

Visites par Virgil Tourneur, Clément Laillé, Frédéric Vezin, Clément Le Borgne

Pour la connexion adresse mail : test@jee.com et mot de passe : 1

Vous pouvez également créer un compte

Installation de l'application web (Désolé pour le désagrément occasionné) :

- Dézipper l'archive.zip et importer les deux projet avec l'extension.war en respectant les noms déjà attribué
- Modifier le Dynamic Web Projet UtiliserVisites. Il faut changer l'adresse, nom d'utilisateur et mot de passe, dans le fichier VérifierLogin.java et CreerUtilisateur.java situé dans puis dans le package src/fr/eseo/login pour qu'ils correspondent à ceux de votre phpmyadmin.
- Il faudra aussi aller chercher dans le WebContent de ce projet et ouvrir TrouverVisite.jsp et changer tous les paramètres de connexion à la base de données
- Faire la même chose pour le Dynamic Web Projet GestionVisites. Dans le fichier GestionVisites.java situé dans le package src/fr/eseo/jee
- Importer les deux fichier sql ClientsVisites et GestionnaireVisites en respectant les noms des fichiers comme noms de base de données
- Lancer le projet

Outils utilisés :

- Git pour travailler en simultané sur le même dossier projet.
- phpMyAdmin pour implémenter puis gérer la base de données
- eclipse : pour coder nos servlets, .jsp, .html, classes java et tests JUnit
- Apache Tomcat 9.0
- Apache CXF

Améliorations apportées :

- annulerVisite :
 - rajout d'un identifiant client en paramètre pour vérifier que le client annule sa réservation et pas celle d'un autre
- base de données :
 - ajout d'un identifiant pour chaque table afin de mettre en place des clés étrangères dans Réservations
- Création de compte :
 - inscription en renseignant un nom, un prénom, une adresse mail et un mot de passe

- Ajout d'un email et d'un mot de passe pour chaque client :
 - Pour ouvrir une session lié à un et un seul client

Tests effectués :

- reserverVisite :
 - réserver une visite existante
 - ne pas pouvoir réserver une visite n'existant pas
- payerVisite
 - payer une réservation existante
 - ne pas pouvoir payer une réservation n'existant pas
- annulerVisite :
 - client qui annule une réservation payée existante et le concernant uniquement
 - client qui annule une réservation non payée existante et le concernant uniquement
 - ne pas pouvoir annuler une réservation n'existant pas
 - ne pas pouvoir annuler une réservation existante mais ne concernant pas le client
- trouverVisite :
 - ne pas trouver une visite n'existant pas
 - trouver une visite existante grâce au type de visite
 - trouver plusieurs visites existantes grâce à la ville
 - trouver une visite existante grâce à la date
 - trouver une visite existante grâce au prix
 - trouver une visite existante grâce à son code

Connexion à un autre groupe (Taxi) :

Nous offrons la possibilité de trouver un ou plusieurs taxis disponibles pour une date donnée en échange de quoi l'autre groupe pourra trouver les visites disponibles pour différentes conditions.

Nous avons ouvert le port mysql, sur la machine hôte, pour tout le monde sinon la connection à la base de données était refusée. Nous ne pouvons pas accéder à la base de données des Taxis (ils n'avaient pas ouvert ce port) mais l'autre groupe a pu utilisé nos fonctions pour nos Visites sans problème.

Travail effectué :

- Laillé Clément :
 - java (classe Réservation et méthodes annulerVisite, payerVisite, reserverVisite)
 - tests (annulerVisite et payerVisite)
 - wsdl serveur
 - processus pour payer une visite (servlet, jsp)
- Le Borgne Clément :
 - java (classe Visite et méthode trouverVisites)

- tests (AllTests, trouverVisite et reserverVisite)
 - wsdl client
 - processus pour annuler une visite (servlet, jsp)
 - Complétion de ce ReadMe
- Tourneur Virgil :
 - Menu et réserver, trouver, annuler, payer (html)
 - Processus pour trouver une visite (servlet, jsp)
 - Mise en forme du CSS des pages web
 - Debug côté client
 - Affichage dynamique des villes, des types de visite et des prix en fonction de la base de données lors de la recherche d'une visite
- Vezin Frédéric :
 - Login (html et servlet)
 - Processus de réservation visite (servlet, jsp)
 - Complétion automatique du champ 'codeVisite' lors d'une réservation depuis la liste des visites
 - Structure et Mise en place base de données (+ .pdf et .sql)
 - Début de complétion et mise en page de ce ReadMe