

LIVRABLE FINAL

Eddy Druet – Rémy Auloy – Loic Boichut – Jocelyn
Bridon – Théo Dondarini – Ci Tang

Sujet : Shaker Magique
De Florence Mendes

I.	RAPPEL DU SUJET	2
II.	COMMENT UTILISER LE SITE ?	3
1.	LA PAGE D'ACCUEIL	3
a.	La barre de recherche	3
b.	Les filtres.....	3
c.	Les cocktails	3
d.	La pagination.....	4
2.	LA PAGE D'INSCRIPTION	4
3.	LA PAGE DE CONNEXION	4
4.	REJOINDRE UN HOTE.....	4
5.	LA PAGE DETAILLEE D'UN COCKTAIL.....	4
6.	LA PAGE DE PROFIL	5
7.	SA CARTE DE COCKTAIL	5
8.	LES INGREDIENTS EN STOCK	6
9.	LE PANNEAU D'ADMINISTRATION.....	6
a.	Liste des cocktails du site	6
b.	Liste des ingrédients du site	7
III.	GUIDE D'INSTALLATION DE L'APPLICATION	9
1.	PREREQUIS	9
1.	DEPLOIEMENT BASE DE DONNEES	10
2.	DEPLOIEMENT DJANGO (BACK-END)	11
a.	Installation des packages	11
b.	Configuration back-end	11
c.	Test du bon fonctionnement.....	11
3.	DEPLOIEMENT REACT (FRONT-END).....	12
a.	Installation de NodeJS	12
b.	Installation des packages	12
c.	Configuration	12
d.	Build	12
4.	DEPLACEMENT DES FICHIERS.....	12
5.	CONFIGURATION DE APACHE.....	12
a.	Activation du module WGSF	13
b.	Création du serveur virtuel back-end.....	13
c.	Création du serveur virtuel front-end	13
d.	Configuration .htaccess	14
e.	Activation des deux serveurs virtuels.....	14
6.	FIN DU DEPLOIEMENT	15
IV.	GUIDE DE MAINTENANCE	16
1.	FRONT-END	16
a.	Structure du projet.....	16
b.	Packages.....	17
c.	Commandes d'administration	18
2.	BACK-END	19
a.	Structure du projet.....	19
b.	Packages.....	21
c.	Commandes d'administration	22
d.	Configurations	23

I. RAPPEL DU SUJET

Notre projet s'appelle « Shaker Magique », le principe de base est simple : concevoir un site web proposant aux utilisateurs une liste de cocktails.

Vous devez savoir que sur ce site, il existe 3 catégories d'utilisateur :

- Le visiteur, c'est-à-dire utiliser le site sans vous connecter à un compte. Un visiteur n'a que très peu de possibilités, il peut rejoindre un hôte (un hôte est un utilisateur, qui offre la possibilité de visualiser sa carte de cocktails), afin de voir les cocktails que celui-ci propose, ou alors consulter parmi la liste de tous les cocktails de la base de données. Ainsi, un visiteur ne peut pas créer sa propre liste de cocktail.
- L'utilisateur, lui possède un compte, ainsi, en plus de pouvoir réaliser ce que peut faire un visiteur, il peut également créer sa propre liste de cocktails et d'ingrédients aux autres utilisateurs/visiteurs.
- Administrateur, explicitement, un administrateur peut tout faire sur le site, et a accès au panneau d'administration via l'URL « /admin ». Ce panneau d'administration offre la possibilité de consulter toute la liste des cocktails du site, il peut également ajouter un nouveau cocktail, en modifier ou même supprimer un cocktail. De même, il a accès à la liste de tous les ingrédients et peut en créer, ajouter et supprimer.

Vous l'aurez compris, l'utilité du site est de pouvoir proposer sa propre liste de cocktails à ses amis lors de « soirées cocktail » ou autre, en fonction des ingrédients que vous avez en stock. Car oui si vous proposez un cocktail dans votre carte, mais que vous n'avez pas tous les ingrédients pour le réaliser alors celui-ci n'apparaîtra pas, à moins de cocher le filtre pour. Mais nous détaillerons cela plus tard.

II. COMMENT UTILISER LE SITE ?

Avant d'expliquer ce que nous pouvons faire sur chaque page, il est important de préciser que le site est entièrement adapté sur mobile, qui était un critère du client. Pour faciliter la visibilité et la lecture des informations, très peu de captures d'écrans seront misent, je vous invite donc à ouvrir le site en parallèle.

1. La page d'accueil

Lorsque l'on arrive sur le site et que l'on n'est pas connecté. Il est affiché sur la page d'accueil, la liste de tous les cocktails de la base de données. Sur cette même page, est disponible une barre de recherche, des filtres à sélectionner ou encore une option de tri. De plus, en bas de la page se trouve une pagination afin d'améliorer la navigation si la liste des cocktails est grande.

a. La barre de recherche

La barre de recherche fournit un outil de sélection des cocktails afin d'afficher les cocktails qui vous correspondent. Elle peut s'utiliser comme une barre de recherche classique, la recherche affichera les cocktails qui contiennent l'intégralité des mots recherchés.

Exemple : « Rhum orange » affichera les cocktails qui contiennent du rhum et de l'orange.

Cela dit des recherches plus complexes sont possible, pour cela nous avons considéré que lorsqu'un utilisateur effectue une recherche il possède au moins un critère particulier, celui-ci se place alors naturellement au début de la recherche, et peut être suivi par des mots séparés par une virgule.

Voici quelques exemples d'utilisation (les espaces après les virgules sont facultatifs) :

- « Orange, rhum, vodka » affichera des cocktails à l'orange contenant soit du rhum soit de la vodka.
- « Rhum sirop de sucre de cannes, orange, ananas » affichera des cocktails au rhum et au sirop de sucre de cannes contenant soit de l'orange soit de l'ananas.
- Une recherche plus large est possible en saisissant une virgule dès le début de la recherche.
« , ananas, orange » affichera les cocktails qui contiennent soit de l'ananas soit de l'orange

b. Les filtres

De plus, vous pouvez choisir d'afficher les cocktails avec la catégorie « Apéritif », « Digestif » ou encore « Sans alcool ». Vous pouvez également cumuler les filtres, par exemple, si vous cochez « Apéritif » et « Digestif », sera retourné uniquement les cocktails possédant les deux catégories.

c. Les cocktails

Comment l'affichage des cocktails est géré ?

Comme dit précédemment, si nous ne sommes pas connectés, il est affiché la liste de tous les cocktails de la base de données. Une fois connecté, vous serez redirigé sur la page d'accueil, et il sera toujours listé tous les cocktails de la base de données. En revanche, le client souhaitait avoir la possibilité de lister les cocktails où un seul ingrédient manquait. C'est pourquoi, vous pouvez sélectionner cette option via le système de tri (disponible juste à côté des filtres).

Pour finir, si vous décidez de rejoindre un autre utilisateur, c'est-à-dire un hôte. Il sera affiché sur la page d'accueil, tous les cocktails RÉALISABLES (ex. sans ingrédients manquants) que cet hôte propose dans sa liste. À noter, il est également possible d'utiliser le tri « 1 ingrédient manquant » lorsque vous rejoignez un hôte.

d. La pagination

Assez rapidement, la pagination est très simple à comprendre, vous pouvez accéder à la page précédente, suivante. Ou accéder directement à la première page ou à la dernière.

2. La page d'inscription

La page d'inscription, permet à l'utilisateur de créer un compte en remplissant un formulaire d'informations personnelles et de connexion afin d'établir notamment un pseudo et un mot de passe. Afin de valider l'enregistrement, l'utilisateur se voit contraint d'accepter la politique de confidentialité mise en vigueur sur l'application. Cette dernière est consultable en cliquant directement sur le terme « *politique de confidentialité* » ou encore via le bouton en bas de page.

3. La page de connexion

La page de connexion permet à l'utilisateur de s'identifier grâce à son pseudo et son mot de passe, il a la possibilité de rester connecté en cochant la case « *Se souvenir de moi* », ainsi il sera connecté dès l'accès au site et cela même après la fermeture du navigateur.

Dans le cas où l'utilisateur ne possède pas de compte, il a accès à la page d'inscription en cliquant sur « *Inscription* » en dessous du bouton de connexion.

4. Rejoindre un hôte

La page pour rejoindre un hôte permet d'accéder aux cocktails proposés par un membre en saisissant le pseudo de ce membre dans le champ et en cliquant sur « *Rejoindre* ».

Pour quitter un hôte, il suffit d'effacer la barre de saisie.

5. La page détaillée d'un cocktail

Cette page est accessible dès que l'on va cliquer sur un cocktail affiché sur la page d'accueil. Une fois sur cette page, les informations du cocktail s'affichent :

- Titre
- Catégorie

- Force
- Ingrédients nécessaires pour être réalisé
- Description
- Image cocktail
- Note du cocktail

Pour préciser, cette page n'est pas seulement une page d'information, en effet, c'est à partir de cette page que vous pourrez attribuer une note sur 5 étoiles à un cocktail. Pour cela, vous devez être connecté, cliquer sur l'étoile correspondant à votre note. Vous pouvez modifier votre note, ou bien même la supprimer en cliquant sur votre note actuelle.

Pour vous aider à comprendre :



Sans note



Avec note

Les étoiles remplies correspondent à la moyenne des notes, car chaque utilisateur peut noter un cocktail. Le contour jaune représente votre note. Ainsi, sur l'exemple, la moyenne est de 3 étoiles sur 5 et votre note est de 4 étoiles sur 5.

Pour finir, si l'on clique sur les étoiles sur la page d'accueil, cela ne marchera pas, le seul moyen de noter un cocktail est d'accéder à la fiche détail du cocktail.

6. La page de profil

URL : « /profil »

Cette page est également accessible uniquement si l'on est connecté, en effet, une fois connecté dans la barre de navigation, un nouveau lien apparaît, en cliquant dessus vous accédez à votre profil.

Vous voyez donc 3 informations modifiables, votre pseudo, votre courriel ou encore votre mot de passe. Si vous modifier vos informations, il suffit de cliquer sur le bouton « Sauvegarder » pour les valider, la page se rechargera validant ainsi vos modifications. À droite du bouton de sauvegarde, se trouve un autre bouton permettant à l'utilisateur de se déconnecter, je précise que vous pouvez également vous déconnecter via la barre de navigation.

Enfin, sur cette même page, se trouve l'accès à votre carte des cocktails et votre liste des ingrédients en stock. Pour y accéder, cliquez sur les boutons.

7. Sa carte de cocktail

URL : « /hote/cocktails »

Tout comme la page de profil, cette page n'est accessible que si vous êtes utilisateur, donc uniquement lorsque vous êtes connecté.

Une fois sur cette page, se trouvent 2 éléments :

- Une barre de recherche

- La liste des cocktails de la base de données

Le fonctionnement de cette page est simple et intuitif. Vous pouvez rechercher par nom, le cocktail que vous souhaitez ajouter. La recherche n'est pas au nom exact, imaginons que votre cocktail se nomme « Alexandra » si vous tapez « Alexan » le cocktail sera retourné.

Si vous souhaitez ajouter un cocktail à votre liste, cliquez simplement sur la case à cocher dans la colonne « Proposer », si cette case à cocher est activée cela signifie que vous proposez le cocktail. Une fois proposés, les cocktails sont immédiatement « poussés vers le haut ». Notez également, que vous pouvez trier la liste en cliquant sur le nom de colonne, ainsi vous pouvez trier la liste des cocktails par nom.

Si vous proposer un cocktail, mais que vous ne disposez pas de tous les ingrédients pour pouvoir réaliser ce cocktail, alors celui-ci ne sera pas affiché sur la page d'accueil. Pour rappel, sont affichés uniquement les cocktails réalisables (sauf si le filtre « 1 ingrédient manquant » est utilisé).

8. Les ingrédients en stock

URL : « /hote/ingredients »

De même, cette page est accessible uniquement aux utilisateurs.

Cette page fonctionne sur le même principe que la page de sa carte de cocktails. Une barre de recherche par nom est disponible pour rechercher votre ingrédient.

Une colonne « Degré » est présente pour pouvoir trier les ingrédients par degré d'alcool.

Si vous souhaitez indiquer ou non si vous avez l'ingrédient dans votre stock, il suffit de cocher ou non la case à cocher, de la même manière que pour votre carte de cocktails.

9. Le panneau d'administration

URL : accessible en tant qu'administrateur « /admin »

Vous pouvez vous en douter, cette page est accessible uniquement si vous êtes un utilisateur administrateurs. Sinon vous serez redirigé sur la page d'accueil.

Sur cette page, deux choix s'offre à vous :

- Accéder à la liste de tous les cocktails de la base de données
- Accéder à la liste de tous les ingrédients de la base de données

Ainsi, vous pourrez ajouter ou non de nouveaux ingrédients et cocktails. Car ce sont uniquement les administrateurs qui peuvent réaliser ces tâches.

a. Liste des cocktails du site

URL : accessible en tant qu'administrateur « /admin/cocktails »

Cette page permet à un administrateur d'ajouter, modifier ou supprimer un cocktail de la base de données.

Pour cela, vous pouvez rechercher votre cocktail via une barre de recherche, tout comme précédemment. Vous pouvez également trier la table par ID, ou par nom en cliquant sur l'en-tête de la colonne.

Si vous souhaitez modifier un cocktail, cliquez sur l'icône de modification, vous serez redirigé sur la page de modification.

Enfin, si vous souhaitez supprimer un cocktail, cliquez sur l'icône de suppression, un message de confirmation vous est demandé, une fois validé votre cocktail disparaîtra.

i. Page d'ajout / modification / suppression de cocktail

La page d'ajout et de modification possède le même squelette. À la différence, que la page de modification chargera les informations du cocktail à modifier.

Pour créer un cocktail, il est nécessaire les mêmes informations évoquées précédemment lorsque nous parlions des fiches détail d'un cocktail. Il vous suffit d'écrire un titre, cocher vos catégories, indiquer une description et choisir la force du cocktail.

Vous devez également choisir une image à transférer, pour votre cocktail.

Concernant les ingrédients, tout comme pour les autres pages, une barre de recherche est disponible pour rechercher directement l'ingrédient.

Tant que vous ne cliquez pas sur la case « Contenir » vous ne pouvez pas indiquer de quantité et l'unité. Si vous souhaitez indiquer ces informations cocher la case « Contenir ».

Une fois toutes vos informations remplies, cliquez sur « Créer le cocktail ».

Concernant la modification, c'est assez intuitif, cliquez sur le bouton de modification et modifier ce que vous souhaitez. Une fois fini cliquer sur le bouton « Modifier le cocktail ».

b. Liste des ingrédients du site

URL : « /admin/ingredients »

Cette page permet à un administrateur d'ajouter, modifier ou supprimer un ingrédient de la base de données.

Comme pour les cocktails, vous pouvez rechercher un ingrédient via la barre de recherche et trier la table.

Comme pour la page d'administration des cocktails, cliquez sur les icônes pour modifier ou supprimer un ingrédient.

ii. Page d'ajout / modification / suppression d'ingrédient

Le squelette de la page d'ajout et de modification est le même. À la différence, que la page de modification chargera les informations du cocktail à modifier.

Si vous cliquez sur « Créer un ingrédient », vous pouvez ajouter votre nouvel ingrédient à la base de données en lui donnant un intitulé et sa force, une fois cela fait, validez.

Pour modifier, cliquer sur l'icône de modification, les informations de l'ingrédient chargerons. Modifiez-les puis validez, vous serez redirigé sur la page précédente.

Pour supprimer, tout simplement, cliquez sur l'icône de suppression.

III. GUIDE D'INSTALLATION DE L'APPLICATION

1. Prérequis

En premier, Django fonctionnant sur Python (version 3.5+), je vous laisse installer ce dernier.

Par la suite, il est possible d'utiliser différents serveurs Web pour déployer Django selon l'interface Python que vous configurez :

- WSGI : Gunicorn, uWSGI, Apache
- ASGI : Daphne, Hypercorn, Uvicorn

L'application étant configurée par défaut pour WSGI, et Apache étant le service le plus commun, nous allons utiliser celui-ci dans ce guide d'installation.

Avant de continuer la suite du tutoriel, je vous laisse installer Apache sur votre serveur.

1. Déploiement base de données

Installez un SGBD sur votre machine, et exécutez le script « BDD/MLD_MySQL_Django.sql » pour construire le schéma nécessaire à l'application.

Vous pouvez exécuter le script « BDD/DATA_MySQL_Django.sql » pour avoir un échantillon de données de test.

2. Déploiement Django (back-end)

Nous allons maintenant démarrer le déploiement de Django sur le serveur.

a. Installation des packages

Il est nécessaire d'installer Django et tous les packages nécessaires au fonctionnement du back-end de l'application.

Pour cela installez-les via la liste des packages exportés dans le fichier « backend/requirements.txt » avec la commande « pip install -r requirements.txt ».

b. Configuration back-end

Ouvrez le fichier de configuration Django « backend/core/settings.py ».

i. Base de données

Remplissez les champs en raccord avec la configuration de votre SGBD.

```
DATABASES = {
    'default': {
        'ENGINE' : 'django.db.backends.mysql',
        'NAME' : '',
        'USER' : '',
        'PASSWORD' : '',
        'HOST' : '',
        'PORT' : '3306',
    }
}
```

ii. Autorisation CORS

CORS doit être configuré pour accepter uniquement les requêtes provenant de l'URL du front-end.

```
CORS_ALLOWED_ORIGINS = [
    # Indiquer l'URL du front-end sans / à la fin
    "http://localhost:3000"
]
```

c. Test du bon fonctionnement

Nous allons vérifier si le serveur démarre correctement. Pour cela, démarrez le serveur avec la commande « python manage.py runserver » dans le répertoire « back-end ». Si vous avez utilisé un environnement virtuel, activez-le d'abord.

Si le serveur démarre et fonctionne, il devrait afficher le message « System check identified no issues (0 silenced). ».

Vous pouvez essayer de consulter une API, par exemple : <http://localhost:8000/api/cocktails/> (le / à la fin est important), un JSON doit s'afficher avec quelques cocktails.

3. Déploiement React (front-end)

Nous allons maintenant déployer React sur le serveur.

a. Installation de NodeJS

NodeJS ne vous sera pas utile en production mais lors du développement du front-end. Il est toutefois nécessaire pour build le projet.

Je vous laisse donc installer NodeJS sur votre machine.

b. Installation des packages

L'application React nécessite plusieurs packages à son fonctionnement.

Dans le répertoire « frontend », exécutez la commande « npm install » pour installer tous les packages listés dans « packages.json ».

c. Configuration

Nous allons configurer l'URL vers le back-end Django, ouvrez le fichier « frontend/src/Axios.js ».

Pensez à mettre le port sur 8000, ainsi qu'ajouter à la fin « /api/ ». Utilisez le même nom de domaine que le front-end.

```
/**
 * URL vers le back-end
 */
const baseUrl = "http://127.0.0.1:8000/api/";
```

d. Build

Nous allons build l'application React pour générer les fichiers sources « standalone » qui n'ont pas besoin de NodeJS.

Dans le répertoire « frontend », exécutez la commande « npm run-script build ». Le résultat se trouve dans le dossier « frontend/build ». Vous pouvez lancer « index.html » pour vérifier si le visuel du site s'affiche. Nous vous inquiétez pas de l'erreur CORS, le serveur back-end n'est pas démarré.

4. Déplacement des fichiers

Déplacez le dossier « backend » vers le répertoire « /var/www/ ».

Déplacez le dossier « frontend » vers le répertoire « /var/www/ ».

5. Configuration de Apache

Maintenant que nous avons préparé le nécessaire au déploiement sur Apache, nous allons configurer celui-ci.

Vu que nous utilisons Django Rest Framework, c'est-à-dire que le front-end communique par l'appel de fonctions de l'API situé sur le back-end, il nous faut 2 serveurs : un pour le front-end qui desservira les fichiers statiques, et un pour le back-end.

Nous utiliserons pour cela, les serveurs virtuels (VirtualHost) d'Apache.

a. Activation du module WSGI

Installez le package Python : « pip install mod_wsgi ».

Ajoutez dans le fichier de configuration Apache, les lignes :

```
LoadModule rewrite_module modules/mod_rewrite.so
LoadModule wsgi_module /usr/lib/apache2/modules/mod_wsgi.so
# Si vous avez installé les packages, Django dans environnement virtuel
# WSGIPythonHome /var/www/backend/virtualenv
WSGIPythonPath /var/www/backend/
```

b. Création du serveur virtuel back-end

Nous allons faire tourner Django sur un serveur virtuel sur le port 8000.

Créez un nouveau fichier « backend.conf » dédié à la configuration du serveur virtuel du back-end et y insérer en prenant soin de changer les champs ServerName et ServerAdmin, et éventuellement les chemins d'accès au backend si le chemin est différent de « /var/www/ » :

```
Listen 8000
<VirtualHost *:8000>
    ServerName nomdomaine.fr/api # Bien mettre le /api
    ServerAdmin email@site.fr
    DocumentRoot /var/www/backend/

    # Assets sur le back-end
    Alias /media/ /var/www/backend/media/

    <Directory /var/www/backend/media/>
        Require all granted
    </Directory>

    WSGIScriptAlias / /var/www/backend/core/wsgi.py
    WSGIPassAuthorization On # Obligatoire pour utiliser Django JWT Token

    <Directory /var/www/backend/core/>
        <Files wsgi.py>
            Require all granted
        </Files>
    </Directory>
</VirtualHost>
```

c. Création du serveur virtuel front-end

Nous allons faire tourner le front-end sur un serveur virtuel sur le port 80 de base.

Créez un nouveau fichier « frontend.conf » dédié à la configuration du serveur virtuel du front-end et y insérer en prenant soin d'avoir pareil que le back-end pour ServerAdmin et ServerName, changez les chemins d'accès si nécessaire :

```
<VirtualHost *:80>
  ServerName nomdomaine.com
  ServerAdmin email@site.fr
  DocumentRoot /var/www/frontend/build/

  # Fichiers statiques lors du build React
  Alias /static/ /var/www/frontend/build/static/
  <Directory /var/www/frontend/build/>
    Require all granted
  </Directory>
</VirtualHost>
```

d. Configuration .htaccess

L'application React utilise un module qui nécessite une réécriture des URL, sans ça il ne fonctionne pas.

Dans le répertoire « /var/www/ », créez un fichier « .htaccess » et insérez ceci :

```
RewriteEngine On
RewriteBase /
RewriteRule ^index\.html$ - [L]
RewriteCond %{REQUEST_FILENAME} !-f
RewriteCond %{REQUEST_FILENAME} !-d
RewriteCond %{REQUEST_FILENAME} !-l
RewriteRule . /index.html [L]
```

Retournez dans le fichier de configuration Apache et cherchez la section correspondant aux permissions, et remplacez par ceci :

```
<Directory /var/www/>
  Options Indexes FollowSymLinks
  AllowOverride All
  Require all granted
</Directory>
```

e. Activation des deux serveurs virtuels

Désactivez la configuration VirtualHost par défaut, et activez vos deux serveurs virtuels :

```
a2dissite 000-default.conf
a2ensite backend.conf
a2ensite frontend.conf
```

Et redémarrez le service Apache.

6. Fin du déploiement

Maintenant que l'application est déployée, vous pouvez désactiver le mode débogage de Django « backend/core/settings.py » :

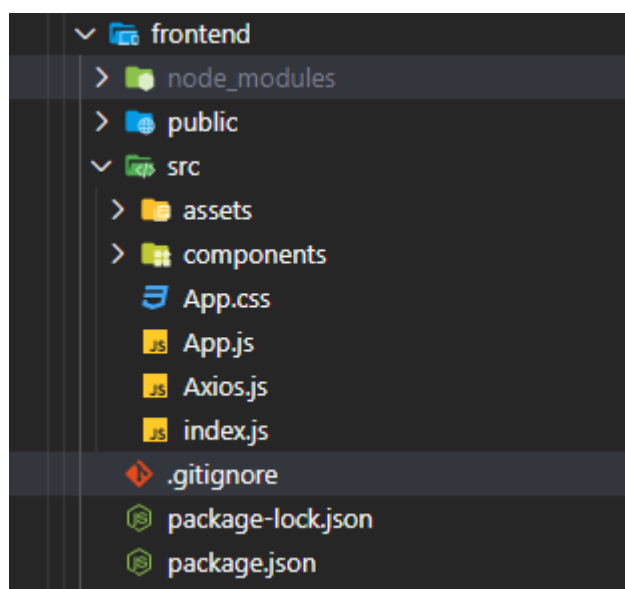
```
DEBUG = False
```


IV. GUIDE DE MAINTENANCE

1. Front-end

a. Structure du projet

Afin de mieux comprendre le projet, détaillons l'arborescence du projet.



Tout ce qui touche au visuel et à la liaison avec le serveur back-end (ex. le front-end) se trouve dans le dossier « frontend ».

Nom répertoire/fichier	Utilité
node_modules	Dossier contenant tout ce qui est nécessaire pour démarrer le serveur front-end, ainsi que quelques extensions.
public	Utile si vous souhaitez, changer le titre de l'application ou encore l'icône du site. Il contient les fichiers statiques.
src	Dossier le plus important, c'est lui qui va regrouper tout ce qui construit le frontend du site.
assets	Se trouve les images et icônes utilisées dans le site. À noter que les images des cocktails ne sont pas présentes ici, car elles sont stockées au niveau back-end.
src/components	Regroupe tous les composants qui font le visuel de votre site (les fichiers .jsx et .css).
App.js	Fichier principal qui réunit tous les composants et crée les chemins d'accès aux différentes pages.
Axios.js	Fichier contenant toutes les fonctions qui permettent de communiquer avec les APIs du serveur backend. C'est notamment, dans ce fichier que vous pouvez changer l'URL avec laquelle vous communiquez, ex. l'URL de votre serveur backend.
index.js	Fichier qui « monte » le composant « App.js », il n'a pas besoin d'être touché.

b. Packages

i. Axios

Permet d'envoyer des requêtes au serveur back-end.

ii. Interweave

Permet d'afficher de façon sécurisée du code HTML provenant d'une source extérieure (ex. depuis un JSON reçu). Empêche les injections JS.

iii. React Table

Permet de créer des tables très modulaires et facilement.

iv. React Router

Permet de créer URL dirigeant vers certains composants spécifiques. Par défaut, React ne permet la création d'URL (ex. « monsite.fr/connexion », « monsite.fr/cocktail/5 »...).

c. Commandes d'administration

Pour démarrer le serveur de développement, je le rappelle, vous devez avoir NodeJS sur votre machine.

Pour lancer le serveur de développement rien de plus simple, ouvrez un cmd, rendez-vous dans le dossier « frontend » et tapez la commande :

```
npm start
```

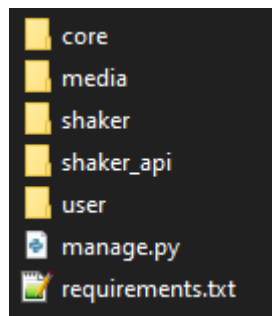
Le serveur va se lancer, et ouvrir automatiquement une page internet sur laquelle se trouve le site.

Si jamais vous la voulez, la commande pour compiler le projet, la commande est :

```
npm run-script build
```

2. Back-end

a. Structure du projet



Nom répertoire/fichier	Utilité	Information complémentaire
requirements.txt	Ensemble des packages Python requis pour le projet.	
manage.py	Outil Django fournissant des lignes de commandes.	
core	Répertoire principal du projet.	
media	Répertoire où stocker les fichiers statiques comme les images.	
media/cocktail	Miniatures de chaque cocktail.	Par défaut, les images prennent le format « IDCOCKTAIL.jpg ».
user	Sous-projet qui englobe l'API d'authentification.	
shaker_api	Sous-projet concernant Django Rest Framework. Il dessert les API.	
shaker	Sous-projet concernant Django de base. Il dessert les modèles, et les fichiers statiques constituant le front-end.	Dans Shaker Magique, il n'est utilisé que pour les modèles. Le contenu statique étant React.
shaker/models.py	Contient tous les modèles mappant les tables de la base de données.	
shaker_api/views.py	Contient toutes les vues. Chaque classe correspond à une API.	
shaker_api/permissions.py	Permet de définir les permissions de lecture et d'écriture sur les vues.	
shaker_api/serializers.py	Permet de transformer les données Python ⇔ JSON pour être exploité dans les vues.	
shaker_api/URLs.py	Permet de définir les points d'accès aux vues.	
user/models.py	Contient tous les modèles mappant les tables de la base de données liés aux utilisateurs.	

user/pwdvalidators.py	Contient tous les validateurs de mot de passes (ex. qu'il doit contenir 6 caractères...).	
user/URLs.py	Permet de définir les points d'accès aux vues de l'API d'authentification.	Voir aussi core/URLs.py qui contient d'autres points d'accès à l'API d'authentification.
user/views.py	Contient toutes les vues de l'API d'authentification.	
core/settings.py	Fichier de configuration du back-end.	
core/URLs.py	Permet de définir les points d'accès aux vues de l'API d'authentification.	

b. Packages

i. Django Rest Framework

Django Rest Framework permet l'implémentation facile d'API Restful sur Django.

Il est utilisé partout dans le projet, le front-end communique ainsi avec des fonctions de l'API pour récupérer les données de la base de données.

Le format de communication est JSON.

En aucun cas, le back-end ne gère l'affichage des données, grâce au Django Rest Framework, il ne s'occupe que de fournir les données dont React gèrera l'affichage.

ii. Simple JWT

JWT Token est le système utilisé pour s'authentifier sur les vues API.

Plus d'informations sur le fonctionnement : <https://jwt.io/>

Simple JWT est le package Python-Django Rest Framework utilisé dans le projet back-end qui implémente ce système d'authentification.

iii. Pillow

Pillow est un package permettant le traitement d'images sur Python.

Il est nécessaire sur le projet Django pour transférer des images sur le serveur.

iv. Django CORS Headers

Django CORS Headers permet d'implémenter l'en-tête Cross-Origin Resource Sharing (CORS) dans les requêtes HTTP.

Il permet de vérifier la source des requêtes.

c. Commandes d'administration

i. Démarrer le serveur de développement

```
python manage.py runserver
```

ii. Créer un compte administrateur

```
python manage.py createsuperuser
```

d. Configurations

Le fichier de configuration se trouve dans « backend/core/settings.py ».

i. Mode débogage

Pour activer ou désactiver le mode débogage.

```
DEBUG = True
```

ii. Authentification SGBD

Ici, vous pouvez écrire les informations de connexion à votre SGBD.

```
DATABASES = {
    'default': {
        'ENGINE' : 'django.db.backends.mysql',
        'NAME' : '',
        'USER' : '',
        'PASSWORD' : '',
        'HOST' : '',
        'PORT' : '3306',
    }
}
```

iii. Configuration Simple JWT

Ici, vous trouverez les différents paramètres de configuration du package Simple JWT.

Vous pouvez notamment configurer les délais d'expiration des tokens « ACCESS_TOKEN_LIFETIME » et « REFRESH_TOKEN_LIFETIME ».

```
SIMPLE_JWT = {
    'ACCESS_TOKEN_LIFETIME' : timedelta(minutes=10),
    'REFRESH_TOKEN_LIFETIME' : timedelta(days=3),
    'ROTATE_REFRESH_TOKENS' : True,
    'BLACKLIST_AFTER_ROTATION' : True,
    'ALGORITHM' : 'HS256',
    'SIGNING_KEY' : SECRET_KEY,
    'VERIFYING_KEY' : None,
    'AUTH_HEADER_TYPES' : ('JWT',),
    'USER_ID_FIELD' : 'id',
    'USER_ID_CLAIM' : 'user_id',
    'AUTH_TOKEN_CLASSES' : ('rest_framework_simplejwt.tokens.AccessToken',),
    'TOKEN_TYPE_CLAIM' : 'token_type',
}
```

iv. Configuration CORS

Pour autoriser certaines sources des requêtes (notamment le nom de domaine du front-end), vous pouvez les spécifier ici.


```
CORS_ALLOWED_ORIGINS = [  
    # Indiquer l'URL du frontend sans / à la fin  
    "http://localhost:3000"  
]
```

v. Configurer l'image par défaut des cocktails

Pour changer l'image par défaut des cocktails, il faut se diriger dans le répertoire « backend/media/cocktail ». Vous devez intituler le fichier : « default.jpg », faites attention que l'extension soit bien « jpg ».

vi. Autres paramètres

- Configurer les critères de validation d'un mot de passe.

AUTH_PASSWORD_VALIDATORS

- Liste des applications supplémentaires installés au projet Django.
Vous y trouvez le sous-projets « shaker », « shaker_api », « user »...

INSTALLED_APPS

- Choisir la classe modèle d'utilisateur par défaut pour l'authentification.

AUTH_USER_MODEL