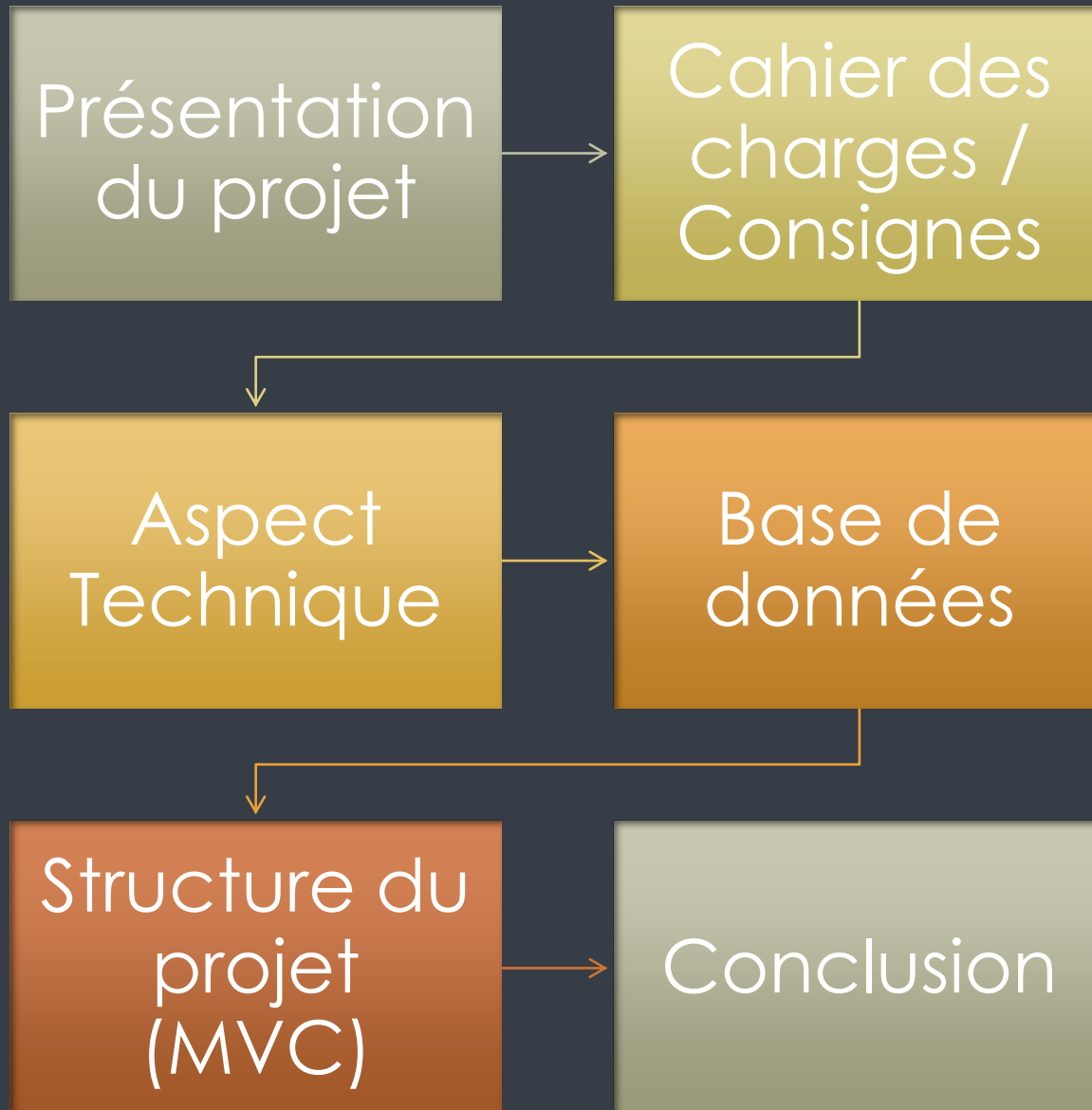


PPE GESTION DES CLIENTS

RÉALISER UNE INTERFACE ADMIN GESTION CLIENTS

Diariyata TOURE SIO2 slam








SOMMAIRE

PRESENTATION

- CETTE PARTIE DE SITE ADMIN A POUR BUT DE PERMETTRE AUX ADMINISTRATEURS DES MAISONS DES LIGUES, DE RECENSER LEURS CLIENTS. POUR CE FAIRE, IL FAUT CRÉER UNE INTERFACE ADMIN QUI PERMET D'AJOUTER UN CLIENT, MODIFIER SONT PROFILE OU LE SUPPRIMER

⚙ Maison des ligues : Interface admin

Identifiant	Nom	Prenom	Age	Adresse Mail
 Modifier ou  supprimer	Eich	Branden	55	brand67@gmail.com
 Modifier ou  supprimer	Ledorf	Rasmus	55	rasmus@php.net
 Modifier ou  supprimer	Berners Lee	Tim	70	tim@w3c.net

+ Ajouter un client

CAHIER DES CHARGES

Aspect fonctionnel de la partie gestion :

1. Agencement du tableau des enregistrements d'informations des clients
2. Création d'un formulaire pour les enregistrement à publier, avec 4 champs obligatoire (nom, prénom, âge et adresse mail).
3. Affichage des warning dans le cas où l'administrateur ne renseigne pas un champ.

ASPECT TECHIQUE

Les technologies utilisées sont : html5, css3, php orienté objet & SQL

Police de caractères : 'Open Sans', sans-serif

Iconographies obligatoires pour la couleur des boutons : #09599A(bleu), #00614E(vert) et #d63031(rouge). La couleur des thèmes doivent être indexée dans des variables CSS

Utilisez Visual Studio Code et phpMyAdmin pour la bdd.

BASE DE DONNEES

- LA BASE DE DONNÉES, CRÉER VIA PHPMYADMIN, NOUS PERMET D'AVOIR UNE CONNEXION DIRECT AVEC L'APPLICATION AFIN DE POUVOIR GARDER EN MÉMOIRE LES INFORMATIONS DES CLIENTS.

```
--          CREATION DE LA BASE

--

CREATE DATABASE
client_ligue;

--          TABLE : clients

--

CREATE TABLE clients
(
  id_client INT(11) AUTO_INCREMENT,
  nom VARCHAR(20) NOT NULL ,
  prenom VARCHAR(50) NOT NULL ,
  age INT(2) NOT NULL ,
  email VARCHAR(150) NOT NULL ,
  PRIMARY KEY (id_client)
);
```

Structure de table		Vue relationnelle								
#	Nom	Type	Interclassement	Attributs	Null	Valeur par défaut	Commentaires	Extra	Action	
<input type="checkbox"/>	1 id_client	int(11)			Non	Aucun(e)		AUTO_INCREMENT	Modifier	Supprimer Plus
<input type="checkbox"/>	2 nom	varchar(20)	utf8mb4_general_ci		Non	Aucun(e)			Modifier	Supprimer Plus
<input type="checkbox"/>	3 prenom	varchar(50)	utf8mb4_general_ci		Non	Aucun(e)			Modifier	Supprimer Plus
<input type="checkbox"/>	4 age	int(2)			Non	Aucun(e)			Modifier	Supprimer Plus
<input type="checkbox"/>	5 email	varchar(150)	utf8mb4_general_ci		Non	Aucun(e)			Modifier	Supprimer Plus

MODELE VUE CONTROLEUR

POUR LA STRUCTURE DU CODE, PHP ORIENTÉ OBJET, DE CE PROJET, NOUS AVONS UTILISÉ LA MÉTHODE MVC. CETTE ARCHITECTURE SE DÉCOMPOSE EN 3 COUCHES QUI NOUS PERMETTENT D'INTERVENIR SUR LE CODE DE MANIÈRE ORGANISER :

- LES MODÈLES CONTIENNENT TOUT LE BACK-END. C'EST-À-DIRE LES CONNEXIONS À LA BASE DE DONNÉES & LES FONCTIONS DE L'APPLICATION (ICI AJOUT, MODIFICATION ET SUPPRESSION DES CLIENTS).
- LES VUES GÈRENT LE FRONT-END, C'EST-À-DIRE L'INTERFACE POUR LES UTILISATEURS. DANS LES CAS PRÉSENT, LES VUES AFFICHENT LE TABLEAU D'INFORMATIONS CLIENT & LES FORMULAIRES D'AJOUT ET DE MODIFICATIONS.
- LES CONTRÔLEURS ONT POUR RÔLE D'ÉTABLIR LA COMMUNICATION ENTRE LE BACK-END ET LE FRONT-END. ILS PERMETTENT LE BON FONCTIONNEMENT DE L'APPLICATION.

LES MODELES

Connexion à la BDD :

```
<?php

function connectToDatabase()
{
    $dsn = 'mysql:dbname=client_ligue;host=localhost';
    $user = 'root';
    $password = '';
    $options = array(
        PDO::ATTR_ERRMODE => PDO::ERRMODE_WARNING,
        PDO::MYSQL_ATTR_INIT_COMMAND => "SET NAMES UTF8"
    );

    $pdo = new PDO($dsn, $user, $password, $options);
    return $pdo;
}
```

Ajout des clients :

```
function ajoutClient($nom, $prenom, $age, $email)
{
    // connexion a la bdd
    $pdo = connectToDatabase();

    // preparation de la requete :
    $query = $pdo->prepare("
    INSERT INTO clients
    (nom, prenom, age, email)
    VALUES (?, ?, ?, ?)
    ");

    // on execute
    $query->execute(
        [
            $nom,
            $prenom,
            $age,
            $email,
        ]
    );
}
```

Modification des infos clients :

```
// modifier.inc.php : modification d'un client
function ModifierClient($id_client, $nom, $prenom, $age, $email){
    // connexion a la bdd
    $pdo = connectToDatabase();







    // preparation de la requete :
    $query = $pdo->prepare("
    UPDATE clients SET
        nom = ?,
        prenom = ?,
        age = ?,
        email = ?
    WHERE id_client = ?
    ");

    // on execute
    $query->execute(
        [
            $nom,
            $prenom,
            $age,
            $email,
            $id_client,
        ]
    );
}
```


LES VUES

Liste des clients :

⚙ Maison des ligues : Interface admin

Identifiant	Nom	Prenom	Age	Adresse Mail
 Modifier ou  supprimer	Eich	Branden	55	brand67@gmail.com
 Modifier ou  supprimer	Ledorf	Rasmus	55	rasmus@php.net
 Modifier ou  supprimer	Berners Lee	Tim	70	tim@w3c.net

+ Ajouter un client

Ajout des clients :


[VOIR LA LISTE DES UTILISATEURS >](#)

Nom :

Prenom

Age:

Adresse mail:

Créer 

Modification et suppression :

[VOIR LA LISTE DES UTILISATEURS >](#)

Nom :

Eich

Prenom


Branden


Age:

55

Adresse mail:

brand67@gmail.com

Mettre a jour 

Supprimer 

LES CONTRÔLEURS

Ajout des clients :

```
if (
    isset($_POST['nom'])
    && isset($_POST['prenom'])
    && isset($_POST['age'])
    && isset($_POST['email'])
) {

    $msg_ok = '<div>Votre enregistrement à bien été pris en compte.</div>';

    ajoutClient($_POST['nom'], $_POST['prenom'], $_POST['age'], $_POST['email']);

    echo $msg_ok;

}

include '../vues/ajout.php';
```

Modification des informations clients :

```
$infosClient = infoClient($_GET['id_client']);

if (!$infosClient) { // si $infosClient == false : l'id ne correspond à aucun client, on redirige vers l'
    header('location:../index.php');
    exit();
}

if (
    isset($_POST['id_client'])
    && isset($_POST['nom'])
    && isset($_POST['prenom'])
    && isset($_POST['age'])
    && isset($_POST['email'])
) {

    ModifierClient($_POST['id_client'], $_POST['nom'], $_POST['prenom'], $_POST['age'], $_POST['email']);

    $_SESSION['message'] = '<div>Vos informations ont bien été mise à jour.</div>';
    header('location:../index.php');

}
```


CONCLUSION

- POUR LA CRÉATION DE CETTE PARTIE DE SITE ADMIN, L'OBJECTIF ÉTAIT DE METTRE EN PLACE LA GESTION DES INFORMATIONS UTILISATEURS, AFIN DE POUVOIR STOCKER LEURS INFORMATIONS, LEURS PERMETTRE LA MODIFICATION & LA SUPPRESSION DE CELLES-CI.
- LA BONNE RÉALISATION DE CE PROJET A PU ÊTRE MENÉ GRÂCE AU BON AGENCEMENT DE LA BASE DE DONNÉES, ET À L'ARCHITECTURE EN MVC DU CODE SOURCE.