.min.css après la compilation .

```
assets > css > # styles.min.css > header
1 header{background-color: #004643;position:relative;display:block}header .liens{text-
```


III-Balise html pour le référencement

Afin de respecter le SEO (Search Engine Optimisation), ce site va posséder :

1.Une balise title

Il s'agit du titre donné à votre page web. Le contenu de la balise Title est l'élément repris par les moteurs de recherche lorsqu'ils affichent les résultats.Elle doit donc répondre à certains critères comme contenir le mot clé principal, ne pas dépasser 65 caractères (10 à 12 mots).

2.Une méta-description

Cette balise HTML a essentiellement pour objectif de faire cliquer les internautes.Le contenu de cette balise est le résumé qui s'affiche dans les résultats de recherche. Long d'une centaine de caractères, il permet de faire un rapide résumé du contenu et d'inciter à visiter le site.

3.Les Balises Hn (pour n allant de 1 à 6)

Les moteurs de recherches utilisent ces informations pour calculer le classement de votre page parmi tous les résultats.Il va être recommandé de mettre une seule balise H1 (qui pour moi est sportvolt dans le footer). Il ne faut pas oublier que ces balises ne servent uniquement au référencement et il ne faut pas les utiliser pour faire du visuel.

4.Les attribut Alt

Cette balise alt va donner un texte alternatif aux images. Ce qui en plus de favoriser l'accessibilité (pour le lecteur audio), elle va permettre d'optimiser le référencement en liant l'image à des mots clés .

5.Sitemap

Ce fichier Sitemap recense dans un format spécifique (XML) la liste de toutes les pages du site .

IV-Une interface web dynamique

1.Définition

Les pages dynamiques permettent de présenter les informations de différentes manières selon l'interaction avec le visiteur sans avoir besoin de recharger la page.

2.JavaScript

JavaScript est un langage de programmation, orienté objet, principalement connu comme le langage de script des pages web.

3.Exemple de code JavaScript

```
function modifParagraphe() {  
  
    if (document.getElementById('paragraphejs2').style.display == 'block') {  
        document.getElementById('paragraphejs2').style.display = 'none';  
        document.getElementById('btn2').style.display = 'none';  
        document.getElementById('btn3').style.display = 'none';  
        document.getElementById('btn1').style.display = 'block';  
        document.getElementById("paragraphe2").value = "";  
        document.getElementById("photop2").value = "";  
    }  
    else {  
        document.getElementById('paragraphejs2').style.display = 'block';  
        document.getElementById('btn2').style.display = 'block';  
        document.getElementById('btn3').style.display = 'block';  
        document.getElementById('btn1').style.display = 'none';  
    }  
};
```

Ce paragraphe aura pour but de décrypter le code ci dessus afin de faire fonctionner le bouton "Ajouter un paragraphe" et "Enlever un paragraphe" de la page ["http://sportvolt.lescadev.fr/histoire/ajouterhistoire"](http://sportvolt.lescadev.fr/histoire/ajouterhistoire).

Ici nous rencontrons la fonction modifParagraphe() , cette fonction ne prend aucun paramètre. Cette fonction a deux objectifs qui vont dépendre de leur réaction à la boucle if (le paragraphejs2 est actuellement en display block ou non), le premier est de le faire disparaître et de vider les 2 input qui sont dans "paragraphejs2 ". Le deuxième objectif est de , au contraire, faire apparaître "paragraphejs2 " si il n'était pas présent visuellement. Ainsi le visuel va changer sans avoir besoin de recharger la page.

Back :

I-La Bdd

1.Qu'est-ce qu'une base de données?

"Une base de données est un ensemble structuré et organisé de données qui représente un système d'informations sélectionnées de telle sorte qu'elles puissent être consultées par des utilisateurs ou par des programmes."

Définition Larousse

2.Qu'est-ce qu'une clés étrangères?

La clé étrangère représente un ou plusieurs champs qui pointe vers la clé primaire d'une autre table. L'objectif de la clé étrangère est d'assurer l'intégrité des données. La clé va aussi nous permettre de faire du one to many et du many to many.

3.Présentation de la base de donnée de Sportvolt

Pour comprendre ce qui va suivre je vous invite à regarder l'annexe2.

Partons de la table utilisateur .Elle va contenir plusieurs valeur : id qui sera une clés primaire en auto increment , l'email qui sera unique pour chaque utilisateur , le pseudo et le mdp qui devra être hashé. La 1ere clé étrangère (qui va de utilisateur à histoire) nous permet de faire du one to many , car un utilisateur peut posséder plusieurs histoires (en stockant id_utilisateur dans la table histoire).

Maintenant regardons utilisateur_histoire_note. Ceci est une table de clés étrangères, elle nous permet de stocker plusieurs informations. Ici cette table sert à faire fonctionner le système de note, on stock l'histoire qui vient d'être notée, l'utilisateur qui vient de notée (afin qu'il ne puisse pas revoter) et la note qui a été choisi par l'utilisateur.

Puis pour présenter une troisième table , regardons histoire_categorie cette table nous permet de faire du one to many. Car pour chaque catégorie choisie on va la noter avec l'id_histoire.

II-Les SQL (Model)

1.Qu'est-ce que le langage SQL?

Le langage SQL (Structured Query Language) est un langage informatique utilisé pour exploiter des bases de données.

2.L'utilisation de Modèle dans un MVC

Les modèles sont des classes qui serviront uniquement à faire des requêtes SQL grâce aux méthodes. Ces classes, pour être utilisées, devront être instanciées. Puis on pourra appeler les méthodes comme dans l'exemple ci-dessous.

```
$Categorie = new Categorie;  
  
$Categorie->ajoutCategorie($connection, $histexist['id'], $categorieid['id']);
```

Ici on instancie la classe Categorie, ainsi \$Categorie va posséder toutes les méthodes de la classe demandée. Puis on va appeler la méthode ajoutCategorie qui a besoin de 3 paramètres pour fonctionner (ici \$connection, \$histexist['id'], \$categorieid['id'])

3.Exemples de requête SQL

Je vais vous montrer certaines requêtes SQL qui ont été effectuées pour ce projet puis on en expliquera le sens.

Exemple n°1

```
$req = $connection->prepare("INSERT INTO histoire (id_utilisateur, titre,  
date_creation) VALUES(:id_utilisateur, :titre, :date_creation)");  
$req->execute(array(  
    'id_utilisateur' => $id_utilisateur,  
    'titre' => $titre,  
    'date_creation' => $date  
));
```

Dans cet extrait, on va appliquer la requête INSERT INTO qui va permettre d'insérer des valeurs dans la bdd. Ici on prépare une insertion dans la table histoire. Trois colonnes sont visées : id_utilisateur, titre, date_creation. Puis dans le execute(), on va envoyer le tableau avec les valeurs que l'on souhaite intégrer dans la bdd. Ici \$id_utilisateur ira dans la colonne id_utilisateur, \$titre ira dans la colonne titre and \$date ira dans la colonne date_creation.

Exemple n°2

```
$reqcategorie = $connection->prepare("SELECT * FROM categorie WHERE sport = ? ");  
$reqcategorie->execute(array($categorie));
```

Dans cet extrait, on va appliquer la requête SELECT qui va permettre de sélectionner des valeurs dans une colonne. Ici SELECT * demande à sélectionner la totalité des colonnes de la table categorie (FROM categorie). La clause WHERE va demander les lignes où la colonnes sport est égale à la valeur donné (ici \$categorie).

Exemple n°3

```
$req = $connection->prepare("DELETE FROM histoire WHERE id= $histoireId ");  
$req->execute();
```

Dans cet extrait, on va appliquer la requête DELETE qui va permettre de supprimer une ou plusieurs lignes de la table donnée . Ici on supprime la ligne de la table histoire où id égale à \$histoireId.

Exemple n°4

```
$req = $connection->prepare("UPDATE histoire SET titre = :titre  
WHERE id= $histoireId ");  
$req->execute(array(  
    'titre' => $titre  
));
```

Dans cet extrait, on va appliquer la requête UPDATE qui va permettre de mettre à jour les valeurs dans la base de données. Ici on va préparer la mise à jour de la colonne titre de la table histoire où id est égale à \$histoireId. Puis comme dans l'exemple1, on va "exécute" avec un tableau où titre va prendre la valeur \$titre .

III-Le PHP (Controllers)

1.Qu'est-ce que le PHP?

Ce langage est principalement utilisé pour produire des sites webs dynamiques. Exécuté du côté serveur (l'endroit où est hébergé le site) il n'y a pas besoin aux visiteurs d'avoir des logiciels ou plugins particulier. Néanmoins, les développeurs qui souhaitent créer un site en PHP doivent s'assurer que l'hébergeur prend en compte ce langage et aussi vérifier la version dont ils auront besoin pour effectuer une mise en ligne d'un site fait avec un framework.

2.L'utilisation de Controllers dans un MVC

Dans un MVC , les controllers ont un rôle primordial qui est de faire le lien entre les views et les modèles. Pour pouvoir appeler un controllers il faudra rentrer dans l'url le nom de celui-ci puis le nom de l'action demandée (par exemple : <http://sportvolt.lescadev.fr/user/connection> , ici on appelle UserController et sa fonction connectionAction). C'est dans ces actions où l'on va généralement required la view (template html).

3.Exemples de code PHP

Je vais vous montrer certains codes PHP qui ont été effectués pour ce projet puis on en expliquera le sens.

Exemple n°1 de connectionAction dans le BackofficeController

```
function connectionAction()
{
    session_start();
    if (isset($_POST['formconnection'])) {

        $email = htmlspecialchars($_POST['email']);
        $pass = htmlspecialchars($_POST['Password']);

        if (!empty($_POST['email']) and !empty($_POST['Password'])) {
            $bdd = new Bdd();
            $connection = $bdd->getConnection();
            $admin = new Admin();
            $reqmail = $admin->verifMail($connection, $email);
            $userexist = $reqmail->fetch();
            if ($userexist && password_verify($pass, $userexist['mdp'])) {

                $_SESSION['adminid'] = $userexist['id'];
                header('Location:' . BASE_URL . 'backoffice/listehistoire');
            } else {
                $erreur = 'Mauvais mots de passe';
            }
        }
    }
}
```

```

    }
    } else {
        $erreur = 'veuillez remplir tous les champs';
    }
}
require('views/backoffice/connection.php');
}

```

Nous allons dans un premier temps analyser la première et dernière ligne. La première "session_start" qui permet d'activer les variable de session et la dernière ligne permet d'aller chercher la view "views/backoffice/connection.php". Maintenant regardons le boucle de condition.

Ici nous allons créer une condition qui réagira quand on cliquera sur le bouton ayant pour name "formconnection". Suite à cela nous allons vérifier si les input email et password sont remplis. Si ces deux condition sont remplis alors on peut rentrer dans dans cette deuxième boucle if.

Après cela on va utiliser la commande htmlspecialchars qui va convertir les caractères spéciaux en entités HTML . Puis nous allons instancier la classe Bdd pour réaliser la méthode getConnection() que nous enregistrons dans \$connection. Nous allons pouvoir instancier la classe Admin afin de pouvoir rechercher les admins dans la bdd grâce à verifMail(). Suite à cela on va décrypter le mot de passe dans la bdd puis l'on va le comparer à celui que l'utilisateur a rentré. Si la condition est remplie on va créer des variable de session qui serviront à reconnaître l'administrateur avant de la rediriger vers la page backoffice/listehistoire.

Exemple n°2 de histoiremodifAction dans le BackofficeController

```

if (isset($_POST['modifhistoire'])) {
    if (isset($_POST['chek'])) {
        $titreupdate = htmlspecialchars($_POST["titre"]);

        $histoire->updateTitre($connection, $histoireId, $titreupdate);
        for ($i = 1; $i < 4; $i++) {
            if (isset($_POST["paragraphe" . $i])) {
                $text = htmlspecialchars($_POST["paragraphe" . $i]);
                $paragraphe->updateParagraphe($connection, $histoireId, $text, $i);
            }
        }
        header('Location:' . BASE_URL . 'backoffice/histoiremodif/' . $histoireId);
    } else {
        $erreur = "La case pour valider n'est pas coché.";
    }
}
}

```

Dans cet exemple nous allons regarder la boucle for. Ici la boucle for va set une valeur \$i qui partira de 1 et ira jusqu'à 3, ainsi à chaque réitération on pourra utiliser \$i . Suite à cela on va créer un condition qui va vérifier si un input paragraphe est rempli. Pour cela on va concaténer "paragraphe" et \$i pour vérifier le contenu de l'input paragraphe 1, 2 et 3. Si le contenu existe alors on utilisera updateParagraphe() pour mettre à jour le paragraphe de l'histoire demandé.

IV-Le Back office

1.Qu'est-ce qu'un back office?

Un back office permet l'ensemble des procédures de gestions administratives qui n'est pas disponible à un utilisateur classique.

2.Comment le mettre en place

Pour un question de sécurité, nous devons créer une adresse url qui ne soit pas facile à trouver et mettre la balise : `<meta name="robots" content="noindex, nofollow">` afin que le site ne puisse pas être trouvable via une recherche (Pour être plus précis il n'y aura ni indexation de la page ni prise en compte des liens).

Dans un second temps un système de connexion est obligatoire pour vérifier l'utilisateur qui essaie de se connecter. Pour l'inscription de l'administrateur, j'ai décidé de mettre un formulaire d'inscription en ligne uniquement le temps de créer mon compte administrateur afin qu'un utilisateur random ne puisse pas créer un compte admin et venir modifier les informations du site en ligne. Pour information le mot de passe de l'administrateur est crypté comme les mots de passe de l'utilisateur classique.

Une fois ceci effectué, l'administrateur pour accéder à la liste des histoires et il aura le choix de les modifier ou de les supprimer.

Le principe de test unitaire :

Un test unitaire est un procédé permettant de s'assurer du bon fonctionnement d'une unité de programme. Si on prend l'exemple de mon projet, on pourrait créer un test unitaire pour vérifier le système de calcul de moyenne pour les notes. On pourrait créer un page qui n'aurait pour but unique d'afficher le résultat de cette fonction et si le résultat est valide.

Par exemple pour le test cité ci dessus, on va posséder la fonction calculMoyenne() qui prendra en paramètre un tableau de valeurs. Le test renverra une erreur si la moyenne est négative ou si elle est supérieure à la note maximum (5 pour mon site). Dans le cas contraire le test unitaire sera validé. Cela permet de séparer les fonctions si on observe une erreur et de les réparer après cette observation.

Sécurité :

Afin d'effectuer des veilles sur la sécurité je me rend sur ces sites :

www.cert.ssi.gouv.fr

www.exploit-db.com/

Ce sont des sites proposant plusieurs articles sur exploitation des vulnérabilités des sites web ainsi que leurs bases de données.

Documentation anglaise (https://www.w3schools.com/php/php_sessions.asp):

J'ai choisi la documentation de W3schools pour détruire les variables de session (que j'utilise dans le controller pour la déconnexion de l'utilisateur).

A session is a way to store information (in variables) to be used across multiple pages.

Unlike a cookie, the information is not stored on the users computer.

What is a PHP Session?

When you work with an application, you open it, do some changes, and then you close it. This is much like a Session. The computer knows who you are. It knows when you start the application and when you end. But on the internet there is one problem: the web server does not know who you are or what you do, because the HTTP address doesn't maintain state.

Session variables solve this problem by storing user information to be used across multiple pages (e.g. username, favorite color, etc). By default, session variables last until the user closes the browser.

So; Session variables hold information about one single user, and are available to all pages in one application.

Destroy a PHP Session

To remove all global session variables and destroy the session, use `session_unset()` and `session_destroy()`:

Example

```
<?php
session_start();
?>
<!DOCTYPE html>
<html>
<body>

<?php
// remove all session variables
session_unset();

// destroy the session
session_destroy();
?>

</body>
</html>
```

Traduction :

Une session est un moyen de stocker des informations (dans des variables) à utiliser sur plusieurs pages.

Contrairement à un cookie, les informations ne sont pas stockées sur l'ordinateur des utilisateurs.

Qu'est-ce qu'une session PHP?

Lorsque vous travaillez avec une application, vous l'ouvrez, apportez des modifications, puis vous la fermez. Cela ressemble beaucoup à une session. L'ordinateur sait qui vous êtes. Il sait quand vous démarrez l'application et quand vous terminez. Mais sur Internet, il y a un problème: le serveur Web ne sait pas qui vous êtes ni ce que vous faites, car l'adresse HTTP ne conserve pas son état.

Lorsque vous travaillez avec une application, vous l'ouvrez, apportez des modifications, puis vous la fermez. Cela ressemble beaucoup à une session. L'ordinateur sait qui vous êtes. Il sait quand vous démarrez l'application et quand vous terminez. Mais sur Internet, il y a un problème: le serveur Web ne sait pas qui vous êtes ni ce que vous faites, car l'adresse HTTP ne conserve pas son état.

Alors; Les variables de session contiennent des informations sur un seul utilisateur et sont disponibles pour toutes les pages dans une seule application.

Détruire une session PHP

Pour supprimer toutes les variables de session globales et détruire la session, utilisez `session_unset()` et `session_destroy()`:

Exemple

```
<?php
session_start();
?>
<!DOCTYPE html>
<html>
<body>

<?php
// supprimer les variables de session
session_unset();

// détruire la session
session_destroy();
?>

</body>
</html>
```

Conclusion:

Ce projet m'aura permis de consolider mes connaissances sur le PHP et sur la logique de mises en place d'un MVC , JS, ce qui a eu pour résultat de me préparer à tous les frameworks que je pourrai croiser dans ma carrière (comme par exemple: symfony qui sera le framework utilisé pour alternance si je réussis mon examen).

Il m'aura aussi permis de me rendre compte que je préfère le back-end, car j'adore créer des bases de données et utiliser ces données pour rendre un résultat demandé comme par exemple mon système de note.

Je vais conclure par des remerciements. Je remercie Elodie pour avoir été à notre écoute et d'avoir réglé nos problèmes administratifs quand on en avait besoin. Je remercie Sébastien Terre de m'avoir accepté en stage à Letseat et de m'avoir proposé une alternance pour 3il. Je remercie Graham Gilbert de nous avoir appris les méthodes agiles ainsi que ces connaissances en bases de données. Je remercie Nicolas Vauché pour son partage de sa passion pour l'informatique et de son soutien dès que nous avions besoin d'aide. Et pour finir je remercie mes camarades de formation avec qui on s'estentraîné et avec qui ces 6 mois sont passés très vite.

Annexes :
Annexe 1 :

4 columns (12/4 = 3)

col-3	col-3	col-3	col-3
-------	-------	-------	-------

3 columns (12/3 = 4)

col-4	col-4	col-4
-------	-------	-------

2 columns (12/2 = 6)

col-6	col-6
-------	-------

1 column (12/1 = 12)

col-12

Annexe 2 :

