

GSBStock

ITH Kagnana
BTS SIO 2



Février 2024 - Mars 2024

SOMMAIRE

01 Contexte “Laboratoire GSB”

- A.** Description du laboratoire
- B.** Domaine d'étude

02 Les objectifs et les besoins

- A.** Professionnel
- B.** Personnel

03 Analyse fonctionnelle

- A.** Conception
- B.** Déploiement
- C.** Tester
- D.** Présentation de l'application

04 Annexes

CONTEXTE GSB

Description du laboratoire

Le secteur d'activité

L'industrie pharmaceutique est un secteur très lucratif dans lequel le mouvement de fusion acquisition est très fort. Les regroupements de laboratoires ces dernières années ont donné naissance à des entités gigantesques au sein desquelles le travail est longtemps resté organisé selon les anciennes structures.

Des déboires divers récents autour de médicaments ou molécules ayant entraîné des complications médicales ont fait s'élever des voix contre une partie de l'activité des laboratoires : la visite médicale, réputée être le lieu d'arrangements entre l'industrie et les praticiens, et tout du moins un terrain d'influence opaque.

L'entreprise

Le laboratoire Galaxy Swiss Bourdin (GSB) est issu de la fusion entre le géant américain Galaxy (spécialisé dans le secteur des maladies virales dont le SIDA et les hépatites) et le conglomérat européen Swiss Bourdin (travaillant sur des médicaments plus conventionnels), lui même déjà union de trois petits laboratoires.

En 2009, les deux géants pharmaceutiques unissent leurs forces pour créer un leader de ce secteur industriel. L'entité Galaxy Swiss Bourdin Europe a établi son siège administratif à Paris. Le siège social de la multinationale est situé à Philadelphie, Pennsylvanie, aux Etats-Unis.

Domaine d'étude

L'entreprise souhaite porter une attention nouvelle à sa force commerciale dans un double objectif : obtenir une vision plus régulière et efficace de l'activité menée sur le terrain auprès des praticiens, mais aussi redonner confiance aux équipes malmenées par les fusions récentes.

La force commerciale d'un laboratoire pharmaceutique est assurée par un travail de conseil et d'information auprès des prescripteurs. Les visiteurs médicaux (ou délégués) démarchent les médecins, pharmaciens, infirmières et autres métiers de santé susceptibles de prescrire aux patients les produits du laboratoire.

L'objectif d'une visite est d'actualiser et rafraîchir la connaissance des professionnels de santé sur les produits de l'entreprise. Les visiteurs ne font pas de vente, mais leurs interventions ont un impact certain sur la prescription de la pharmacopée du laboratoire.

LES OBJECTIFS & BESOINS

Professionnels

L'objectif de ce projet est d'apporter une solution de gestion de stocks aux employés de GSB.

La gestion de stocks se fera par le biais de commande faites par les employés ou par les clients de l'entreprise.

On prendra également en compte les différents niveaux d'accès des produits, permettant ainsi de limiter l'accès à certains éléments dangereux ou rares.

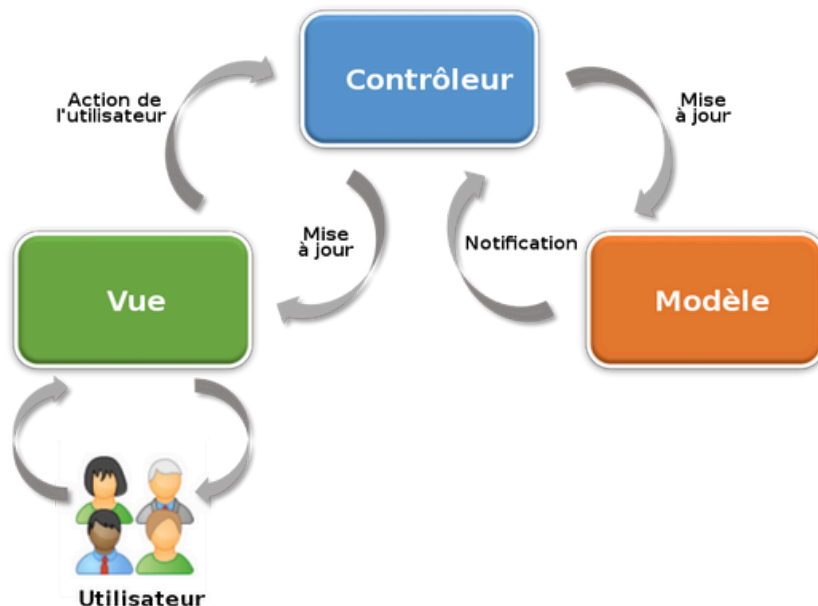
Personnel

Ce projet a également pour but personnel de :

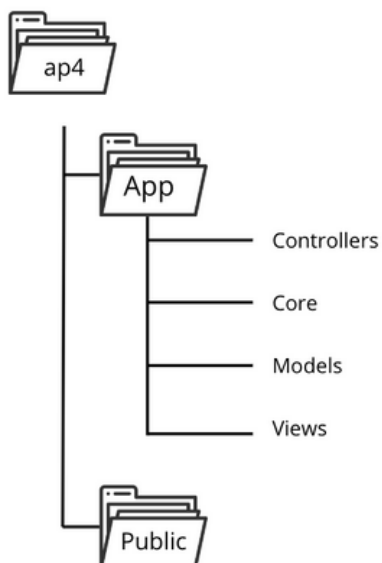
- Découvrir PHP
- D'approfondir mes connaissances en base de données relationnelle (MySQL)
- D'approfondir mes connaissances sur Docker
- Déployer le projet sur un serveur distant

ANALYSE FONCTIONNELLE

Conception – Architecture



Dans ce projet PHP, nous avons décidé d'utiliser l'architecture MVC pour organiser nos fichiers



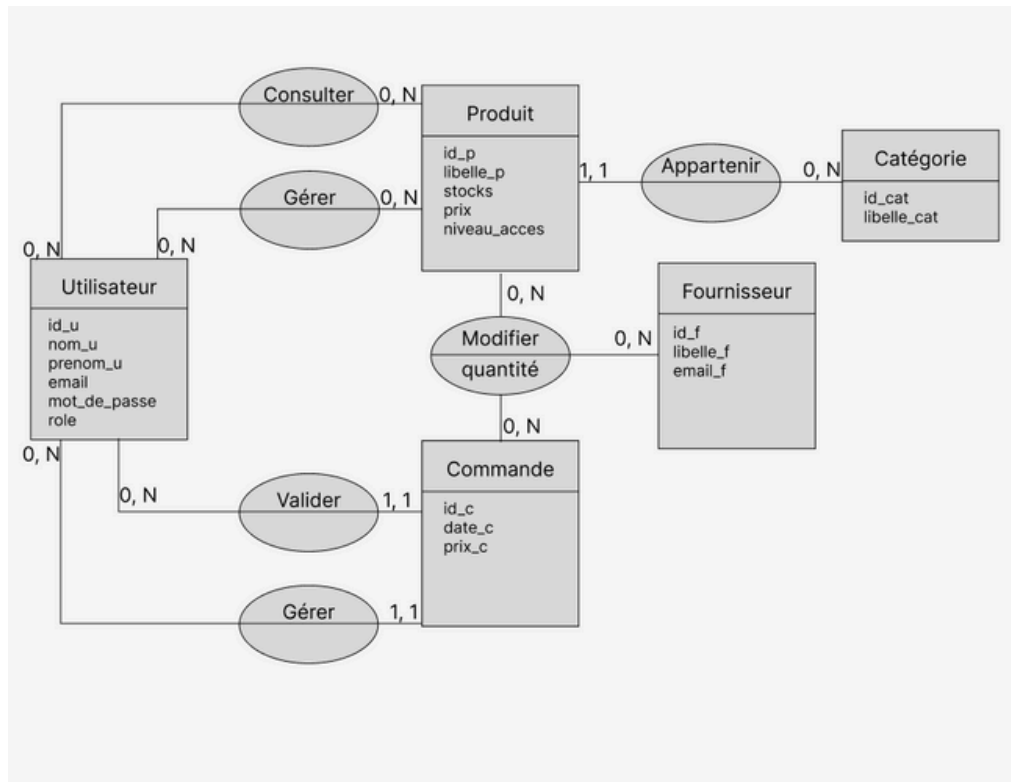
Le contrôleur permet de gérer l'action faite par l'utilisateur et de rediriger l'utilisateur vers la bonne page.

Le contrôleur est en contact avec le modèle qui va gérer les requêtes. Il récupère les erreurs que le modèle renvoie

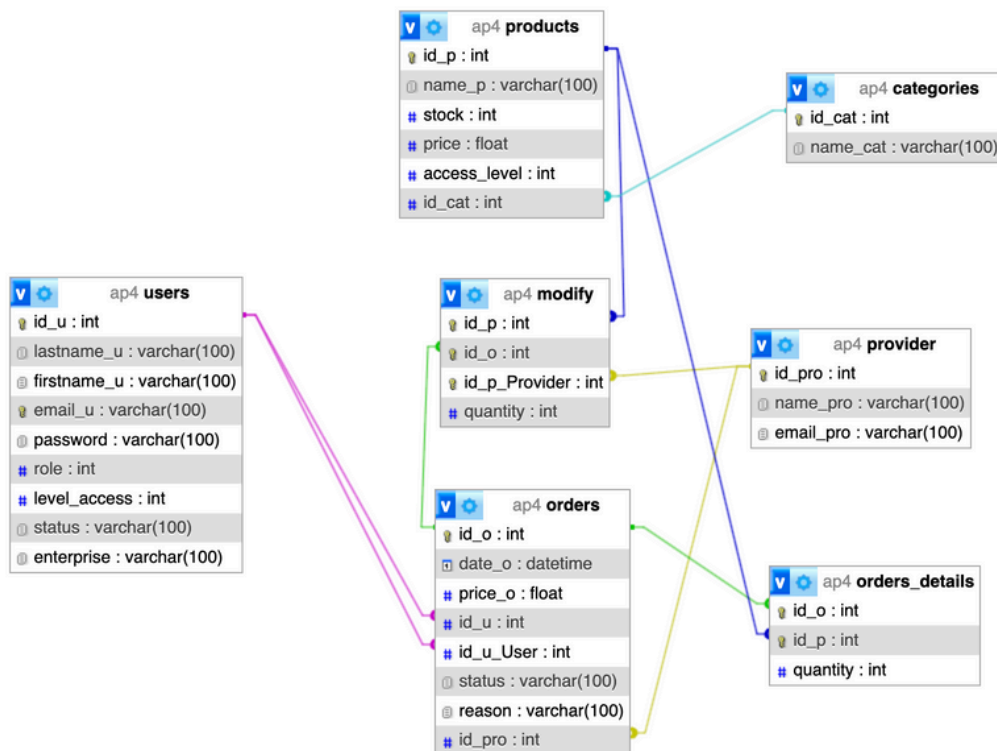
Le dossier core permet de regrouper les fichiers qui ne sont pas classables mais qui ont une utilité dans l'application.

Par exemple on va y retrouver les classes qui vont être héritées par les contrôleurs ou les modèles.

Conception – Base de donnée



MCD DE NOTRE APPLICATION



TABLES ET RELATIONS

Conception – Base de donnée

Modifications apportées à la base de données

Lors du développement de l'application, certains changements sont apportés à la base pour correspondre aux besoins du projet.

Le niveau d'accès a été ajouté pour réguler les différents niveaux de produits que les utilisateurs peuvent commander.

```
ALTER TABLE users  
ADD COLUMN level_access INT DEFAULT 1
```

On ajoute la colonne status dans la table user pour permettre le cas où l'utilisateur souhaite créer lui-même un compte. Par mesure de sécurité, on attends que l'administrateur valide le compte avant de pouvoir permettre à l'utilisateur de se connecter.

```
ALTER TABLE users  
ADD COLUMN status VARCHAR(100) DEFAULT "En attente de validation"
```

Dans notre application nous aimerions aussi créer des comptes clients. On veut alors préciser de quelle entreprise ils viennent

```
ALTER TABLE users  
ADD COLUMN entreprise VARCHAR(100)
```

Conception – Base de donnée

Trigger

Afin de pouvoir automatiquement modifier les stocks après la validation d'une commande, on va créer un trigger

```
DELIMITER $$
CREATE TRIGGER UpdateProductStockTrigger AFTER UPDATE ON
orders
FOR EACH ROW
BEGIN
  IF NEW.status = "Validé" THEN
    UPDATE products
    JOIN orders_details ON orders_details.id_o = NEW.id_o
    SET stock = orders_details.quantity + products.stock
    WHERE products.id_p = orders_details.id_p;
  END IF;
END
$$
DELIMITER
```

Ce trigger s'exécute après la modification d'une commande uniquement si son status a été changé en validé.

On va alors mettre à jour la table **products** en changeant pour chaque produit appartenant à la commande le stock.

Dans le cas d'une commande entrante ou sortante, on fait la même chose car la colonne *quantity* dans la table **orders_details** peut contenir des entiers négatifs

Les stocks sont alors modifiés uniquement après la validation de la commande. Si la commande est en attente de traitement ou refusée, les stocks ne changeront pas.

Conception – Navigation

Fonctionnement

La navigation est gérée par l'App.php dans le dossier core. Ce dernier va décomposer les différents éléments de l'url et utiliser le contrôleur associé.

Méthode qui permet de décomposer l'url en un tableau.

```
41      /** function that get the url and separed the different section in url
42      */
43      1 reference | 0 overrides
44      public function parseUrl()
45      {
46          if (isset ($_GET["url"])) {
47              return explode("/", $_GET["url"]); # return an array of string
48          } else {
49              return "login";
50          }
51      }
```

Méthode qui exécute la fonction du contrôleur renseignée par l'url.

```
12      {
13          $url = $this->parseUrl();
14
15          # get controller
16          if (file_exists("../app/controllers/" . ucfirst($url[0]) . "Controller.php")) {
17              $this->controller = ucfirst($url[0]) . "Controller";
18
19              if (!$this->checkRole($this->controller)) {
20                  $this->controller = "ErrorController";
21              }
22
23              unset($url[0]);
24          }
25
26          # set controller instance
27          $this->controller = new $this->controller;
28
29          # to process if multiple views or request are handle by the controller
30          if (isset ($url[1]) && $url[1] != "o") {
31              if (method_exists($this->controller, $url[1])) {
32                  $this->method = $url[1];
33                  $this->controller->{$this->method}();
34                  unset($url[1]);
35              }
36          } else {
37              $this->controller->index();
38          }
```

Conception – Navigation

Changement dans la vue

Plusieurs vues dans l'application ont une possibilité de recherche et de filtrage.

Lorsque l'on exécute la fonction pour filtrer, on apporte des modifications dans la vue principale.

On peut également prendre le cas après la création ou la modification d'un élément qui nous ramène vers la page principale de cet élément.

Par exemple, lorsque l'on valide une commande, nous sommes redirigés vers la page avec la liste des commandes.

Ces opérations peuvent entraîner des messages d'erreurs.

Pour dynamiser le code, les fonctions qui servent à afficher les éléments d'une vue, peuvent recevoir un tableau **\$extra**. Ce tableau contient les éléments que notre vue peut traiter.

Dans controllers/OrderController.php

```
21 public function index($extra = null)
22 {
23     if (isset ($extra["filter"]) && $extra["filter"] != "all") {
24         $res = $_SESSION["userRole"] < 2 ? $this->order->getFilteredOrder($extra["filter"], $extra["searchName"] ?? null) :
25             $this->order->getFilteredUserOrder($extra["filter"], $_SESSION["userId"]);
26     } else {
27         $res = $_SESSION["userRole"] < 2 ? $this->order->getOrder($extra["searchName"] ?? null) : $this->order->getUserOrders($_SESSION["userId"]);
28     }
29     $filter = null;
30     $error = null;
31
32     $orders = $res["data"];
33     $error = $res["error"] ?? null;
34
35     if (isset ($extra["error"])) {
36         $error = $res["error"];
37     }
38
39     $data = ["all" => $orders, "error" => $error, "filter" => $filter];
40
41     if (isset ($extra)) {
42         $sendData = array_merge($data, $extra);
43         $this->view("order/orders-view", $sendData);
44     } else {
45         $this->view("order/orders-view", $data);
46     }
47 }
```

On remarque que l'on va traiter les différentes variables que peut contenir **\$extra**.

Première ligne on vérifie qu'il existe une valeur dans **\$extra** avec la clé "filter" et si elle est différent de "all" pour récupérer les commandes.

On vérifie également s'il existe une valeur dans **\$extra** avec la clé "error". Cette variable est utilisée si notre opération d'ajout ou de modification a entraîné une erreur.

Conception – Sécurité

Hachage de mot de passe

Il est important de hacher le mot de passe pour éviter qu'il soit visible en clair dans la base de donnée.

On va utiliser la fonction `password_hash()` de php.

```
function createUser()
{
    try {
        $this->db->query("INSERT INTO users (lastname_u, firstname_u, email_u, password, role, status, enterprise)
VALUES (:lastname, :firstname, :email, :password, :role, :status, :client_name)");
        $this->db->bind("lastname", $this->lastname);
        $this->db->bind("firstname", $this->firstname);
        $this->db->bind("email", $this->email);
        $this->db->bind("role", $this->role);
        $hashedPassword = password_hash($this->password, PASSWORD_DEFAULT);
        $this->db->bind("password", $hashedPassword);
        $this->db->bind("client_name", $this->enterprise ?? "");
        $this->db->bind("status", $this->status);
        $this->db->fetch();
    } catch (Exception $e) {
        return ["error" => "Impossible d'ajouter un utilisateur pour le moment."];
    }
}
```

Pour permettre à l'utilisateur de se connecter avec le mot de passe même s'il est haché en base, on va utiliser la fonction `password_verify()` qui va vérifier que le password donné par l'utilisateur correspond à celui en base.

```
public function login($email, $password)
{
    try {
        $this->db->query("SELECT * FROM users WHERE email_u=:email");
        $this->db->bind("email", $email);
        $result = $this->db->fetch();
    } catch (Exception $e) {
        return ["data" => false, "error" => "Une erreur est survenue lors de la connexion."];
    }

    # prevent user to login when his status is not valid
    if ($result["status"] == "En attente de validation") {
        return ["data" => false, "error" => "L'administrateur n'a pas encore validé votre compte."];
    } else if ($result["status"] == "Refusé") {
        return ["data" => false, "error" => "L'administrateur a refusé votre demande"];
    }

    if (count($result) != 0 && password_verify($password, $result[4])) {
        $user = new User();
        $user->setUser(
            $result["enterprise"],
            $result["lastname_u"],
            $result["firstname_u"],
            $result["email_u"],
            "", # place for the password
            $result["role"],
            $result["level_access"],
            $result["status"],
            $result["id_u"],
        );
        $user->saveInSession();
        return ["data" => true];
    } else {
        return ["data" => false, "error" => "Vos identifiants sont invalides."];
    }
}
```

Conception – Sécurité

Gestion d'erreur

Il est important de gérer les erreurs que peuvent nous renvoyer notre SGBD lorsqu'une requête a échoué. Ainsi nous pouvons prévenir les utilisateurs que l'opération n'a pas abouti.

Toutes les fonctions vont retourner :

- une **donnée** si nécessaire
- une **erreur** si elle existe

Dans un bloc try/catch, nous allons exécuter la requête et si il y a une erreur, la récupérer grâce au catch.

Nous allons ensuite retourner une donnée et/ou une erreur. Ainsi on pourra afficher la réponse à l'utilisateur

```
/** Function to get all categories
 * @return [] categories
 */
0 references | 0 overrides
public function getCategories()
{
    try {
        $this->db->query("SELECT * FROM categories");
        $result = $this->db->fetchAll();
    } catch (PDOException $exception) {
        return ["data" => [], "error" => "Les catégories n'ont pas pu être récupérées."];
    }

    return ["data" => $result];
}
```

Ensuite dans le ProductController

```
57     $res = $this->product->getCategories();
58     $error = $res["error"] ?? null;
59
60     if (isset ($extra["error"])) {
61         $error = $extra["error"];
62     }
63
64     # display form to add product
65     $this->view("product/form-product-view", [
66         "allCat" => $res["data"],
67         "error" => $error ?? null,
68         "form" => $newProduct ?? null,
69     ]);
70 }
```

S'il y a une erreur lors de la requête.

Accueil	Stocks	Commande	Utilisateurs	Fournisseurs	Déconnexion
---------	--------	----------	--------------	--------------	-------------

Filtres :











Rechercher

Catégorie

Tous

Réinitialiser les filtres

Les catégories n'ont pas pu être récupérées.

id	Libellé	Prix à l'unité (€)	Stock	Niveau d'accès	Catégorie	Action
2	Blouse blanche	12.5	50	1	Combinaison	 
3	Molécule ARM	102	1	3	Composant médicamenteux	 
4	Fliale 25ml	3.8	98	1	Verrerie	 
5	Masque en papier (lot de 20)	10.5	196	1	Combinaison	 
8	Pipette	3.8	346	1	Matériel de recherche	 

Pour la suppression de certains éléments on va vérifier que l'erreur n'est pas dû au fait qu'il est utilisé dans une autre table. Pour cela, on utilise une enum dans core/Database.php et on va récupérer le type d'erreur en fonction du code.

```
/** Function to return error type if needed
 * @param ?string $code
 */
3 references | 0 overrides
public function getTypeError($code)
{
    return match ($code) {
        "23000" => ErrorType::IsFK,
        default => ErrorType::UnkownError
    };
}
```

Dans models/Product.php

```
public function deleteProduct($id)
{
    try {
        $this->db->query("DELETE FROM products WHERE id_p = :id");
        $this->db->bind("id", $id);
        $this->db->fetch();
    } catch (PDOException $exception) {
        $errorType = $this->db->getTypeError($exception->getCode());
        $errorMsg = $errorType == ErrorType::IsFK ? "Le produit est associé à une commande. Il ne peut pas être supprimé." :
            "Le produit n'a pas pu être supprimé";
        return ["error" => $errorMsg];
    }
}
```

Conception – Sécurité

Gérer les entrées utilisateurs

Pour éviter les erreurs en base et pour s'assurer que les données fournies sont bien celles que l'on attend on va vérifier que l'entrée correspond au type demandé et n'est pas vide

Par exemple, quand on veut ajouter un fournisseur on vérifie que le champ de nom et d'email ne sont pas vides mais également que l'email correspond au format classique d'un email grâce à un regex.

```
public function setProvider($name, $email, $id = null)
{
    $this->name = $name;
    $this->email = $email;
    $this->id = $id;
    if (!isset ($name) || !isset ($email) || $name == "" || $email == "") {
        return ["error" => "Certains champs sont vides."];
    }

    if (!preg_match("/[\w\-\.\.]+\@([\w-]+\.\.)+[\w-]{2,4}/", $email)) {
        return ["error" => "L'email est invalide."];
    }
}
```

On impose également à l'utilisateur d'avoir un mot de passe un minimum sécurisé. Le mot de passe doit contenir au minimum :

- Une majuscule
- Une minuscule
- Un chiffre
- Un caractère spécial
- Une longueur de 8 caractères

Dans ce cas là, on utilise également un Regex pour vérifier que toute les conditions sont remplies.

Dans controllers/UserController.pph

```
36 if (isset ($_POST["origin"]) && $_POST["origin"] == "user") {
37     $check = $newUser->setUser($enterprise, $lastname, $firstname, $email, $password, $role);
38     if (isset ($check["error"])) {
39         $this->view("user/create-account-view", ["error" => $check["error"], "form" => $newUser]);
40         return;
41     }
42     if (!preg_match("/(?!.*[a-z])(?!.*[A-Z])(?!.*\d)(?!.*[!@#$%^&*]) [A-Za-z\d@$!%*?&]{8,}/", $password)) {
43         $error = "Le mot de passe doit contenir au minimum une majuscule, une minuscule, un chiffre et un spécial caractère.
44         Il doit également avoir une longueur minimum de 8 caractères.";
45         $this->view("user/create-account-view", ["error" => $error, "form" => $newUser]);
46         return;
47     }
48     $res = $newUser->createUser();
49     $this->view("user/login-view", ["error" => $res["error"] ?? null]);
```

Affichage de l'erreur dans le formulaire.

Crée un compte

Et profitez des fonctionnalités de notre site

Le mot de passe doit contenir au minimum une majuscule, une minuscule, un chiffre et un spécial caractère. Il doit également avoir une longueur minimum de 8 caractères.

Votre status

☒ Employé ☐ Client

Nom de l'entreprise (Client uniquement)

Nom

Prénom

Email

Mot de passe

Valider

← Retour

Corbeille

Contrôler les accès aux pages et fonctionnalités

On souhaite également empêcher les utilisateurs n'ayant pas les accès d'accéder à des pages et à certaines fonctions de notre application.

Dans l'App.php qui contrôle la navigation, nous allons vérifier que les utilisateurs ont les accès à la page qu'ils souhaitent accéder.

```
52  /** private func to check if user has access to the page
53  * @param string $controller
54  * @return bool if user has access
55  */
56  private function checkRole($controller)
57  {
58      $role = $_SESSION["userRole"] ?? null;
59
60      if (!isset ($role)) {
61          return match ($controller) {
62              "LoginController" => true,
63              "UserController" => true, #check will be done inside controller
64              "HomeController" => true, #check will be done inside controller
65              default => false,
66          };
67      } else {
68          return match ($controller) {
69              "ProviderController" => $role == 0 ? true : false,
70              "ProductController" => $role < 2 ? true : false,
71              "UserController" => true, #check will be done inside controller
72              "OrderController" => true, #check will be done inside controller
73              "LoginController" => true,
74              "HomeController" => true,
75          };
76      }
```

On vérifie si l'utilisateur est connecté et s'il a les droits pour accéder à la page. Si ce n'est pas le cas il sera redirigé.

Pour certaines pages, on ne limite uniquement que quelques fonctionnalités. On prend par exemple le cas où l'utilisateur a le rôle employé. Ce dernier peut consulter toutes les commandes mais il ne peut pas les contrôler.

De ce fait, nous allons vérifier que l'utilisateur a le bon rôle pour accéder à la page de contrôle mais également aux fonctions qui y sont liées.

Dans controllers/OrderController.php

```
98     public function control($extra = null)
99     {
100         if (!$this->checkAccess("admin")) {
101             $this->view("error/prohibited-view");
102             return; # force exit
103         }
```

Dans le cas où l'utilisateur essaie d'accéder à une page dont il n'a pas les autorisations, il sera redirigé vers une page d'erreur.



Déploiement

Configuration

Le projet est doté d'une configuration Docker qui permet de faciliter le lancement de l'application sur n'importe quelle machine possédant docker mais permet également de faciliter le déploiement.



Le fichier docker-compose.yml est un fichier de configuration pour les conteneurs utilisé dans mon projet

Le projet se compose en 3 conteneurs :

- web : avec une image php:8.2-apache
- db : avec une image mysql:8.0
- phpmyadmin

Explications de la configuration :

Le network permet à nos conteneurs de communiquer entre eux.

Les volumes permettent de stocker l'informations en mémoire de manière permanente.

Le Dockerfile est un fichier nécessaire à la création d'un conteneur, notamment celui de php.

Installer le projet localement

Dans ce projet, ajouter un .env à la racine. Ceci est un exemple de valeur utilisée dans l'application. Remplacez les champs SQL par ceux de votre choix et l'hôte correspondant.

HOST = http://localhost:8089

MYSQL_ROOT_PASSWORD = votreMotDePasseRoot

MYSQL_DATABASE = ap4

MYSQL_USER = votreUtilisateur

MYSQL_PASSWORD = votreMotDePasseUser

Faites attention si jamais vous changez le nom du fichier .env, veuillez également faire le changement dans le fichier docker-compose.yml dans les champs env_file.

Pour créer et lancer le container avec Docker :

```
docker-compose up -d
```

Si vous êtes en local, vous pouvez accéder à phpmyadmin :

<http://localhost:8080> .Sinon avec votre hôte au port 8080. Connectez vous avec les identifiants que vous avez renseigné dans votre dossier .env

Connectez vous de préférence avec le compte root pour faire l'importation du script `scrpt-create.sql` afin de pouvoir importer tous le contenu du script.

Maintenant vous pouvez tester l'application web sur le port 8089.

Aller à l'étape **Déploiement/Test** pour avoir le tableau des identifiants.

Étapes du déploiement sur un serveur distant

Afin de pouvoir accéder à l'application via l'adresse : <https://gsb-stocks.kagnana-ith.fr/login>

Sur cette machine, j'installe l'apt docker :

<https://docs.docker.com/engine/install/ubuntu/>

Ensuite je git clone mon projet.

```
git clone https://github.com/iKagnana/ap4-GSBStock.git
```

Pour build et lancer mes conteneurs, j'utilise la commande suivante :

```
sudo docker compose up -d
```

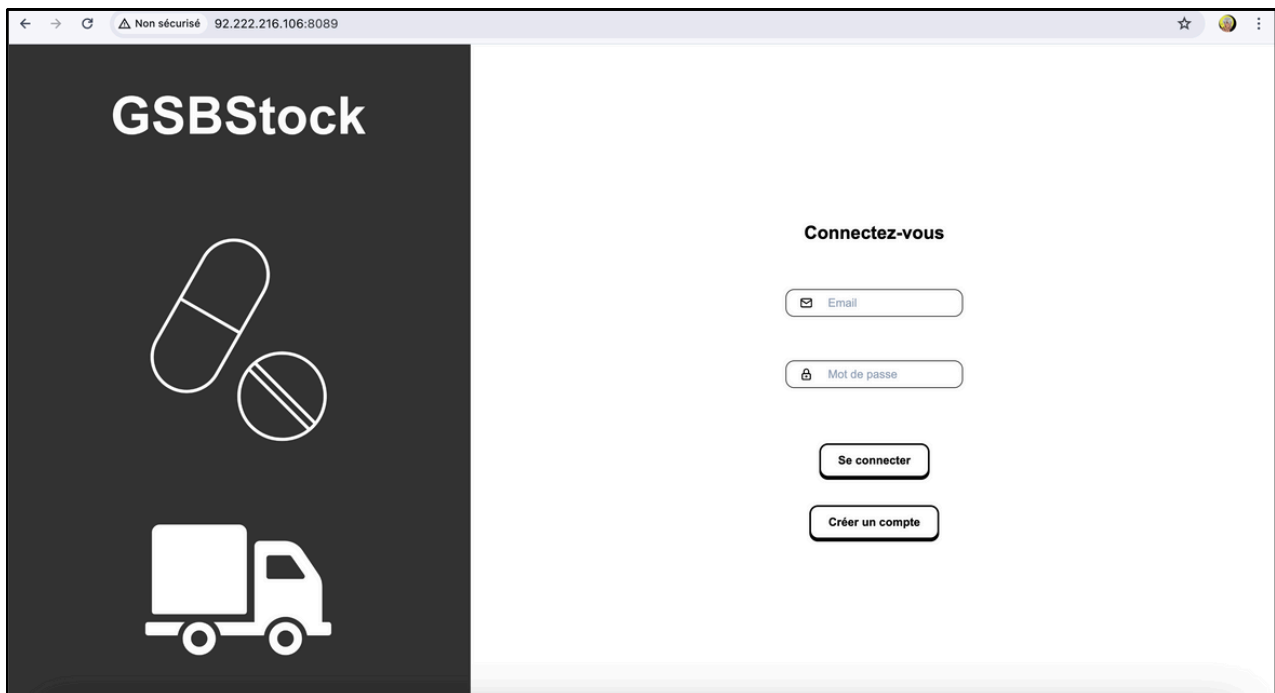
Cette commande permet de build, créer, lancer et attacher les conteneurs à un service

le flag `-d` permet d'exécuter les conteneurs en arrière-plan.

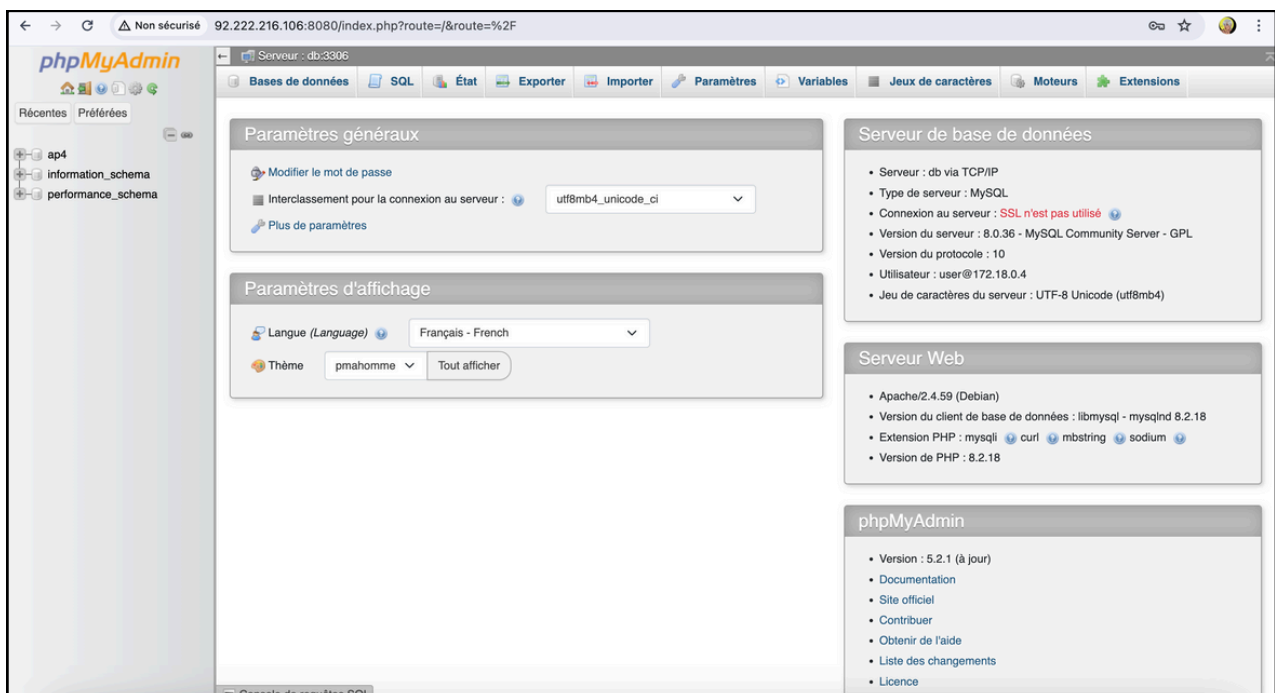
Ensuite on peut vérifier que nos conteneurs tourne avec la commande :

```
docker ps
```

Maintenant si on va sur l'adresse ip 92.222.216.106:8089



On peut également accéder à phpmyadmin pour importer notre dump de la base de donnée pour l'utiliser dans notre application.



Déploiement

Virtual Host

Afin de pouvoir accéder à notre application depuis l'adresse :

<https://gsb-stocks.kagnana-ith.fr/login>

Il faut qu'Apache agisse comme un reverse proxy pour rediriger vers notre conteneur.

```
<VirtualHost *:80>
```

```
ServerAdmin webmaster@localhost
```

```
ServerName GSB-Stocks.kagnana-ith.fr
```

```
DocumentRoot /var/www/gsb-stock.kagnana-ith/html/ap4-GSBStock
```

```
ProxyPreserveHost On
```

```
ProxyRequests Off
```

```
ProxyPass / http://localhost:8089/
```

```
ProxyPassReverse / http://localhost:8089/
```

```
ErrorLog ${APACHE_LOG_DIR}/error.log
```

```
CustomLog ${APACHE_LOG_DIR}/access.log combined
```

```
RewriteEngine on
```

```
RewriteCond %{SERVER_NAME} =GSB-Stocks.kagnana-ith.fr [OR]
```

```
RewriteCond %{SERVER_NAME} =gsb-stocks.kagnana-ith.fr
```

```
RewriteRule ^ https://%{SERVER_NAME}%{REQUEST_URI} [END,NE,R=permanent]
```

```
</VirtualHost>
```

Certificat SSL

Afin de pouvoir bénéficier d'une connexion sécurisée sur notre site, il nous faut un certificat SSL. Pour générer ce certificat, nous allons utiliser Certbot.

<https://certbot.eff.org/instructions?ws=apache&os=ubuntufocal>

Toutes les instructions sont disponibles sur le site de Certbot.

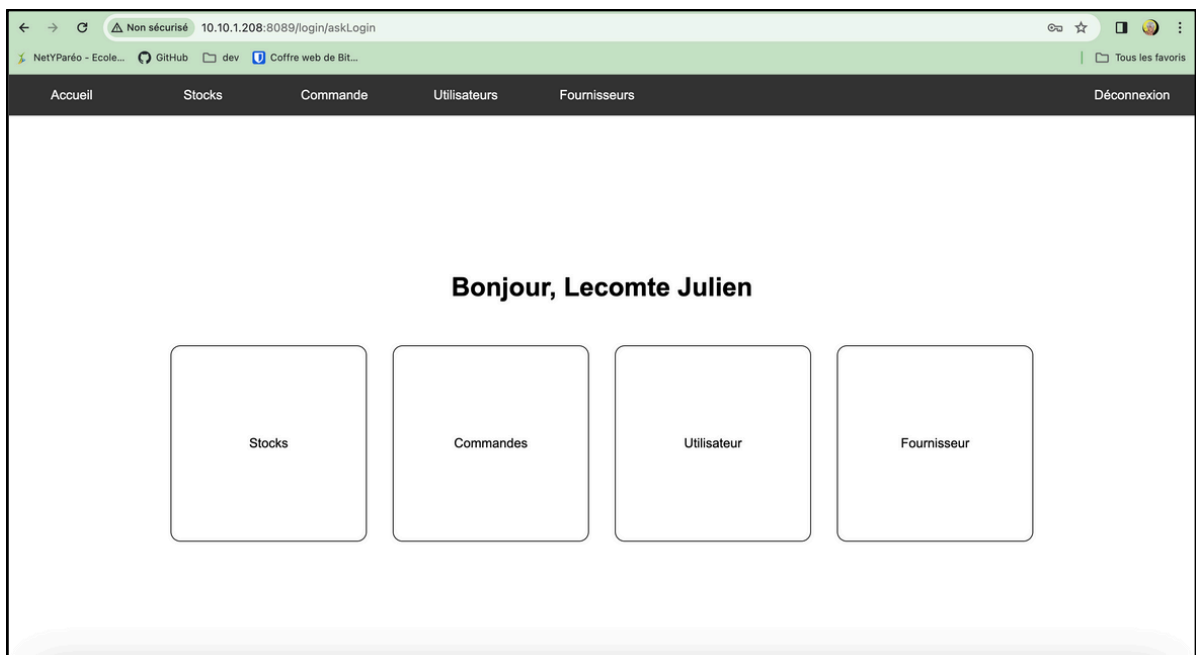
Déploiement

Test

Différents comptes pour tester

	Administrateur	Employé	Client
Email	lecomte.julien@gsb.com	b.tapis@gsb.com	d.jean@fidele.com
Mot de passe	leComteJ123!	bTapis12!	test

Et voilà !



Présentation de l'application

L'affichage des vues peut différer en fonction du rôle de l'utilisateur.

Par exemple, un client peut uniquement créer des commandes sortantes et consulter les siennes.

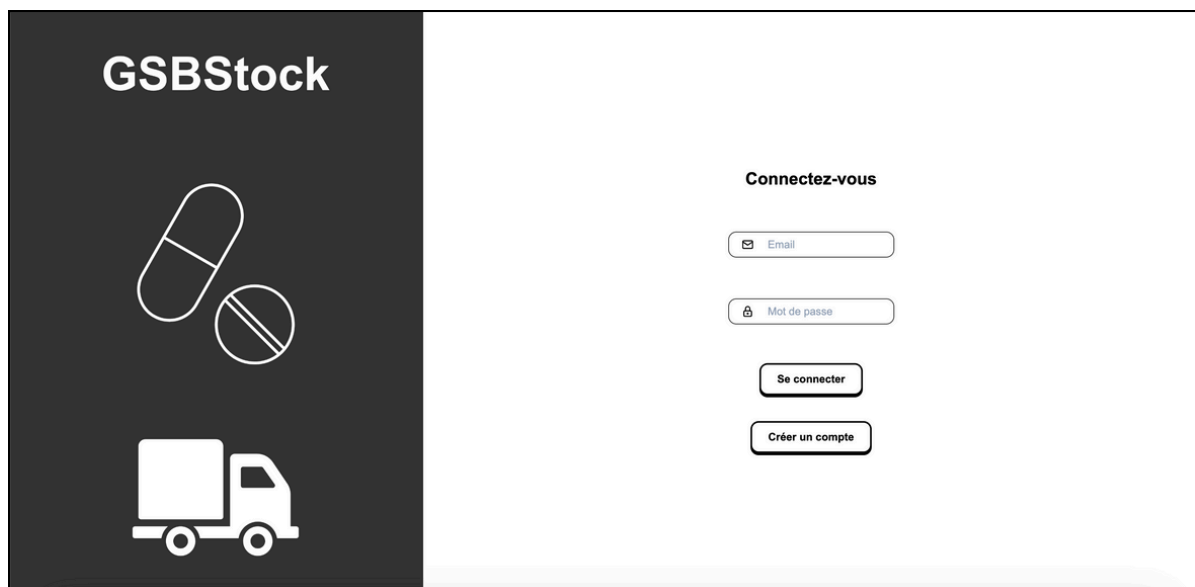
Un employé est aussi limité, ce dernier ne peut pas créer d'utilisateur ou encore contrôler les commandes.

Nous allons nous mettre à la place de l'administrateur pour montrer toutes les fonctionnalités et pages de l'application.

Voici les règles :

- administrateur : peut gérer les produits, commandes (entrantes et sortantes), utilisateurs et fournisseurs
- employé : peut gérer les produits, voir toutes les commandes et faire une commande (entrante ou sortante)
- client : ne peut voir que ses commandes et faire des commandes sortantes

Page de connexion



GSBStock

Connectez-vous

Email

Mot de passe

Se connecter

Créer un compte

Page de création de compte

Crée un compte

Et profitez des fonctionnalités de notre site

Votre status

☐ Employé ☐ Client

Nom de l'entreprise (Client uniquement)

Nom

Prénom

Email

Mot de passe

Valider

← Retour

Dans le cas où un utilisateur s'inscrit, il a besoin d'attendre la validation par l'administrateur pour pouvoir se connecter.

Si son compte est validé et que ses identifiants sont valides, il accède à la page d'accueil.

Accueil

Stocks

Commande

Utilisateurs

Fournisseurs

Déconnexion

Bonjour, ITH Kagnana

Stocks

Commandes

Utilisateur

Fournisseur

LA PARTIE PRODUIT

Accueil

Stocks

Commande

Utilisateurs

Fournisseurs

Déconnexion

Consulter

Ajouter

Filtres :

Rechercher

Catégorie

Tous

Réinitialiser les filtres

Id	Libellé	Prix à l'unité (€)	Stock	Niveau d'accès	Catégorie	Action
2	Blouse blanche	12.5	50	1	Combinaison	<div><div></div><div></div></div>
3	Molécule ARM	102	1	3	Composant médicamenteux	<div><div></div><div></div></div>
4	Fiole 25ml	3.8	98	1	Verrerie	<div><div></div><div></div></div>
5	Masque en papier (lot de 20)	10.5	196	1	Combinaison	<div><div></div><div></div></div>
8	Pipette	3.8	346	1	Matériel de recherche	<div><div></div><div></div></div>

10.10.1.208-8089heroduck

Deux actions possibles :

- Voir les détails
- Ajouter au panier

Détails d'un produit

Accueil

Stocks

Commande

Utilisateurs

Fournisseurs

Déconnexion

← Retour

Libellé

Blouse

Prix à l'unité €

12,5

Stock

50

Attention il n'est pas conseillé de modifier cette valeur.

Niveau d'accès

1

Catégorie

Matériel de recherche

Valider

Supprimer

Formulaire d'ajout de produit

Accueil

Stocks

Commande

Utilisateurs

Fournisseurs

Déconnexion

Libellé du produit

Prix à l'unité (€)

Stock

Niveau d'accès

Catégorie

Composant médicamenteux

Valider

Page accessible depuis le sous menu "Ajouter" de Stocks dans la barre de navigation

LA PARTIE COMMANDE

La liste des commandes n’affiche pas les mêmes données en fonction du rôle de l’utilisateur. Si ce dernier est un client, il ne verra alors que ses commandes.

Or les administrateurs et les employés ont accès à toutes les commandes faites.

Accueil

Stocks

Commande

Utilisateurs

Fournisseurs

Déconnexion

Consulter

Ajouter

Contrôler

Filtres :

Rechercher par demandeur

Status

Tous

Réinitialiser les filtres

Commandes

Id	Date	Prix total	Demandeur	Status	Raison	
1	2024-03-03 15:42:59	39€	ITH Kagnana	Validé		Détails
2	2024-03-05 21:20:36	48€	ITH Kagnana	Refusé	pas besoin	Détails
3	2024-03-05 22:07:21	162€	ITH Kagnana	Validé		Détails
4	2024-03-05 22:08:33	369€	ITH Kagnana	Validé	RAS	Détails
10	2024-03-18 10:39:32	37€	ITH Kagnana	Validé		Détails
11	2024-03-18 11:11:28	918€	ITH Kagnana	Validé		Détails
12	2024-03-18 11:11:48	204€	ITH Kagnana	En attente de validation		Détails
9	2024-03-18 08:13:32	24€	Dupont Jean	Validé		Détails

10.10.1.208-8089order

Possibilité d’avoir le détails de la commande

Détails d’une commande

Accueil

Stocks

Commande

Utilisateurs

Fournisseurs

Déconnexion

Filtres :

Rechercher par demandeur

Status

Tous

Réinitialiser les filtres

Commandes

Id	Date	Prix total	Demandeur	Status	Raison	
1	2024-03-03 15:42:59	39€	ITH Kagnana	Validé		X
Commande Client						
	Libellé	Quantité	Prix unité	Catégorie	Stock	
	Blouse blanche	3	12.5€	Combinaison	50	
2	2024-03-05 21:20:36	48€	ITH Kagnana	Refusé	pas besoin	Détails
3	2024-03-05 22:07:21	162€	ITH Kagnana	Validé		Détails
4	2024-03-05 22:08:33	369€	ITH Kagnana	Validé	RAS	Détails
10	2024-03-18 10:39:32	37€	ITH Kagnana	Validé		Détails
11	2024-03-18 11:11:28	918€	ITH Kagnana	Validé		Détails
12	2024-03-18 11:11:48	204€	ITH Kagnana	En attente de validation		Détails
9	2024-03-18 08:13:32	24€	Dupont Jean	Validé		Détails

Le contrôle de la commande est uniquement accessible à l'administrateur. Il peut ainsi valider ou refuser une commande ce qui déclenchera la modification des stocks en base.

Il peut également préciser la raison de sa confirmation ou de son refus.

Page de contrôle des commandes

AccueilStocksCommandeUtilisateursFournisseursDéconnexion

X

Demands à traiter

Id	Date	Prix total	Demandeur	Status	Raison
12	2024-03-18 11:11:48	204	ITH Kagnana	En attente de validation	

Status : Validé

Commentaire :

Valider

Demands traitées

id	Date	Prix total	Demandeur	Status	Raison
1	2024-03-03 15:42:59	39	ITH Kagnana	Validé	
2	2024-03-05 21:20:36	48	ITH Kagnana	Refusé	pas besoin
3	2024-03-05 22:07:21	162	ITH Kagnana	Validé	
4	2024-03-05 22:08:33	369	ITH Kagnana	Validé	RAS
9	2024-03-18 08:13:32	24	Dupont Jean	Validé	
10	2024-03-18 10:39:32	37	ITH Kagnana	Validé	
11	2024-03-18 11:11:28	918	ITH Kagnana	Validé	

Page accessible depuis le sous menu "Contrôler" de Commande dans la barre de navigation

Si jamais on veut valider une commande mais qu'il n'y a pas assez de stock, ça affiche une erreur et empêche la validation.

AccueilStocksCommandeUtilisateursFournisseursDéconnexion

Attention, vous n'avez pas assez de stock pour ce produit : Molécule ARM

Demands à traiter

id	Date	Prix total	Demandeur	Status	Raison
12	2024-03-18 11:11:48	204	ITH Kagnana	En attente de validation	

Détails

Contrôler

Demands traitées

id	Date	Prix total	Demandeur	Status	Raison
1	2024-03-03 15:42:59	39	ITH Kagnana	Validé	
2	2024-03-05 21:20:36	48	ITH Kagnana	Refusé	pas besoin
3	2024-03-05 22:07:21	162	ITH Kagnana	Validé	
4	2024-03-05 22:08:33	369	ITH Kagnana	Validé	RAS
9	2024-03-18 08:13:32	24	Dupont Jean	Validé	
10	2024-03-18 10:39:32	37	ITH Kagnana	Validé	
11	2024-03-18 11:11:28	918	ITH Kagnana	Validé	

La création d'une commande est le point d'entrée de notre application. C'est principalement grâce à celle-ci que l'on va gérer les stocks.

Page formulaire de commande

Accueil

Stocks

Commande

Utilisateurs

Fournisseurs

Déconnexion

Rechercher par produit :

Rechercher

Libellé	Prix	Catégorie	Stock	Quantité	Action
Blouse blanche	12.5	Combinaison	50	1	+
Molécule ARM	102	Composant médicamenteux	1	1	+
Fiole 25ml	3.8	Verrerie	98	1	+
Masque en papier (lot de 20)	10.5	Combinaison	196	1	+
Pipette	3.8	Matériel de recherche	346	1	+

☒ Commande client
 ☐ Commande fournisseur

Fournisseur (Commande fournisseur uniquement)

MatMed

Panier

Libellé	Quantité	Prix	Catégorie	Prix total €	
Fiole 25ml	1	3.8	Verrerie	3.8	- x
Blouse blanche	2	12.5	Combinaison	25	- x
Molécule ARM	1	102	Composant médicamenteux	102	- x

Valider

Vider le panier

Possibilité de rechercher les produits en fonction de leur nom

Ne sont affichés que les produits dont l'utilisateur a les accès

Les produits s'auto-incrémentent lors de l'ajout

Page accessible depuis le sous menu "Ajouter" de Commande dans la barre de navigation

Le panier ne se réinitialise uniquement quand :

- l'utilisateur confirme sa commande
- l'utilisateur vide le panier
- l'utilisateur se déconnecte

Sinon, le panier ne change pas. Même quand on change de page ou que l'on ferme et reviens.

Cela est du grâce à la variable de session `$_SESSION["cart"]`

Dans `controllers/OrderController.php`

```

167 public function form($extra = null)
168 {
169     $resProduct = $this->product->getProductAccessible($extra["searchName"] ?? null);
170     $resProvider = $this->provider->getProvider();
171
172     $sendProduct = $resProduct["data"];
173     $allProviders = $resProvider["data"];
174
175     $error = isset ($resProduct["error"]) || isset ($resProvider["error"]) ? "Des données n'ont pas pu être récupéré" : null;
176
177     if (isset ($extra["error"])) {
178         $error = $extra["error"];
179     }
180
181     $cart = $_SESSION["cart"] ?? [];
182     $this->view("order/form-order-view", ["allProducts" => $sendProduct, "cart" => $cart, "providers" => $allProviders, "error" => $error]);
183 }

```

LA PARTIE UTILISATEURS

Accueil Stocks Commande **Utilisateurs** Fournisseurs Déconnexion

Consulter
Ajouter

Rechercher Status Tous Reinitialiser les filtres

Id	Entreprise	Nom	Prenom	Email	Rôle	Niveau d'accès	Status	
1	GSB	ITH	Kagnana	ikagnana@gmail.com	Administrateur	3	Validé	
2	GSB	Doe	John	j.doe@mail.com	Utilisateur	1	Refusé	
5	Fidele	Dupont	Jean	d.jean@fidele.com	Cliant	2	Validé	
9	GSB	Tapis	Bernard	b.tapis@gsb.com	Utilisateur	1	Validé	
11	GSB	Lecomte	Julien	lecomte.julien@gsb.com	Administrateur	3	Validé	

10.10.1.208-8089/usage/

Filtrage par nom/prénom ou email & par status

Possibilité d'avoir le détails de l'utilisateur

Détails d'un utilisateur

[← Retour](#)

Rôle Client

Nom de l'entreprise (Client uniquement)

Nom

Prénom

Email

Niveau d'accès maximum 1

Status Refusé

[Valider](#)

[Supprimer](#)

Formulaire d'ajout d'un utilisateur

Accueil Stocks Commande **Utilisateurs** Fournisseurs Déconnexion

Rôle Client

Nom de l'entreprise (Client uniquement)

Nom

Prénom

Email

Mot de passe

Niveau d'accès maximum 1

[Valider](#)

Page accessible depuis le sous menu "Ajouter" de Utilisateurs dans la barre de navigation

Les règles pour le mot de passe sont toujours les mêmes

LA PARTIE FOURNISSEURS

La partie fournisseurs permet de gérer les fournisseurs qui vont être sollicités lors d'une commande entrante.

Filtrage par nom
ou par email

AccueilStocksCommandeUtilisateursFournisseursDéconnexion

Rechercher...

Id	Nom	Email	
1	MatMed	matmed@mail.com	
2	Pharmacie des Lys	pharmacie-lys@hotmail.com	

Possibilité d'avoir le
détails du
fournisseur

Détail d'un fournisseur

AccueilStocksCommandeUtilisateursFournisseursDéconnexion

Rechercher...

1	<input type="text" value="MatMed"/>	<input type="text" value="matmed@mail.com"/>	<div>Modifier</div> <div>Supprimer</div> <div></div>
2	Pharmacie des Lys	pharmacie-lys@hotmail.com	

Présentation de l'application

Comment utiliser les éléments visuels

Navigation

La navigation est facilitée par le menu de navigation qui permet d'accéder à toutes les pages accessibles.

Pour ajouter un produit ou un utilisateurs, il suffit d'aller dans le sous-menu correspondant.

Le filtrage des données

Pour chaque page, on y retrouve une partie recherche et filtres.

Pour valider la recherche il faut impérativement appuyer sur le bouton.

Le filtrage se fait directement dans la requête SQL. Ce processus est préféré surtout quand le volume de données est grand.

```
42 public function getProvider($searchName = null)
43 {
44     try {
45         $this->db->query("SELECT * FROM provider WHERE name_pro LIKE :searchName or email_pro LIKE :searchName");
46         $this->db->bind("searchName", "%" . $searchName . "%");
47         $result = $this->db->fetchAll();
48     } catch (Exception $e) {
49         return ["data" => [], "error" => "Impossible de récupérer les fournisseurs."];
50     }
51
52     $allProvider = [];
53     for ($i = 0; $i < count($result); $i++) {
54
55         $provider = new Provider();
56         $provider->setProvider(
57             $result[$i]["name_pro"],
58             $result[$i]["email_pro"],
59             $result[$i]["id_pro"],
60         );
61
62         array_push($allProvider, $provider);
63     }
64
65     return ["data" => $allProvider];
```

Ici l'exemple est simple. Cependant il arrive parfois que l'on tri aussi en fonction du status comme par exemple pour les commandes. On fera alors une requête `getFilteredOrder()` qui prends le status désirée et le nom de l'appliquant s'il a été renseigné.

Annexes

Lien GITHUB vers le projet

<https://github.com/iKagnana/ap4-GSBStock>

Lien FIGMA MDC + Visualisation des pages

<https://www.figma.com/file/76U4UHDcESkAPfhtJLmOui/Conception-AP4?type=design&node-id=4-64&mode=design&t=unN81yIrS0EWVUs6-0>