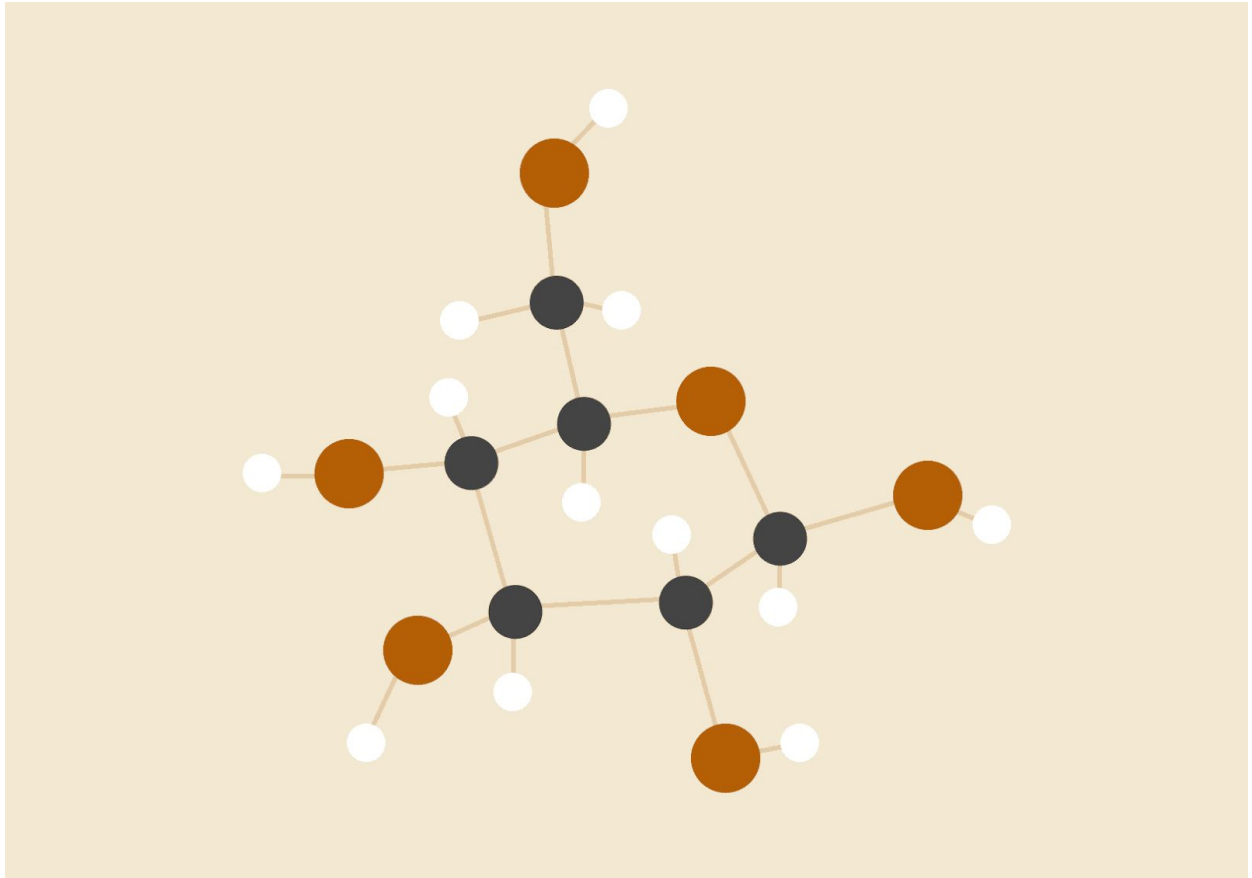


RAPPORT MINI PROJET



HEMAIZI Camilia

NOUARI Heythem

10/06/2020

Master 1 - STL

DESCRIPTION DU PROJET

Notre projet est une application web qui permet de réserver des vols en utilisant l'API : **AVIATION STACK** (<https://aviationstack.com/documentation>) pour la récupération des vols entre un aéroport de départ et un aéroport d'arriver , un jour donné par l'utilisateur .

Manuel du projet :

Nous avons développé ce projet sur Eclipse Donc il est préférable de lancer ce projet sur cet outils en liant le projet au serveur Tomcat 7 . et créer la base de donnée Mysql comme précisée dans la section Base de Donnée (page 3)

une fois cela est fait il faut accéder à l'url : <http://localhost:8080>: PCR/index pour accéder à la page d'accueil. Puis s'inscrire en cliquant sur le bouton dans la nav-barre, le site vous redirige après vers la page d'accueil . Maintenant il faut cliquer sur " réserver un vol" puis sélectionner un aéroport de départ , un deuxième d'arriver et la date de votre voyage et validez le formulaire . une liste de vols correspondants à votre recherche . sélectionnez un vol en cliquant sur réserver .

c'est fait ! vous pouvez consulter vos réservation maintenant dans la rubrique "mon compte -> mes réservations".

Remarque: l'API ces dernier jour montre des vols avec des aéroports peu connus donc pour tester l'application il faut accédé à ce lien :

http://api.aviationstack.com/v1/flights?access_key=a60a384bbabf6a2f875b4b4f94aeee47 saisir les aéroports de départ et d'arrivée du data[0] par exemple pour s'assurer qu'il y a bien des vols correspondants à votre recherche . Nous vous informons aussi que l'api a rendu payante la fonctionnalité de rechercher des vols par date . donc pour les clés gratuite (comme la nôtre) elle n'affiche que les vols du lendemain . notre application ne prends pa compte de la date saisie dans le formulaire (sinon les résultats seront toujours vide) . Nous nous excusons pour cela mais nous avons pris connaissance de ça en retard et nous avons changer d'api avant celle pour une raison pareille (Air France-klm). Mais nous voulons vous signaler que tout ce qu'on doit changer pour que l'application pourra réserver des vols dans la date saisie par l'utilisateur

nous devons juste rajouté '&flight_date=date_donnée_par_le_client' à la fin de l'url de connexion. Et bien sûr payer l'abonnement.

vous trouverez sur ma chaine youtube une video démonstrative :

https://www.youtube.com/channel/UCeVgF285xtojZw2ppZT5CiQ?view_as=subscriber

Fonctionnalités :

- Connexion et inscription
- Recherche d'un vol
- Réservation d'un vol
- Consultation des informations personnels
- Consultation des réservations

ARCHITECTURE DÉTAILLÉE

Servlets:

- SignIn : la servlet affiche la vue signin.jsp qui communique avec la base de Données pour vérifier l'existence d'un compte client correspondant à ces identifiants récupéré de la jsp signin.jsp après validation du formulaire.
- SignUp : gère l'affichage de la vue signup.jsp et le traitement du formulaire de cette dernière afin d'enregistrer un utilisateur.
- ListeVols : se connecte avec l'API et parcourt les objets Json pour rechercher les vols correspondants à la recherche .
- Réservation : gère la création des réservations et leur affichage .
- Propose : affiche la vue contenant le formulaire de recherche de vol.
- Déconnexion : déconnecte l'utilisateur en tuant la session et redirige l'utilisateur vers la page d'accueil.
- Profil : gère l'affichage des infos personnelles de l'utilisateur .

Beans:

- Utilisateur : sert à sauvegarder les informations de l'utilisateur dans la variable Session.
- Vol : nous aide dans la recherche des vols dans la servlet
- My Reservation : c'est un vol attribué à un utilisateur
- JsonResponse Handler : permet de Parser la réponse JSON récupérée via l'API.

Base de Données

pour avoir une base de données compatible avec notre application il faut créer une base de données Mysql portant le nom "bd_pcr", son mot de passe : "123" et un user nommé 'java' en lui attribuant toutes les permissions sur cette bdd.

ensuite vous trouverez dans notre projet le fichier "requetes.sql" qui contient les requêtes sql de création de tables avec quelques insertions.

CHALLENGES

les points que nous jugeons durs dans ce projet sont :

- choisir la bonne API
- la recherche à l'intérieur de la réponse JSON
- la gestion des problèmes de connexion avec la base de données au début du projet.
- et spécifiquement pour notre projet : apprendre quelques notions de java ee qui est un langage qu'on a jamais utilisé mais que nous sommes contents d'apprendre ses bases à travers ce projet.
- la difficulté de récupérer le projet via git et de l'importer dans eclipse-workspace et la base de données .chose qui a freiné fortement notre avancement car on a dû travailler le projet sur un seul ordinateur.

CONCLUSION

Ce projet a approfondi nos connaissances en web .Nous a fait tous les deux apprendre à programmer en JAVA EE et l'utilisation d' APIs . Par manque de temps nous avons eu à choisir entre implémenter des fonctionnalités qu'on a jugé plus importantes que d'autres. Nous avons aimé rajouter par exemple l'option du contact avec un service Mailer , possibilité de modification des infos personnelles et génération de billet sous format pdf .