

# Rapport de projet

---

Cahier des charges :.....	2
Schéma enchaînement des écrans :.....	3-4
Structure applicative :.....	5
Charte graphique :.....	6
Cahier de suivi Agil:.....	7
Charte Down :.....	8
Tableau de progression :.....	8
Mentions Légales :.....	9
Structure BD :.....	11-22
code site applicative :.....	23-68

# Cahier des charges :

## Règles de programmation imposées :

Application réalisée en php avec structure MVC.

La couche modèle utilisera une API REST pour l'accès à la BD.

Le framework bootstrap sera utilisé pour gérer la structure des pages et rendre le site responsive.

Le projet sera réalisé en binôme en appliquant la méthode agile et en utilisant github ou gitlab.

## Fonctionnalités attendues :

- Présentation des médicaments fabriqués par GSB avec pour chacun possibilité d'afficher les effets thérapeutiques et les effets secondaires ainsi que les interactions avec d'autres médicaments.
- Présentation des activités complémentaires proposées par GSB avec possibilité de s'inscrire à l'une des activités.

- Le site comportera les mentions juridiques nécessaires, le droit des images utilisées sera vérifié

### Documents à fournir :

- Schéma d'enchaînement des écrans,
- Structure BD mise à jour,
- Structure applicative,
- Script de sauvegarde de la BD
- Charte graphique
- Cahier de suivi Agile
- Etude juridique : mentions obligatoires à mentionner sur le site


## Calendrier (2 semaines= 4 séances) :

L'application devra être réalisée en 2 semaines. Voici une proposition de planning :


- séance 1 : détermination du backlog et des sprints (prévoir un sprint init projet et un sprint finish projet...)
- séances suivantes : la séance débute par une mêlée, on tient à jour le cahier de suivi Agile

## Schéma enchaînement des écrans :

### Page liste des médicaments :

 Galaxy Swiss Bourdin <a href="#">Conférence</a> <a href="#">Mentions légales</a> <span>Se dé</span>		
Liste des médicaments de GSB		
ID	Nom	Actions
1	Paracétamol	<a href="#">Détails</a>
2	Ibuprofène	<a href="#">Détails</a>
3	Tramadol	<a href="#">Détails</a>
4	Omeprazole	<a href="#">Détails</a>
5	Amoxicilline	<a href="#">Détails</a>
6	Diazépam	<a href="#">Détails</a>
7	Lévothyroxine	<a href="#">Détails</a>
8	Aspirine	<a href="#">Détails</a>
9	Acétylsalicylate	<a href="#">Détails</a>

### Page détail des médicaments :

 Galaxy Swiss Bourdin <a href="#">Conférence</a> <a href="#">Mentions légales</a> <span>Se déconnecter</span>	
<b>Description</b>	
Le paracétamol, aussi appelé acétaminophène, est un composé chimique utilisé comme antalgique et antipyrétique, qui figure parmi les médicaments les plus communs, utilisés et prescrits au monde.	
Soulagement de la douleur et réduction de la fièvre.	
Risque de lésions hépatiques en cas de surdosage.	
Ibuprofène	

### Page activités :

Liste des Conférence				
ID	Nom	Description	Date	
1	conference1.0	reunion mensuelle	2024-01-05	<input type="text" value="Choisir un utilisateur"/> <a href="#">Rejoindre</a>
2	conference2.0	reunion mensuelle	2024-02-02	<input type="text" value="Choisir un utilisateur"/> <a href="#">Rejoindre</a>
3	conference3.0	reunion mensuelle	2024-03-01	<input type="text" value="Choisir un utilisateur"/> <a href="#">Rejoindre</a>

### Page juridique :

# Mentions Légales

## 1. Informations légales

Raison sociale : Galaxy Swiss Bourdin

Forme juridique : Société Anonyme

Capital social : 60 000 000 €

Siège social : Philadelphie, Pennsylvanie - Etats-Unis

Siège social administratif : Paris - France

Numéro RCS : 486 794 234 RCS Paris

Numéro SIRET : 894 134 555 07803

Email : contact@GSB.com

Téléphone : +33 1 23 45 67 89

Directeur de la publication : Doseille Patrique

## 2. Hébergeur du site

Nom de l'hébergeur : Vandam Jean-Claude

Adresse : Pas ici en tout cas

Téléphone : 06 03 23 60 30

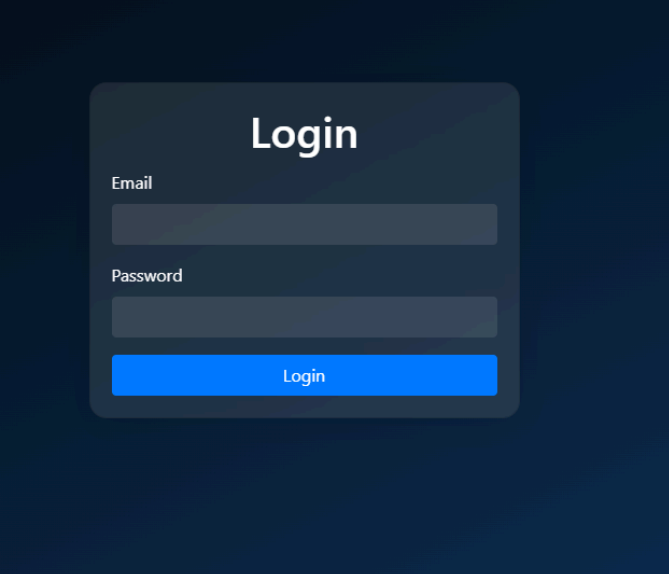
## 3. Responsabilité

L'éditeur du site se réserve le droit de modifier, compléter ou supprimer tout contenu du site, en particulier en cas de violation de droits de tiers.

## 4. Propriété intellectuelle

L'ensemble du contenu du site (textes, images, logos, etc.) est la propriété exclusive de Galaxy Swiss Bourdin ou fait l'objet d'une autorisation d'utilisation.

## Page Login :

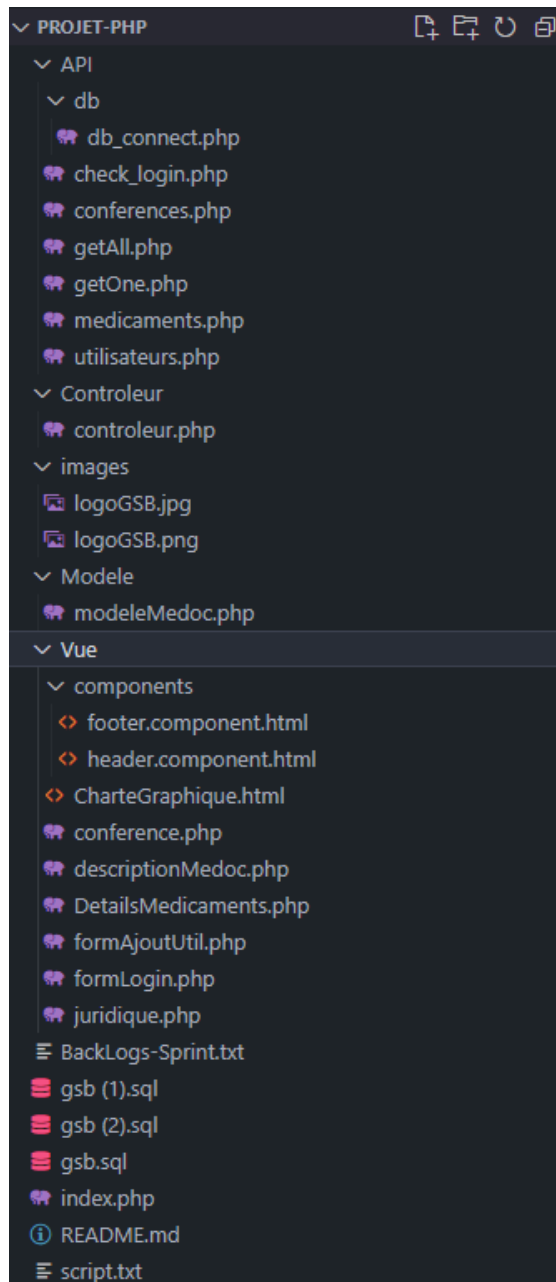
A login form interface with a dark blue background. The form is a lighter blue rounded rectangle. At the top, the word "Login" is written in white. Below it, there are two input fields: "Email" and "Password", both with white text. The "Email" field is above the "Password" field. At the bottom of the form, there is a blue button with the word "Login" in white.

## Page nosChercheurs:

### Liste des chercheurs

ID	Nom	Prenom	Specialite
1	Houel	Nicolas	Apothicaire
2	Charas	Moyse	Apothicaire
3	Parmentier	Antoine Augustin	Chimie alimentaire
4	Fourneau	Ernest	Chimie therapeutique

## Structure applicative :



# Charte graphique :

## Logo :



## Couleurs du site :

fond : #FFFFFF

header / footer : #C3E8F8

Titre (h1/h2/h3) : #00A1D4



## Typologie utiliser :

**Poppins**

**GSB**

:: Caractères de la police

A	B	C	D	E	F	G	H	I
J	K	L	M	N	O	P	Q	R
S	T	U	V	W	X	Y	Z	
a	b	c	d	e	f	g	h	i
j	k	l	m	n	o	p	q	r
s	t	u	v	w	x	y	z	
0	1	2	3	4	5	6	7	8
9	.	,	;	:	\$	#	'	!
"	/	?	%	&	(	)	@	

## Cahier de suivi Agile :

### Tâches à faire :

#### Structure :

Faire la partie Modèle  
Faire la partie Vue  
faire login  
ajout de composants pour les éléments récurrents  
Faire la partie Controleur  
Ajouter bootstrap

#### Gestion des médicaments :

afficher la liste des médicaments  
Afficher les effets thérapeutiques et les effets secondaires  
Interaction avec d'autres médicaments  
Afficher les détails d'un médicaments et de tous les médicaments

#### Gestion de la BD :

Faire la base de données  
Faire 2 triggers coté BD  
script sauvegarde de BD  
Faire MCD  
Faire MLD  
Faire la charte graphique

ChartDown  
Mettre une page juridique

### Sprints à faire :

faire structure

faire bd

Gérer juridique

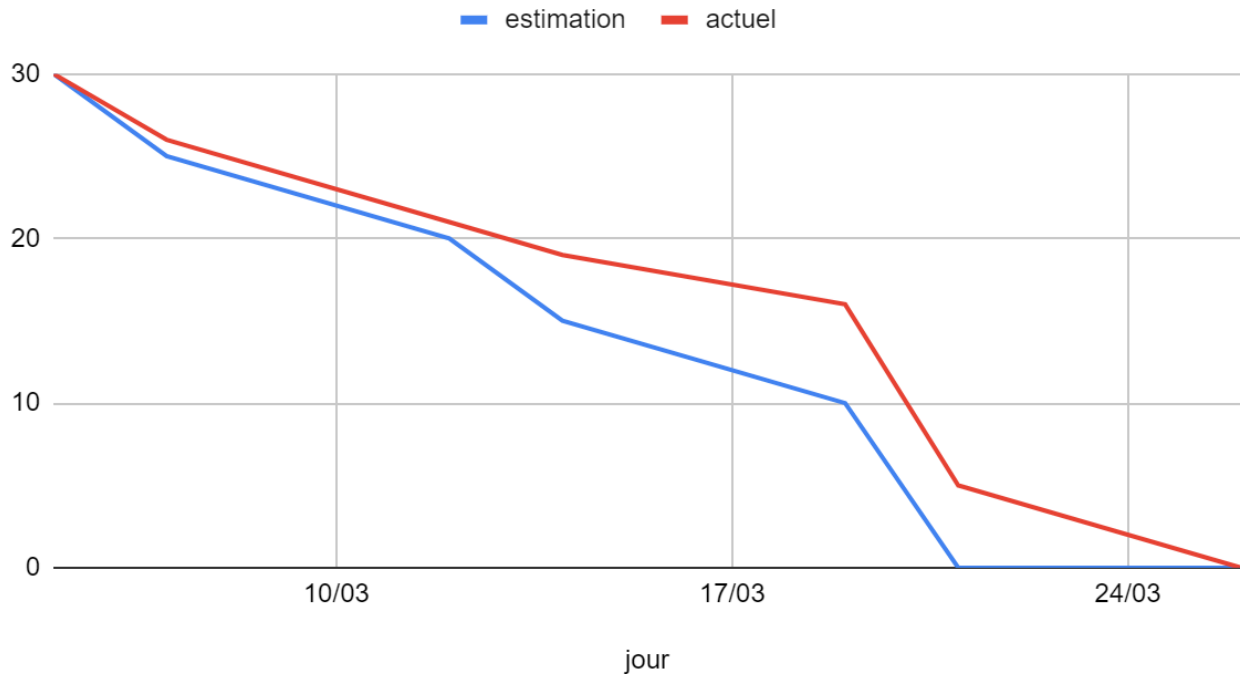
Afficher les détails d'un médicaments et de tous les médicaments  
faire structure

Créer, modifier, supprimer et s'inscrire, se désinscrire d'une activité  
autres

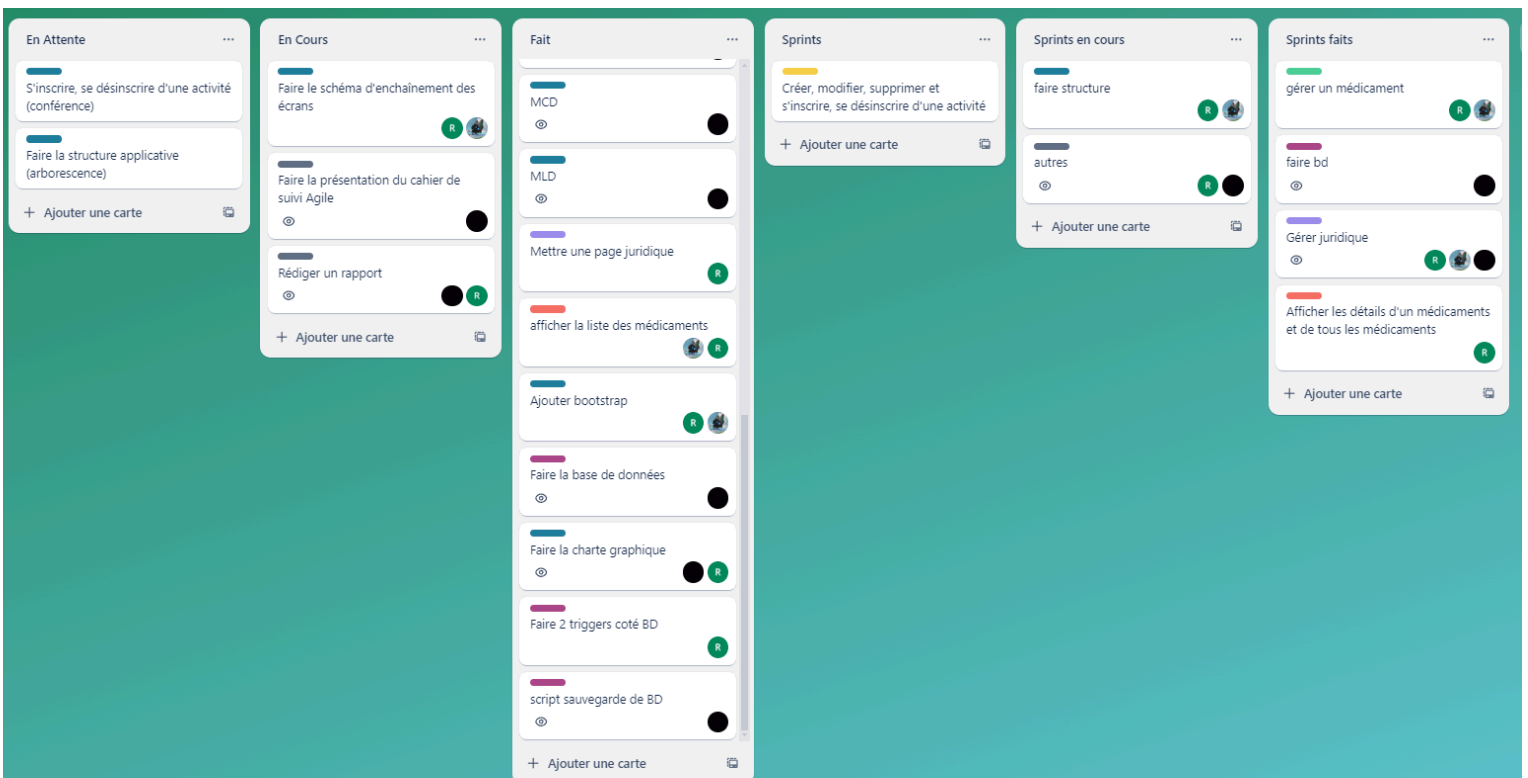


# Charte Down :

## Chartdown



## Tableau de progression :



## Mentions Légales :

### 1. Informations légales

Raison sociale : Galaxy Swiss Bourdin

Forme juridique : Société Anonyme

Capital social : 60 000 000 €

Siège social : Philadelphie, Pennsylvanie - Etats-Unis

Siège social administratif : Paris - France

Numéro RCS : 486 794 234 RCS Paris

Numéro SIRET : 894 134 555 07803

Email : [contact@GSB.com](mailto:contact@GSB.com)

Téléphone : +33 1 23 45 67 89

Directeur de la publication : Doseille Patrique

### 2. Hébergeur du site

Nom de l'hébergeur : Vandam Jean-Claude

Adresse : Pas ici en tout cas

Téléphone : 06 03 23 60 30

### 3. Responsabilité

L'éditeur du site se réserve le droit de modifier, compléter ou supprimer tout contenu du site, en particulier en cas de violation de droits de tiers.

#### 4. Propriété intellectuelle

L'ensemble du contenu du site (textes, images, logos, etc.) est la propriété exclusive de Galaxy Swiss Bourdin ou fait l'objet d'une autorisation d'utilisation.

#### 5. Cookies

Ce site utilise des cookies pour améliorer l'expérience utilisateur. En naviguant sur ce site, vous acceptez l'utilisation de ces cookies.

#### 6. Données personnelles

Les données personnelles collectées sur ce site sont utilisées uniquement dans le cadre prévu par la loi. Pour en savoir plus, consultez notre politique de confidentialité.

#### 7. Droit à l'image

Toutes les images utilisées sur ce site sont soit la propriété de Galaxy Swiss Boudin, soit utilisées avec la permission de leurs propriétaires respectifs.

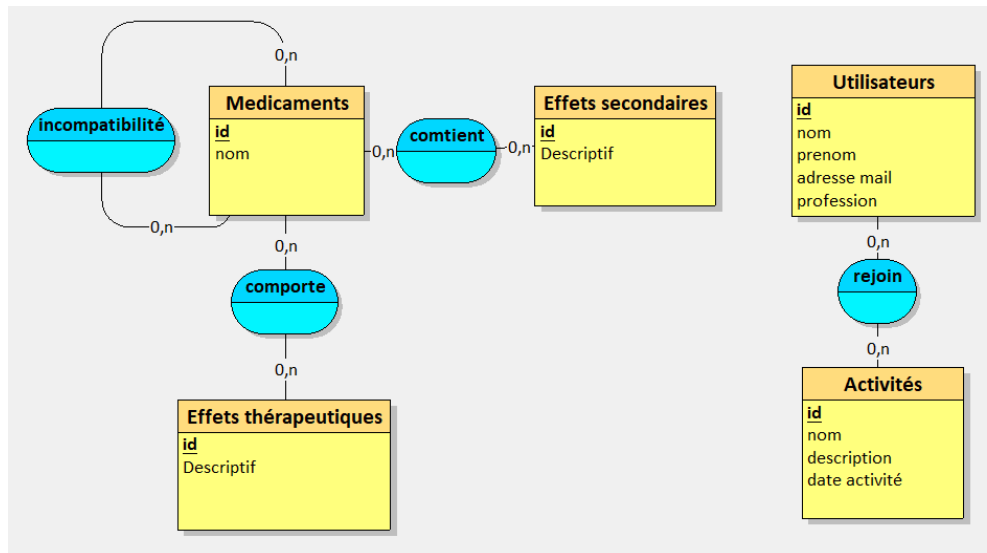
Si vous estimez qu'une image ne devrait pas apparaître, veuillez nous contacter pour que nous puissions résoudre le problème.

Les images de personnes présentes sur le site ont été utilisées avec leur consentement ou sont utilisées dans le respect des lois sur le droit à l'image.

Si vous souhaitez retirer une image de ce site, veuillez nous contacter avec les détails appropriés.

## Structure BD :

### MCD:



### MLD:

Médicaments = (id INT, nom VARCHAR(50));

Utilisateurs = (id INT, nom VARCHAR(50), prenom VARCHAR(50),  
adresse\_mail INT), profession VARCHAR(50));

Activités = (id INT, nom VARCHAR(50), description VARCHAR(50),  
date\_activité DATE);

Effets\_thérapeutiques = (id INT, Descriptif VARCHAR(50));

Effets\_secondaires = (id INT, Descriptif VARCHAR(50));

rejoin = (#id, #id\_1);

comporte = (#id, #id\_1);

incompatibilité = (#id, #id\_1);

contient = (#id, #id\_1);

```
@import url('https://fonts.cdnfonts.com/css/poppins');
```

```
<link href="https://fonts.cdnfonts.com/css/poppins" rel="stylesheet">
```

## Code SQL :

```
SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";

START TRANSACTION;

SET time_zone = "+00:00";

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;

/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS
*/;

/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;

/*!40101 SET NAMES utf8mb4 */;

-- Base de données : `gsb`

-- Structure de la table `activités`

DROP TABLE IF EXISTS `activités`;

CREATE TABLE IF NOT EXISTS `activités` (

  `id` int(11) NOT NULL AUTO_INCREMENT,

  `nom` varchar(50) DEFAULT NULL,

  `description` varchar(50) DEFAULT NULL,

  `date_activité` date DEFAULT NULL,

  PRIMARY KEY (`id`)

) ENGINE=InnoDB AUTO_INCREMENT=4 DEFAULT CHARSET=utf8mb4;

-- Déchargement des données de la table `activités`

INSERT INTO `activités` (`id`, `nom`, `description`, `date_activité`) VALUES

(1, 'conference1.0', 'reunion mensuelle', '2024-01-05'),

(2, 'conference2.0', 'reunion mensuelle', '2024-02-02'),

(3, 'conference3.0', 'reunion mensuelle', '2024-03-01');
```

```

-- Déclencheurs `activités`

DROP TRIGGER IF EXISTS `check_date_activite_insert`;

DELIMITER $$

CREATE TRIGGER `check_date_activite_insert` BEFORE INSERT ON `activités` FOR
EACH ROW BEGIN

    IF NEW.date_activité < CURDATE() THEN

        SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Vous ne pouvez pas ajouter une
activité avec une date antérieure à la date actuelle.';

    END IF;

END

$$

DELIMITER ;

-- Structure de la table `comporte`

DROP TABLE IF EXISTS `comporte`;

CREATE TABLE IF NOT EXISTS `comporte` (

    `id_medocT` int(11) NOT NULL AUTO_INCREMENT,

    `idC` int(11) NOT NULL,

    PRIMARY KEY (`id_medocT`,`idC`),

    KEY `id_1` (`idC`)

) ENGINE=MyISAM AUTO_INCREMENT=17 DEFAULT CHARSET=utf8mb4;

-- Déchargement des données de la table `comporte`

INSERT INTO `comporte` (`id_medocT`, `idC`) VALUES

(1, 1),

(2, 2),

(3, 3),

```

(4, 4),  
(5, 5),  
(6, 6),  
(7, 7),  
(8, 8),  
(9, 9),  
(10, 10),  
(11, 11),  
(12, 12),  
(13, 13),  
(14, 14),  
(15, 15),  
(16, 16);

-- Structure de la table `contient`

DROP TABLE IF EXISTS `contient`;

CREATE TABLE IF NOT EXISTS `contient` (

  `id\_medocC` int(11) NOT NULL AUTO\_INCREMENT,

  `id` int(11) NOT NULL,

  PRIMARY KEY (`id\_medocC`, `id`),

  KEY `id\_1` (`id`)

) ENGINE=MyISAM AUTO\_INCREMENT=17 DEFAULT CHARSET=utf8mb4;

-- Déchargement des données de la table `contient`

INSERT INTO `contient` (`id\_medocC`, `id`) VALUES

(1, 1),

```

(2, 2),
(3, 3),
(4, 4),
(5, 5),
(6, 6),
(7, 7),
(8, 8),
(9, 9),
(10, 10),
(11, 11),
(12, 12),
(13, 13),
(14, 14),
(15, 15),
(16, 16);

-- Structure de la table `effets_secondaires`

DROP TABLE IF EXISTS `effets_secondaires`;

CREATE TABLE IF NOT EXISTS `effets_secondaires` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `Descriptif` varchar(500) DEFAULT NULL,
  PRIMARY KEY (`id`)
) ENGINE=MyISAM AUTO_INCREMENT=17 DEFAULT CHARSET=utf8mb4;

-- Déchargement des données de la table `effets_secondaires`

INSERT INTO `effets_secondaires` (`id`, `Descriptif`) VALUES

```



(1, 'Risque de lésions hépatiques en cas de surdosage.'),  
 (2, 'Irritation gastrique et ulcères.'),  
 (3, 'les nausées et les sensations vertigineuses'),  
 (4, 'Risque accru d'infections gastro-intestinales.'),  
 (5, 'Réactions allergiques, parfois sévères.'),  
 (6, 'Somnolence, dépendance physique et psychique.'),  
 (7, 'Palpitations cardiaques, troubles du sommeil.'),  
 (8, 'Augmente le risque de saignement gastrique et d'ulcères.'),  
 (9, 'Risque accru de douleurs musculaires et de lésions hépatiques.'),  
 (10, 'Constipation, dépression respiratoire.'),  
 (11, 'Troubles gastro-intestinaux, hypoglycémie.'),  
 (12, 'Insomnie, diminution de la libido.'),  
 (13, 'Risque accru de saignements, interactions médicamenteuses.'),  
 (14, 'Toux sèche, hypotension.'),  
 (15, 'Somnolence, dépendance.'),  
 (16, 'Prise de poids, augmentation de la pression artérielle.');

-- Structure de la table `effets\_thérapeutiques`

DROP TABLE IF EXISTS `effets\_thérapeutiques`;

CREATE TABLE IF NOT EXISTS `effets\_thérapeutiques` (

  `id` int(11) NOT NULL AUTO\_INCREMENT,

  `Descriptif` varchar(500) DEFAULT NULL,

  PRIMARY KEY (`id`)

) ENGINE=MyISAM AUTO\_INCREMENT=17 DEFAULT CHARSET=utf8mb4;

-- Déchargement des données de la table `effets\_thérapeutiques`

INSERT INTO `effets\_thérapeutiques` (`id`, `Descriptif`) VALUES

- (1, 'Soulagement de la douleur et réduction de la fièvre.'),
- (2, 'Anti-inflammatoire non stéroïdien, soulage la douleur et réduit l'inflammation.'),
- (3, 'Soulage la douleur en agissant sur des cellules nerveuses particulières de la moelle épinière et du cerveau.'),
- (4, 'Réduction de la production d'acide gastrique, traitement des ulcères gastriques et du reflux gastro-œsophagien.'),
- (5, 'Antibiotique, traitement des infections bactériennes.'),
- (6, 'Anxiolytique, anticonvulsivant, relaxant musculaire.'),
- (7, 'Hormone thyroïdienne de remplacement, traitement de l'hypothyroïdie.'),
- (8, 'Analgésique, anti-inflammatoire, prévention des caillots sanguins.'),
- (9, 'Réduction du cholestérol, prévention des maladies cardiovasculaires.'),
- (10, 'Analgésique puissant, utilisé pour soulager la douleur sévère.'),
- (11, 'Traitement du diabète de type 2, aide à contrôler la glycémie.'),
- (12, 'Antidépresseur, traitement des troubles anxieux et de la dépression.'),
- (13, 'Anticoagulant, prévention des caillots sanguins.'),
- (14, 'Inhibiteur de l'enzyme de conversion de l'angiotensine, traitement de l'hypertension et de l'insuffisance cardiaque.'),
- (15, 'Anxiolytique, traitement des troubles anxieux et des crises de panique.'),
- (16, 'Corticostéroïde, traitement des inflammations et des réactions allergiques.');

-- Structure de la table `incompatibilité`

DROP TABLE IF EXISTS `incompatibilité`;

CREATE TABLE IF NOT EXISTS `incompatibilité` (

  `id` int(11) NOT NULL AUTO\_INCREMENT,

  `id\_1` int(11) NOT NULL,

  PRIMARY KEY (`id`,`id\_1`),

  KEY `id\_1` (`id\_1`)

) ENGINE=MyISAM AUTO\_INCREMENT=17 DEFAULT CHARSET=utf8mb4;

-- Déchargement des données de la table `incompatibilité`

INSERT INTO `incompatibilité` (`id`, `id\_1`) VALUES

(1, 2),

(2, 1),

(3, 4),

(4, 3),

(5, 6),

(6, 5),

(7, 8),

(8, 7),

(9, 10),

(10, 9),

(11, 12),

(12, 11),

(13, 14),

(14, 13),

(15, 16),

(16, 15);

-- Structure de la table `medicaments`

DROP TABLE IF EXISTS `medicaments`;

CREATE TABLE IF NOT EXISTS `medicaments` (

  `id` int(11) NOT NULL,

  `nom` varchar(50) DEFAULT NULL,

  `description` text NOT NULL,

PRIMARY KEY (`id`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8mb4;

-- Déchargement des données de la table `medicaments`

INSERT INTO `medicaments` (`id`, `nom`, `description`) VALUES

(1, 'Paracétamol', 'Le paracétamol, aussi appelé acétaminophène, est un composé chimique utilisé comme antalgique et antipyrétique, qui figure parmi les médicaments les plus communs, utilisés et prescrits au monde.'),

(2, 'Ibuprofène', 'Ibuprofène est la dénomination commune internationale de l'acide 2-4-propyl phénylpropanoïque. Il s'agit de la substance active d'un médicament AINS utilisé pour soulager les symptômes de l'arthrite, de la dysménorrhée primaire, de la pyrexie et comme analgésique, spécialement en cas d'inflammation.'),

(3, 'Tramadol', 'Le tramadol est un antalgique central développé par la firme allemande Grünenthal GmbH dans les années 1970. Il est classé dans la catégorie des analgésiques de niveau 2, catégorie comprenant également la codéine, les extraits d'opium et anciennement le dextropropoxyphène.'),

(4, 'Omeprazole', 'L'oméprazole est une substance active médicamenteuse de la famille des inhibiteurs de la pompe à protons qui réduit la sécrétion acide de l'estomac.'),

(5, 'Amoxicilline', 'L'amoxicilline est un antibiotique  $\beta$ -lactamine bactéricide de la famille des aminopénicillines indiqué dans le traitement des infections bactériennes à germes sensibles.'),

(6, 'Diazépam', 'Le diazépam est un médicament de la famille des benzodiazépines. Il possède des propriétés anxiolytiques, sédatives, amnésiantes et hypnotiques. Néanmoins, en raison de ses effets importants de somnolence résiduelle le lendemain matin, il est de moins en moins utilisé comme hypnotique en clinique.'),

(7, 'Lévothyroxine', 'La lévothyroxine, aussi connue sous le nom de L-thyroxine, T4 synthétique ou 3,5,3',5'-tetraiodo-L-thyronine, est une forme synthétique de la thyroxine, également utilisée comme médicament.'),

(8, 'Aspirine', 'L'acide acétylsalicylique, plus connu sous le nom commercial d'aspirine, est un anti-inflammatoire non stéroïdien. C'est la substance active de nombreux médicaments aux propriétés antalgiques, antipyrétiques et anti-inflammatoires. Il est surtout utilisé comme antiagrégant plaquettaire.'),

(9, 'Atorvastatine', 'L'atorvastatine est un médicament de type statine utilisé pour son action hypocholestérolémiante. Cette molécule a été découverte par la société américaine Warner-Lambert et lancée en 1997.'),

(10, 'Morphine', 'La morphine est le principal alcaloïde de l'opium, le latex du pavot somnifère. C'est une molécule complexe utilisée en médecine comme analgésique et comme drogue pour son action euphorisante. L'opium est utilisé en médecine depuis le III<sup>e</sup> millénaire av.'),

(11, 'Metformine', 'La metformine est un antidiabétique oral de la famille des biguanides normoglycémisants utilisé dans le traitement du diabète de type 2. Son rôle est de diminuer l'insulino-résistance de l'organisme intolérant aux glucides et de diminuer la néoglucogenèse hépatique.'),

(12, 'Citalopram', 'Le citalopram, ou plus exactement le bromhydrate de citalopram, est un antidépresseur inhibiteur sélectif de la recapture de la sérotonine, il est utilisé pour le traitement de la dépression, associé ou non à des troubles de l'humeur, et dans l'ensemble des troubles de l'anxiété.'),

(13, 'Warfarine', 'Le coumaphène ou warfarine est un composé organique de la famille des coumarines. C'est une substance active de produit phytosanitaire, qui présente un effet rodenticide, avec toutefois de possibles phénomènes de résistance aux rodenticides.'),

(14, 'Lisinopril', 'Le lisinopril est un médicament inhibiteur de l'enzyme de conversion. Il est utilisé pour traiter l'hypertension artérielle, l'insuffisance cardiaque et l'après crises cardiaques. Pour l'hypertension artérielle, il s'agit généralement d'un traitement de première intention.'),

(15, 'Alprazolam', 'L'alprazolam est une benzodiazépine de la famille des triazolobenzodiazépines utilisée comme un anxiolytique d'action rapide. En France, l'alprazolam est la benzodiazépine la plus prescrite et est connue sous son nom commercial Xanax.'),

(16, 'Prednisone', 'La prednisone est un corticostéroïde, prodrogue métabolisée par le foie par réduction du carbonyle en hydroxyde par la 11-β-oxydoréductase hépatique, en prednisolone.');

-- Structure de la table `rejoint`

DROP TABLE IF EXISTS `rejoint`;

CREATE TABLE IF NOT EXISTS `rejoint` (

  `id` int(11) NOT NULL,

  `id\_act` int(11) NOT NULL,

  PRIMARY KEY (`id`, `id\_act`),

  KEY `id\_act` (`id\_act`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

```

-- Structure de la table `utilisateurs`

DROP TABLE IF EXISTS `utilisateurs`;

CREATE TABLE IF NOT EXISTS `utilisateurs` (
  `id` int(50) NOT NULL,
  `nom` varchar(50) DEFAULT NULL,
  `prenom` varchar(50) DEFAULT NULL,
  `adresse_mail` varchar(50) DEFAULT NULL,
  `profession` varchar(50) DEFAULT NULL,
  `MDP` varchar(20) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

-- Déchargement des données de la table `utilisateurs`

INSERT INTO `utilisateurs` (`id`, `nom`, `prenom`, `adresse_mail`, `profession`, `MDP`)
VALUES
(1, 'Jean', 'Christophe', 'Jean.christophe@gmail.com', 'orthophoniste', 'Soleil123'),
(2, 'Andrer', 'quentin', 'quentin123@gmail.com', 'generaliste', '23'),
(3, 'Philippe', 'Baptiste', 'Baptiste.philipe@gmail.com', 'orl', 'gvojerzs'),
(4, 'admin', 'admin', 'admin@gmail.com', 'admin', 'root');

-- Déclencheurs `utilisateurs`

DROP TRIGGER IF EXISTS `check_profession_insert`;

DELIMITER $$

CREATE TRIGGER `check_profession_insert` BEFORE INSERT ON `utilisateurs` FOR
EACH ROW BEGIN

  IF NEW.profession = " OR NEW.profession IS NULL THEN

```

```
SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'La profession ne peut pas être vide.';
```

```
END IF;
```

```
END
```

```
$$
```

```
DELIMITER ;
```

```
-- Contraintes pour les tables déchargées
```

```
-- Contraintes pour la table `rejoint`
```

```
ALTER TABLE `rejoint`
```

```
ADD CONSTRAINT `rejoint_ibfk_1` FOREIGN KEY (`id`) REFERENCES `utilisateurs`  
(`id`),
```

```
ADD CONSTRAINT `rejoint_ibfk_2` FOREIGN KEY (`id_act`) REFERENCES `activités`  
(`id`);
```

```
COMMIT;
```

```
/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
```

```
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
```

```
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
```

## Code site applicative :

### API:

#### utilisateurs.php:

`include("../db/db_connect.php");`: Inclut le fichier de connexion à la base de données.

`$request_method = $_SERVER["REQUEST_METHOD"];`: Récupère la méthode HTTP utilisée dans la requête (GET, POST, etc.).

`switch ($request_method) { ... };` Commence une structure de commutation basée sur la méthode HTTP.

`case 'GET': ...;` Si la méthode de la requête est GET, exécute le code à l'intérieur.

`if (!empty($_GET["id"])) { ... } else { getUtilisateurs(); };` Vérifie s'il y a un paramètre "id" dans la requête GET. Si oui, il le récupère et le convertit en un entier. Sinon, il appelle la fonction `getUtilisateurs()`.

`function getUtilisateurs() { ... };` Définit la fonction pour récupérer tous les utilisateurs depuis la base de données.

`global $conn;`: Récupère la connexion à la base de données définie dans le fichier `db_connect.php`.

`$query = "SELECT * FROM utilisateurs";`: Définit la requête SQL pour sélectionner tous les utilisateurs dans la table "utilisateurs".

`$response = array();`: Initialise un tableau pour stocker les résultats.

`$conn->query("SET NAMES utf8");`: Définit le jeu de caractères à utiliser pour la connexion à la base de données.

`$result = $conn->query($query);`: Exécute la requête SQL.

`while($row = $result->fetch()) { ... };` Parcourt chaque ligne de résultat et les ajoute au tableau de réponse.

`$result->closeCursor();`: Ferme le curseur de résultat pour libérer les ressources.



`header('Content-Type: application/json');`:: Définit l'en-tête HTTP pour indiquer que le contenu renvoyé est au format JSON.

`echo json_encode($response, JSON_PRETTY_PRINT);`:: Convertit le tableau de réponse en format JSON et l'affiche, en utilisant l'option `JSON_PRETTY_PRINT` pour une meilleure lisibilité.

### [medicaments.php](#)

```
<?php

// Inclure le fichier de connexion à la base de données
include "../db/db_connect.php"

// Récupérer la méthode de requête HTTP (GET, POST, etc.)
$request_method  $_SERVER "REQUEST_METHOD"

// Gérer les différentes méthodes de requête
switch $request_method
    case 'GET'
        // Si un identifiant est fourni dans la requête GET
        if empty $_GET "id"
            $id  intval $_GET "id"
            // Appeler les fonctions pour récupérer les détails
            du médicament
                getMédicament $id
                effetT $id
                effetS $id
                incompatible $id
            else
                // Si aucun identifiant n'est fourni, récupérer tous
                les médicaments
                    getMédicaments

        break

// Fonction pour récupérer tous les médicaments
function getMédicaments
    global $conn
    // Requête SQL pour récupérer tous les médicaments de la base
    de données
```

```

$query    "SELECT * FROM medicaments"
$response  array

// Exécuter la requête SQL
$conn query "SET NAMES utf8"
$result   $conn query $query
// Boucler à travers les résultats et les stocker dans un
tableau
while $row $result fetch
    $response $row

// Fermer le curseur de résultat
$result closeCursor
// Définir l'en-tête de réponse comme JSON
header 'Content-Type: application/json'
// Encoder le tableau de résultats en JSON et l'afficher de
manière lisible
echo json_encode $response

// Fonction pour récupérer les détails d'un médicament spécifique
function getMedicament $id
    global $conn
    // Requête SQL pour récupérer les détails d'un médicament
    spécifique
    $query    "SELECT id, description FROM medicaments"
    if $id    0
        $query    " WHERE id ="    $id    " LIMIT 1"

    // Exécuter la requête SQL
    $conn query "SET NAMES utf8"
    $result   $conn query $query
    $response  array    // Initialiser le tableau de réponse

    // Boucler à travers les résultats et les stocker dans le
    tableau de réponse
    while $row $result fetch
        $response $row

```

```

// Fermer le curseur de résultat
$result closeCursor
// Définir l'en-tête de réponse comme JSON
header 'Content-Type: application/json'
// Encoder le tableau de résultats en JSON et l'afficher de
manière lisible
echo json_encode $response

// Fonction pour récupérer les effets thérapeutiques d'un
médicament
function effetT $id
    global $conn
    // Requête SQL pour récupérer les effets thérapeutiques d'un
médicament
    $query "SELECT effets_thérapeutiques.id, Descriptif as
description FROM effets_thérapeutiques
        JOIN comporte ON effets_thérapeutiques.id =
comporte.idC
        JOIN Medicaments ON comporte.id_medocT =
Medicaments.id
        WHERE Medicaments.id =" $id
    $response array

    // Exécuter la requête SQL
    $conn query "SET NAMES utf8"
    $result $conn query $query
    // Boucler à travers les résultats et les stocker dans le
tableau de réponse
    while $row $result fetch
        $response $row

// Fermer le curseur de résultat
$result closeCursor
// Définir l'en-tête de réponse comme JSON
header 'Content-Type: application/json'
// Encoder le tableau de résultats en JSON et l'afficher de
manière lisible
echo json_encode $response

// Fonction pour récupérer les effets secondaires d'un médicament

```

```

function effetsS $id
    global $conn
    // Requête SQL pour récupérer les effets secondaires d'un
    médicament
    $query "SELECT effets_secondaires.id, Descriptif as
description FROM effets_secondaires
        JOIN contient ON effets_secondaires.id = contient.id
        JOIN Medicaments ON contient.id_medocC =
Medicaments.id
        WHERE Medicaments.id =" $id
    $response array

    // Exécuter la requête SQL
    $conn query "SET NAMES utf8"
    $result $conn query $query
    // Boucler à travers les résultats et les stocker dans le
    tableau de réponse
    while $row $result fetch
        $response $row

    // Fermer le curseur de résultat
    $result closeCursor
    // Définir l'en-tête de réponse comme JSON
    header 'Content-Type: application/json'
    // Encoder le tableau de résultats en JSON et l'afficher de
    manière lisible
    echo json_encode $response

// Fonction pour récupérer les médicaments incompatibles avec un
médicament spécifique
function incompatible $id
    global $conn
    // Requête SQL pour récupérer les médicaments incompatibles
    avec un médicament spécifique
    $query "SELECT medicaments.id, medicaments.nom as
description FROM medicaments
        JOIN incompatibilité ON medicaments.id =
incompatibilité.id
        WHERE incompatibilité.id_1 =" $id
    $response array

```

```

// Exécuter la requête SQL
$conn query "SET NAMES utf8"
$result $conn query $query

// Boucler à travers les résultats et les stocker dans le
tableau de réponse

```

### getOne.php

```

<?php
// URL de l'API pour récupérer les détails d'un médicament
avec l'identifiant 1
$url "http://127.0.0.1/API/medicaments.php?id=1"

// Options de la requête HTTP
$options array
    'http' array
        'header' "Content-Type:
application/x-www-form-urlencoded\r\n"
        'method' 'GET'

// Création du contexte pour la requête HTTP
$context stream_context_create $options

// Effectuer une requête HTTP GET à l'URL spécifiée avec les
options spécifiées
// et récupérer le contenu de la réponse
$result file_get_contents $url false $context

// Afficher le contenu de la réponse pour le débogage
var_dump $result
?>

```

### getAll.php

```

<?php
// URL de l'API pour récupérer les données sur les
médicaments
$url = "http://127.0.0.1/API/medicaments.php";

// Options de la requête HTTP

```

```

$options = array(
    'http' => array(
        'header' => "Content-Type:
application/x-www-form-urlencoded\r\n", // En-tête de la requête
        'method' => 'GET' // Méthode de la requête
    )
);

// Création du contexte pour la requête HTTP
$context = stream_context_create($options);

// Effectuer une requête HTTP GET à l'URL spécifiée avec les
options spécifiées
// et récupérer le contenu de la réponse
$result = file_get_contents($url, false, $context);

// Supprimer le premier caractère de la réponse (utile dans
certains cas)
$string = substr($result, 1);

// Décoder la réponse JSON en tableau associatif
$string_decode = json_decode($string, true);

// Afficher les données sous forme de liste non ordonnée HTML
echo "<ul>";
for ($i = 0; $i < count($string_decode); $i++) {
    echo "<li>".$string_decode[$i]["nom"]."</li>"; //
Afficher le nom de chaque médicament
}
echo "</ul>";
?>

```

### [conferences.php](#)

```

<?php
// Inclure le fichier de connexion à la base de données
include("../db/db_connect.php");

// Récupérer la méthode de requête HTTP (GET, POST, etc.)
$request_method = $_SERVER["REQUEST_METHOD"];

```

```

// Gérer les différentes méthodes de requête
switch ($request_method) {
    case 'GET':
        // Si un identifiant est fourni dans la requête GET
        if (!empty($_GET["id"])) {
            $id = intval($_GET["id"]);
        } else {
            // Si aucun identifiant n'est fourni, récupérer
            toutes les conférences
            getConferences();
        }
        break;
    case 'POST':
        // Si la méthode de requête est POST, inscrire
        l'utilisateur à une conférence
        inscriptAct();
        break;
}

// Fonction pour récupérer toutes les conférences
function getConferences(){
    global $conn;
    // Requête SQL pour récupérer toutes les conférences de la
    base de données
    $query = "SELECT * FROM activités";
    $response = array();

    // Exécuter la requête SQL
    $conn->query("SET NAMES utf8");
    $result = $conn->query($query);
    // Boucler à travers les résultats et les stocker dans un
    tableau
    while($row = $result->fetch()){
        $response[] = $row;
    }

    // Fermer le curseur de résultat
    $result->closeCursor();
    // Définir l'en-tête de réponse comme JSON
    header('Content-Type: application/json');
    // Encoder le tableau de résultats en JSON et l'afficher de
    manière lisible

```

```

        echo json_encode($response, JSON_PRETTY_PRINT);
    }

    // Fonction pour inscrire l'utilisateur à une conférence
    function inscriptAct(){
        global $conn;
        // Récupérer l'identifiant de l'utilisateur et l'identifiant
        de la conférence depuis la requête POST
        $id_user = $_POST["id_user"];
        $id_act = $_POST["id_act"];
        // Requête SQL pour inscrire l'utilisateur à la conférence
        spécifiée
        $query = "INSERT INTO rejoint VALUES ($id_user, $id_act)";
        $conn->query("SET NAMES utf8");

        // Exécuter la requête SQL et vérifier si l'insertion a
        réussi ou non
        if($conn->query($query)){
            $response = array('status' => 1, 'status_message' =>
            'Insertion réussie'); // Message de réussite
        } else {
            $response = array('status' => 0, 'status_message' =>
            'Erreur d'insertion'); // Message d'erreur
        }

        // Définir l'en-tête de réponse comme JSON
        header('Content-Type: application/json');
        // Encoder le tableau de réponse en JSON et l'afficher
        echo json_encode($response);
    }
    ?>

```

### check\_login.php

```

<?php
// Inclure le fichier de connexion à la base de données
include("../db/db_connect.php");

// Vérifier si la méthode de requête est POST
if ($_SERVER['REQUEST_METHOD'] === 'POST') {
    // Récupérer le login et le mot de passe depuis la requête
    POST
    $login = $_POST['login'];

```



```

$password = $_POST['password'];

// Vérifier si le mot de passe n'est pas nul et correspond au
mot de passe enregistré dans la base de données
if ((getPW($login, $conn) == $password) && $password != null)
{
    // Si le mot de passe correspond, retourner un message de
succès
    $response = [
        'status' => 'success',
        'message' => 'Login successful'
    ];
} else {
    // Sinon, retourner un message d'erreur
    $response = [
        'status' => 'error',
        'message' => 'Invalid login or password'
    ];
}

// Définir l'en-tête de réponse comme JSON
header('Content-Type: application/json');
// Encoder le tableau de réponse en JSON et l'afficher
echo json_encode($response);
} else {
    // Si la méthode de requête n'est pas POST, retourner un
message d'erreur
    $response = [
        'status' => 'error',
        'message' => 'Invalid request method'
    ];

    // Définir l'en-tête de réponse comme JSON
    header('Content-Type: application/json');
    // Encoder le tableau de réponse en JSON et l'afficher
    echo json_encode($response);
}

// Fonction pour récupérer le mot de passe d'un utilisateur à
partir de la base de données
function getPW($login, $conn){

```

```

        // Requête SQL pour récupérer le mot de passe de
        l'utilisateur avec le login spécifié
        $query = "SELECT MDP FROM utilisateurs WHERE adresse_mail =
        :login";
        $stmt = $conn->prepare($query);
        $stmt->bindParam(':login', $login);
        $stmt->execute();
        $result = $stmt->fetch();

        // Vérifier si le résultat contient une valeur pour le mot de
        passe
        if (isset($result['MDP'])) {
            return $result['MDP']; // Retourner le mot de passe
        } else {
            return null; // Retourner null si le mot de passe
            n'existe pas (ou gérer le cas selon les besoins)
        }
    }
}
?>

```

### db\_connect.php

```

<?php
    // Informations de connexion à la base de données
    $serveurName    'localhost' // Nom du serveur
    $username       'root'      // Nom d'utilisateur
    $password       ''          // Mot de passe (vide dans cet exemple)

    // Création d'une nouvelle instance de la classe PDO pour
    établir la connexion à la base de données MySQL
    $conn    new PDO "mysql:host=$serveurName;dbname=gsb"
    $username $password

    // Configuration de PDO pour signaler les erreurs
    $conn    setAttribute      ATTR_ERRMODE
    ERRMODE_EXCEPTION

?>

```

### controleur.php

<?php

// Fonction pour afficher les médicaments

function displayMedocs

```

// Démarrer la session si elle n'est pas déjà démarrée

if isset $_SESSION

    session_start

// Vérifier si l'utilisateur est connecté

if isset $_SESSION 'login'

    // Afficher le formulaire de connexion s'il n'est pas connecté

    require_once "vue/formLogin.php"

else

    // Récupérer les données sur les médicaments depuis l'API et
    // afficher les détails

    $medicaments_json
    file_get_contents 'http://localhost/phpgroupe/api/medicaments.php'

    $medicaments json_decode $medicaments_json true

    require_once "vue/detailsMédicaments.php"

// Fonction pour connecter l'utilisateur

function loginUser $login $password

    // Démarrer la session

    session_start

    // Vérifier les informations de connexion

    if checkLogin $login $password

        // Afficher un message d'erreur si les informations de
        // connexion sont incorrectes

        $_SESSION 'message' "Login ou mot de passe incorrect"

```

```

require_once "vue/formLogin.php"

else

    // Connecter l'utilisateur et afficher les détails des
    médicaments

    $_SESSION['login'] = $login

    displayMedocs

// Fonction pour déconnecter l'utilisateur

function logoutUser

    // Démarrer la session

    session_start

    // Détruire la session

    session_destroy

    // Rediriger vers la page d'accueil

    header "Location: index.php"

// Fonction pour afficher les détails d'un médicament

function details $id_medicament

    try

        // Récupérer les détails du médicament depuis l'API et afficher
        la vue correspondante

        $detail_json

        file_get_contents "http://localhost/phpgroupe/api/medicaments.php?id='
        $id_medicament

```

```
// Remplacer les caractères spéciaux pour assurer la validité  
du JSON
```

```
$detail_json    str_replace "[\]" "\", " $detail_json
```

```
$details    json_decode $detail_json true
```

```
require_once "vue/descriptionMedoc.php"
```

```
catch Exception $e
```

```
// Afficher une erreur en cas d'exception
```

```
echo 'Erreur : ' $e getMessage
```

```
// Fonction pour afficher les conférences
```

```
function conferences
```

```
// Récupérer les données sur les conférences depuis l'API et  
afficher la vue correspondante
```

```
$conferences_json
```

```
file_get_contents 'http://localhost/phpgroupe/api/conferences.php'
```

```
$conferences    json_decode $conferences_json true
```

```
// Récupérer les données sur les utilisateurs depuis l'API
```

```
$utilisateurs_json
```

```
file_get_contents 'http://localhost/phpgroupe/api/utilisateurs.php'
```

```
$utilisateurs    json_decode $utilisateurs_json true
```

```
require_once "vue/conference.php"
```

```
// Fonction pour rejoindre une activité
```

```

function rejoindreAct

    // Récupérer les données du formulaire

    $data    array 'id_user'    $_POST "user-select"    'id_act'
    $_POST "id_act"

    // Préparer les options pour la requête HTTP

    $options    array 'http'    array 'header'    "Content-Type:
application/x-www-form-urlencoded\r\n"    'method'    'POST'    'content'
    http_build_query $data

    $appel    'http://localhost/phpgroupe/api/conferences.php'

    // Effectuer la requête HTTP pour rejoindre l'activité

    $context    stream_context_create $options

    $result    file_get_contents $appel    false    $context

    // Démarrer la session

    session_start

    // Afficher un message en fonction du résultat de la requête

    if json_decode $result    true    'status'    1

        $message    "Vous avez rejoint la conférence"

        $couleurMessage    "alert-success"

    else

        $message    "Erreur lors de l'inscription"

        $couleurMessage    "alert-danger"

    $_SESSION 'divMessage'    '<div class="alert '    $couleurMessage
'">'

    $_SESSION 'messageConf'    $message

```

```
// Rediriger vers la page de conférences
```

```
header "Location: index.php?action=CO"
```

```
// Fonction pour afficher les informations juridiques
```

```
function juridique
```

```
// Afficher la page des informations juridiques
```

```
require_once "vue/juridique.php"
```

```
?>
```

modeleMedoc.php :

```
<?php
```

```
// Fonction pour vérifier les informations de connexion
```

```
function checkLogin($login, $password)
```

```
{
```

```
    // URL de l'API pour vérifier les informations de connexion
```

```
    $url = "http://localhost/phpgroupe/api/check_login";
```

```
    // Données à envoyer à l'API (login et mot de passe)
```

```
    $data = array('login' => $login, 'password' => $password);
```

```
    // Options pour la requête HTTP POST
```

```
    $options = array(
```

```

    'http' => array(
        'header' => "Content-type: application/x-www-form-urlencoded\r\n",
        'method' => 'POST',
        'content' => http_build_query($data) // Convertir les données en chaîne de
requête URL
    )
);

// Créer le contexte de la requête HTTP
$context = stream_context_create($options);

// Effectuer la requête HTTP POST et récupérer la réponse
$result = file_get_contents($url, false, $context);

// Décoder la réponse JSON en tableau associatif
$response = json_decode($result, true);

// Vérifier si la réponse de l'API est 'success'
if ($response['status'] == 'success') {
    return true; // Retourner vrai si l'authentification est réussie
} else {
    return false; // Retourner faux si l'authentification a échoué
}
}
?>

```



Vue :

juridique.php:

```
<!DOCTYPE html>

<html lang="fr">

<head>

    <!-- Balises meta pour la définition du jeu de caractères et du viewport -->

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1,
shrink-to-fit=no">

    <!-- Titre de la page -->

    <title>Mentions Légales</title>

    <!-- Lien vers la feuille de style Bootstrap CSS -->

    <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css">

    <!-- Style CSS personnalisé -->

    <style>

        .container {

            margin-top: 30px;

        }

    </style>

</head>

<body>

    <?php include "../Vue/components/header.component.html"; ?>

    <!-- Contenu de la page -->
```

```
<div class="container" id="mentions-legales">

  <div class="row">

    <div class="col-lg-12">

      <!-- Titre de la section "Mentions Légales" -->

      <h1 class="mt-5">Mentions Légales</h1>


      <!-- Informations légales -->

      <h2>1. Informations légales</h2>

      <!-- Détails sur la raison sociale, la forme juridique, etc. -->


      <!-- Hébergeur du site -->

      <h2>2. Hébergeur du site</h2>

      <!-- Détails sur l'hébergeur du site -->


      <!-- Responsabilité -->

      <h2>3. Responsabilité</h2>

      <!-- Informations sur la responsabilité de l'éditeur du site -->


      <!-- Propriété intellectuelle -->

      <h2>4. Propriété intellectuelle</h2>

      <!-- Informations sur les droits de propriété intellectuelle -->


      <!-- Cookies -->

      <h2>5. Cookies</h2>

      <!-- Informations sur l'utilisation des cookies -->
```

```

        <!-- Données personnelles -->

        <h2>6. Données personnelles</h2>

        <!-- Informations sur la collecte et l'utilisation des données personnelles -->


    </div>

</div>

</div>

<!-- Section Droit à l'image -->
<div class="container" id="droit-image">
    <div class="row">
        <div class="col-lg-12">
            <!-- Droit à l'image -->

            <h2>7. Droit à l'image</h2>

            <!-- Informations sur les droits à l'image -->

        </div>
    </div>
</div>

<!-- Section Contact -->
<div class="container" id="contact">
    <div class="row">
        <div class="col-lg-12">
            <!-- Formulaire de contact ou autres informations pertinentes -->

```

```

        </div>

    </div>

</div>

<!-- Pied de page Bootstrap -->

<?php include "../Vue/components/footer.component.html"; ?>

<!-- Scripts JavaScript -->

<!-- Bibliothèque jQuery -->

<script src="https://code.jquery.com/jquery-3.3.1.slim.min.js"></script>

<!-- Bibliothèque Popper.js -->

<script
src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"></scri
pt>

<!-- Bibliothèque Bootstrap JS -->

<script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js"></script
>

</body>

</html>

formLogin.php

<!DOCTYPE html>

<html lang="en">

<head>

    <!-- Déclaration du jeu de caractères -->

    <meta charset="UTF-8">

    <!-- Titre de la page -->

```

```

<title>Login</title>

<!-- Lien vers la feuille de style Bootstrap -->

<link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"

integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9
JvoRxT2MZw1T" crossorigin="anonymous">

<!-- Style CSS personnalisé -->

<style>

    /* Fond et couleur du texte du corps de la page */

    body {

        background: linear-gradient(to bottom right, #000000, #113a66);

        color: white;

    }

    /* Style de la carte de connexion */

    .card {

        background: rgba(142, 142, 142, 0.19);

        border-radius: 16px;

        box-shadow: 0 4px 30px rgba(0, 0, 0, 0.1);

        backdrop-filter: blur(5px);

        -webkit-backdrop-filter: blur(5px);

        color: white;

    }

    /* Style des champs de formulaire */

```

```
.form-control {  
    background-color: rgba(255, 255, 255, 0.1);  
    border: 1px solid rgba(255, 255, 255, 0.1);  
    color: white;  
}
```

*/\* Style des champs de formulaire au focus \*/*

```
.form-control:focus {  
    background-color: rgba(255, 255, 255, 0.2);  
    border: 1px solid rgba(255, 255, 255, 0.2);  
    color: white;  
}
```

*/\* Style du bouton de connexion \*/*

```
.btn-primary {  
    background-color: #007bff;  
    border-color: #007bff;  
}
```

*/\* Style du bouton de connexion au survol \*/*

```
.btn-primary:hover {  
    background-color: #0056b3;  
    border-color: #0056b3;  
}
```

```

</style>

</head>

<body>

<!-- Contenu de la page -->

<div class="container d-flex justify-content-center align-items-center" style="height:
100vh;">

  <div class="card" style="width: 400px;">

    <div class="card-body">

      <!-- Titre de la carte de connexion -->

      <h1 class="card-title text-center">Login</h1>

      <!-- Affichage du message d'erreur s'il existe -->

      <?php if(isset($_SESSION['message'])): ?>

        <div class="alert alert-danger"><?php echo $_SESSION['message'];
?></div>

        <!-- Suppression du message d'erreur de la session après affichage -->

        <?php unset($_SESSION['message']); ?>

      <?php endif; ?>

      <!-- Formulaire de connexion -->

      <form method="POST" action="index.php?action=LG">

        <div class="form-group">

          <label for="username">Email</label>

          <input type="text" class="form-control" id="username"
name="username">

        </div>

```

```

        <div class="form-group">

            <label for="password">Password</label>

            <input type="password" class="form-control" id="password"
name="password">

        </div>

        <button type="submit" class="btn btn-primary btn-block">Login</button>

    </form>

</div>

</div>

</div>

<!-- Lien vers les scripts JavaScript Bootstrap et jQuery -->

<script src="https://code.jquery.com/jquery-3.3.1.slim.min.js"></script>

<script
src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.7/umd/popper.min.js"></scri
pt>

<script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/js/bootstrap.min.js"></script
>

</body>

</html>

formAjoutUtil.php

<!DOCTYPE html>

<html lang="en">

<head>

    <!-- Déclaration du jeu de caractères -->

    <meta charset="UTF-8">

```



```

<!-- Titre de la page -->

<title>Login</title>

<!-- Lien vers la feuille de style Bootstrap -->

<link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.css"

integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9
JvoRxT2MZw1T" crossorigin="anonymous">

<!-- Style CSS personnalisé -->

<style>

    /* Animation de fondu */

    @keyframes fadeInAnimation {

        0% {

            opacity: 0;

        }

        100% {

            opacity: 1;

        }

    }

    /* Fond et couleur du texte du corps de la page */

    body {

        background: linear-gradient(to bottom right, #000000, #113a66);

        color: white;

    }

```

```
/* Style de la carte de connexion */  
  
.card {  
  
    background: rgba(142, 142, 142, 0.19);  
  
    border-radius: 16px;  
  
    box-shadow: 0 4px 30px rgba(0, 0, 0, 0.1);  
  
    backdrop-filter: blur(5px);  
  
    -webkit-backdrop-filter: blur(5px);  
  
    color: white;  
  
}
```

```
/* Style des champs de formulaire */  
  
.form-control {  
  
    background-color: rgba(255, 255, 255, 0.1);  
  
    border: 1px solid rgba(255, 255, 255, 0.1);  
  
    color: white;  
  
}
```

```
/* Style des champs de formulaire au focus */  
  
.form-control:focus {  
  
    background-color: rgba(255, 255, 255, 0.2);  
  
    border: 1px solid rgba(255, 255, 255, 0.2);  
  
    color: white;  
  
}
```

```
/* Style du bouton de connexion */
```

```

.btn-primary {
    background-color: #007bff;
    border-color: #007bff;
}

/* Style du bouton de connexion au survol */
.btn-primary:hover {
    background-color: #0056b3;
    border-color: #0056b3;
}

/* Animation de fondu pour la transition entre les parties du formulaire */
#part2 {
    animation: fadeInAnimation ease 1s;
    animation-iteration-count: 1;
    animation-fill-mode: forwards;
    opacity: 1;
}

/* Style pour masquer les parties du formulaire non actives */
#part1.hidden, #part2.hidden {
    display: none;
}

</style>

```

```

</head>

<body>

<!-- Contenu de la page -->

<div class="container d-flex justify-content-center align-items-center" style="height:
100vh;">

    <div class="card" style="width: 400px;">

        <div class="card-body">

            <!-- Affichage du message d'erreur s'il existe -->

            <?php if(isset($message)): ?>

                <div class="alert alert-danger"><?php echo $message; ?></div>

            <?php endif; ?>

            <!-- Formulaire de connexion -->

            <form method="POST" action="Controleur/controleur.php?action=lg">

                <!-- Première partie du formulaire -->

                <div id="part1">

                    <h1 class="card-title text-center">Vos informations</h1>

                    <div class="form-group">

                        <label for="username">Nom</label>

                        <input type="text" class="form-control" id="name" name="name">

                    </div>

                    <div class="form-group">

                        <label for="password">Prénom</label>

                        <input type="text" class="form-control" id="surname"
name="surname">

```

```

</div>

<div class="form-group">

    <label for="profession">Profession</label>

    <input type="text" class="form-control" id="job" name="job">

</div>

    <button type="button" class="btn btn-primary btn-block"
onclick="next_part()">Suivant</button>

</div>

<!-- Deuxième partie du formulaire, initialement cachée -->

<div id="part2" class="hidden">

    <h1 class="card-title text-center">Créer un compte</h1>

    <div class="form-group">

        <label for="username">Email</label>

        <input type="text" class="form-control" id="username"
name="username">

    </div>

    <div class="form-group">

        <label for="password">Mot de passe</label>

        <input type="password" class="form-control" id="password"
name="password">

    </div>

    <button type="submit" class="btn btn-primary
btn-block">Valider</button>

</div>

</form>

</div>

</div>

```

```
</div>
```

```
<!-- Script JavaScript pour la transition entre les parties du formulaire -->
```

```
<script>
```

```
function next_part() {
```

```
    var part1 = document.getElementById("part1");
```

```
    var part2 = document.getElementById("part2");
```

```
    // Ajoute la classe hidden pour masquer part1
```

```
    part1.classList.add("hidden");
```

```
    // Supprime la classe hidden pour afficher part2
```

```
    part2.classList.remove("hidden");
```

```
}
```

```
</script>
```

```
</body>
```

```
</html>
```

```
descriptionMedoc.php
```

```
<!DOCTYPE html>
```

```
<html lang="fr">
```

```
<head>
```

```
    <!-- Déclaration du jeu de caractères et de la vue pour les appareils mobiles -->
```

```
    <meta charset="UTF-8">
```

```
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
    <!-- Titre de la page -->
```

```

<title>Détails du médicament</title>

<!-- Lien vers la feuille de style CSS -->

<link rel="stylesheet" href="styles.css">

</head>

<body>

    <!-- Inclusion du composant d'en-tête -->

    <?php include "../Vue/components/header.component.html"; ?>

    <!-- Contenu principal de la page -->

    <div style="min-height: 100vh">

        <!-- Vérification de la présence de détails de médicament -->

        <?php if (!empty($details)): ?>

            <!-- Affichage des détails sous forme de tableau -->

            <table class="table">

                <tr>

                    <th>Description</th>

                </tr>

                <!-- Boucle pour afficher chaque détail du médicament -->

                <?php foreach ($details as $medicament): ?>

                    <tr>

                        <td><?php echo $medicament["description"]; ?></td>

                    </tr>

                <?php endforeach; ?>

            </table>

            <?php else: ?>

                <!-- Message affiché si aucun détail n'est trouvé pour le médicament -->

```

```

        <p>Aucun détail trouvé pour ce médicament.</p>

    <?php endif; ?>

</div>

<!-- Inclusion du composant de pied de page -->

<?php include "../Vue/components/footer.component.html"; ?>

</body>

</html>

conference.php

<!DOCTYPE html>

<html lang="fr">

<head>

    <!-- Déclaration du jeu de caractères -->

    <meta charset="utf-8">

    <!-- Inclusion du Bootstrap CSS -->

    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css"
rel="stylesheet">

    <!-- Titre de la page -->

    <title>Conférence</title>

</head>

<body class="bg-light">

    <!-- Inclusion du composant d'en-tête -->

    <?php include "../Vue/components/header.component.html"; ?>

    <!-- Contenu principal de la page -->

    <div class="container mt-5" style="min-height: 100vh">

```



```

<!-- Titre de la liste des conférences -->

<h1>Liste des Conférence</h1><br />

<!-- Démarrage de la session PHP -->

<?php session_start(); ?>

<!-- Affichage du message de confirmation de la conférence -->

<?php if(isset($_SESSION['messageConf'])): ?>

    <!-- Affichage du message avec la div de classe correspondante -->

    <?php echo $_SESSION['divMessage']?><?php echo $_SESSION['messageConf'];
?></div>

    <!-- Suppression de la variable de session après affichage -->

    <?php unset($_SESSION['messageConf']); ?>

<?php endif; ?>

<!-- Tableau pour afficher les conférences -->

<table class="table">

    <tbody>

        <table class="table table-bordered table-striped">

            <!-- En-tête du tableau -->

            <thead>

                <tr>

                    <th>ID</th>

                    <th>Nom</th>

                    <th>Description</th>

                    <th>Date</th>

                </tr>

            </thead>

            <!-- Corps du tableau -->

            <tbody>

```

```

<!-- Boucle pour afficher chaque conférence -->

<?php foreach ($conferences as $act): ?>

    <tr>

        <!-- Affichage des détails de la conférence -->

        <td><?php echo $act["id"]; ?></td>

        <td><?php echo $act["nom"]; ?></td>

        <td><?php echo $act["description"]; ?></td>

        <td><?php echo $act["date_activité"]; ?></td>

        <td>

            <div class="form-group">

                <!-- Formulaire pour rejoindre une conférence -->

                <form method="post" action="index.php?action=REJ">

                    <!-- Sélecteur pour choisir un utilisateur -->

                    <select class="form-control" id="user-select"
name="user-select">

                        <option value="">Choisir un utilisateur</option>

                        <!-- Boucle pour afficher chaque utilisateur -->

                        <?php foreach ($utilisateurs as $user): ?>

                            <option value="<?php echo $user["id"]; ?>">

                                <?php echo $user["nom"]; ?>

                            </option>

                        <?php endforeach; ?>

                    </select>

                    <br />

                    <!-- Champ caché pour l'ID de l'utilisateur -->

                    <input type="hidden" name="id_user" value="<?php echo
$user["id"]; ?>">

```

```

        <!-- Champ caché pour l'ID de la conférence -->
        <input type="hidden" name ="id_act" value="<?php echo
$act["id"]; ?>">

        <!-- Bouton pour soumettre le formulaire -->

        <button type="submit" name="rejoindre" value="Details"
class="btn btn-primary">Rejoindre</button>

    </form>

</div>

</td>

</tr>

<?php endforeach; ?>

</tbody>

</table>

</tbody>

</table><br /><br />

</div>

<!-- Inclusion du composant de pied de page -->

<?php include "../Vue/components/footer.component.html"; ?>

</body>

```

```

</html>

```

[DetailsMedicaments.php](#)

```

<!DOCTYPE html>

```

```

<html lang="fr">

```

```

<head>

```

```

    <!-- Déclaration du jeu de caractères -->

```

```

    <meta charset="utf-8">

```

```

<!-- Inclusion du Bootstrap CSS -->

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css"
rel="stylesheet">

<!-- Titre de la page -->

<title>Liste des médicaments de GSB</title>

</head>

```

```

<body class="bg-light">

<!-- Inclusion du composant d'en-tête -->

<?php include "../Vue/components/header.component.html"; ?>

<!-- Contenu principal de la page -->

<div class="container mt-5" style="min-height: 100vh">

    <!-- Titre de la liste des médicaments -->

    <h1>Liste des médicaments de GSB</h1><br />

    <!-- Tableau pour afficher les médicaments -->

    <table class="table">

        <tbody>

            <table class="table table-bordered table-striped">

                <!-- En-tête du tableau -->

                <thead>

                    <tr>

                        <th>ID</th>

                        <th>Nom</th>

                        <th>Actions</th>

                    </tr>

                </thead>

                <!-- Corps du tableau -->

```

```

<tbody>

    <!-- Boucle pour afficher chaque médicament -->

    <?php foreach ($medicaments as $medoc): ?>

        <tr>

            <!-- Affichage de l'ID du médicament -->

            <td><?php echo $medoc["id"]; ?></td>

            <!-- Affichage du nom du médicament -->

            <td><?php echo $medoc["nom"]; ?></td>

            <!-- Formulaire pour afficher les détails du médicament -->

            <td>

                <form method="post" action="index.php?action=DET">

                    <!-- Champ caché pour transmettre l'ID du médicament -->

                    <input type="hidden" name="id_medicament" value="<?php echo
$medoc["id"]; ?>">

                    <!-- Bouton pour soumettre le formulaire -->

                    <button type="submit" name="details" value="Details" class="btn
btn-primary">Détails</button>

                </form>

            </td>

        </tr>

    <?php endforeach; ?>

</tbody>

</table>

</tbody>

</table><br /><br />

</div>

<!-- Inclusion du composant de pied de page -->

```

```

    <?php include "../Vue/components/footer.component.html"; ?>

</body>

</html>

CharteGraphique.html

<!DOCTYPE html>

<html lang="fr">

<head>

    <!-- Déclaration du jeu de caractères -->

    <meta charset="UTF-8">

    <!-- Définition de la taille de la vue pour les appareils mobiles -->

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <!-- Titre de la page -->

    <title>Charte Graphique</title>

    <!-- Inclusion du Bootstrap CSS -->

    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css"
rel="stylesheet">

    <!-- Styles CSS personnalisés -->

    <style>

        /* Styles pour le corps de la page */

        body {

            font-family: 'Arial', sans-serif;

            background-color: #f4f4f4;

            color: #333;

```

```

margin: 0;

padding: 0;
}

/* Styles pour l'en-tête */

header {

background-color: #007BFF;

color: #fff;

text-align: center;

padding: 1em;
}

/* Styles pour les sections principales */

section {

max-width: 800px;

margin: 2em auto;

padding: 1em;

background-color: #fff;

box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
}

/* Styles pour les titres de section */

h1, h2, h3 {

color: #007BFF;
}

```

```
/* Styles pour les liens */  
  
a {  
    color: #007BFF;  
    text-decoration: none;  
}  
  
/* Styles pour les liens au survol */  
  
a:hover {  
    text-decoration: underline;  
}  
  
/* Styles pour le pied de page */  
  
footer {  
    text-align: center;  
    padding: 1em;  
    background-color: #007BFF;  
    color: #fff;  
    position: fixed;  
    bottom: 0;  
    width: 100%;  
}  
  
</style>  
  
</head>  
  
<body>
```



<!-- En-tête de la page -->

<header class="bg-primary text-white">

<h1>Titre de la Page</h1>

<p>Sous-titre ou description</p>

</header>

<!-- Section pour afficher l'original logo -->

<section class="container">

<h2>Original Logo</h2>

<p>Publiée le 12/03/2024</p>

<!-- Insertion des images du logo -->









</section>

<!-- Section pour afficher les couleurs d'image visuelle -->

<section class="container">

<h2>Visual image colours</h2>

<p>Publiée le 12/03/24</p>

<!-- Lien vers quelque chose d'autre -->

<a href="#">Lien vers quelque chose d'autre</a>

</section>

<!-- Section pour afficher la typographie -->

```
<section class="container">

  <h2>Typographie</h2>

  <p>Publiée le 12/03/24</p>

  <!-- Lien vers quelque chose d'autre -->

  <a href="#">Lien vers quelque chose d'autre</a>

</section>
```

```
<!-- Pied de page -->

<footer class="bg-primary text-white">

  <p>&copy; 2024 Nom de votre entreprise. Tous droits réservés.</p>

</footer>
```

```
<!-- Inclusion des scripts Bootstrap JS et jQuery (optionnel) -->

<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>

<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>
```

```
</body>
```

```
</html>
```

[header.component.html](#)

```
<!-- Inclusion du fichier CSS de Bootstrap -->

<link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
```

```
<!-- Barre de navigation Bootstrap -->

<nav class="navbar navbar-expand-lg navbar-dark bg-dark">
```

```

<!-- Logo et lien de la marque -->

<a class="navbar-brand" href="index">

    Galaxy Swiss Bourdin

</a>


<!-- Bouton de basculement pour les petits écrans -->

<button class="navbar-toggler" type="button" data-toggle="collapse"
data-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false"
aria-label="Toggle navigation">

    <span class="navbar-toggler-icon"></span>

</button>


<!-- Conteneur pour les éléments de la barre de navigation -->

<div class="collapse navbar-collapse" id="navbarNav">

    <!-- Liste des liens de navigation -->

    <ul class="navbar-nav mr-auto">

        <li class="nav-item">

            <a class="nav-link"
href="/phpgroupe/index.php?action=CO">Conférence</a>

        </li>

        <li class="nav-item">

            <a class="nav-link" href="/phpgroupe/index.php?action=JU">Mentions
légales</a>

```

```

        </li>

</ul>

<!-- Texte de droite de la barre de navigation -->

<div class="navbar-text text-right">

    <a href="index?action=LO" style="padding-left:10px;">Se déconnecter</a>

</div>

</div>

</nav>

```

### footer.component.html

```

<!-- Inclusion du fichier CSS de Bootstrap -->

<link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.
min.css">

<!-- Style CSS pour le pied de page -->

<style>

    footer {

        position: relative; /* Position relative pour pouvoir utiliser
le bas comme référence */

        bottom: 0; /* Alignement en bas de la page */

        width: 100%; /* Largeur pleine de l'écran */

    }</style>

<!-- Pied de page Bootstrap -->

<footer class="py-2 bg-dark text-white">

    <div class="container text-center">

        <p class="m-0" style="padding-bottom: 20px;">© 2024 Galaxy
Swiss Bourdin</p>

    </div>

</footer>

```

