|  |
| --- |
| Projet langage C – Réservation Hôtelière |
| Algorithmes et RFC |
| FOLIO Brice – MAYODE Hervé – LERAYER Adrien |

12/01/2014

SOMMAIRE

[SUJET 2](#_Toc377224871)

[RFC 3](#_Toc377224872)

[I. Résumé 3](#_Toc377224873)

[II. Fonctionnement général 3](#_Toc377224874)

[III. Fonctions 4](#_Toc377224875)

[III.1. Fonctions Client 4](#_Toc377224876)

[III.2. Fonctions serveur 5](#_Toc377224877)

[IV. Sécurité 5](#_Toc377224878)

[Algorithmes 6](#_Toc377224879)

# SUJET

Le but de ce projet est de développer un système simplifié de réservations de nuitées d’hôtel dans des villes touristiques. Plus particulièrement, une centrale va jouer un rôle intermédiaire de mise en correspondance de demandes des clients et de disponibilités offertes par les hôtels. Ainsi, une centrale va donner :

·      La possibilité à chaque hôtel qui est intéressé par ses services de s’enregistrer tout en mentionnant en particulier sa catégorie (un chiffre correspondant à son nombre d’étoiles) ainsi que le nombre de chambres disponibles.

·      À des acteurs touristiques (voyageurs, tour-opérateurs, agences de voyages, particuliers, etc…) de déposer des demandes de réservation.

Le système implémentant le rôle de la centrale tentera de traiter automatiquement du mieux possible les demandes de réservation en fonction des disponibilités des hôtels abonnés.

Toute réservation devra préciser :

·      Le nom de la ville ;

·      La date de la nuitée ;

·      Le code de la catégorie de l’hôtel souhaitée ;

Pour traiter une telle demande, la centrale doit essayer de trouver, parmi tous les hôtels abonnés de la catégorie concernée, un qui peut lui garantir la réservation.

À l’issue du traitement d’une commande, elle doit être capable d’indiquer :

·      Soit le fait que la réservation a été honorée ;

·      Soit qu’aucune réservation n’est possible (tous les hôtels sont complets à la date choisie).

# RFC

## Résumé

La RFC présente décrit les annonces ainsi que les phases de négociation entre les différents clients et le serveur afin d’effectuer les actions suivantes :

* Effectuer une réservation.
* Enregistrer son hôtel dans la base de fichiers du serveur.

Comme indiqué ci-dessus l’application fonctionnera sous une architecture client-serveur.

Le langage utilisé est le C.

Nous avons choisi de stocker les informations sous forme de fichier et non en base de données car il existe quelques problèmes d’interprétation avec le langage SQL, ainsi que des difficultés de portabilité.

## Fonctionnement général

Un client distant pourra se connecter au serveur à l’aide d’un poste quelconque.

Ensuite lors de l’authentification le client se verra attribuer différents attributs.

Le client pourra être :

* Soit un acteur touristique.
* Soit un client.

Suite à cela, en fonction des attributs le serveur mettra à disposition du client des services qui lui sera propre en fonction de ses attributs.

Cela implique qu’il y a plusieurs catégories de client et qu’il y a donc des éléments de sécurité à mettre en place qui seront décrites plus tard.

Lorsque le client aura effectué ces opérations, toutes les informations renseignées par celui-ci seront stockées dans des fichiers.

Une fois que le client aura quitté l’application la liaison sera rompu entre le poste client et le serveur jusqu’à la prochaine connexion.

## Fonctions

### III.1. Fonctions Client

Ci-dessous nous retrouvons l’ensemble des principales fonctions utilisées par le client dont nous aurons besoin dans notre application :

* Connecter

Cette fonction permettra au poste client d’établir une liaison avec le poste serveur.

* Déconnecter

Cette fonction permettra au poste client de rompre la liaison avec le poste serveur.

* Authentifier

Cette fonction permettra au poste client d’envoyer les informations d’authentification afin que le serveur sache à quels services il aura droit.

* Desauthentifier

Cette fonction va permettre à un gérant d’hôtel de se connecter en tant que client afin de vérifier si son hôtel est bien présent dans la liste des hôtels de l’application.

* ConsulterParCategorie

Cette fonction permettra au client de consulter les différents hôtels répertoriés dans la base de fichier du serveur en fonction de leur catégorie.

* ConsulterParNom

Cette fonction permettra au client de consulter les différents hôtels répertoriés dans la base de fichier du serveur selon leur nom.

* ConsulterParLocalisation

Cette fonction permettra au client de consulter les différents hôtels répertoriés dans la base de fichier du serveur selon leur localisation.

* Réserver

Cette fonction permettra au client d’indiquer au serveur qu’il souhaite réserver une nuit dans l’hôtel sélectionné.

* EnregistrerHotel

Cette fonction permettra aux gérants d’hôtels de saisir les informations concernant leur hôtel qui sont les suivants :

* NomHotel.
* CategorieHotel.
* NombreChambre.
* NuiteeDisponible.
* DescriptionHotel.
* Editer

Cette fonction va permettre à un gérant de modifier les informations relatives à son hôtel.

* ConsulterHotels

Cette fonction va permettre au poste client de demander la liste des hôtels au poste serveur.

## III.2. Fonctions serveur

Les fonctions utilisées par le serveur sont les suivantes :

* Autoriser

Cette fonction autorisera le poste client à accéder aux différents services du poste serveur, le serveur vérifiera les informations d’authentifications envoyées par le client et celles se trouvant dans le fichier Authentification.txt.

* Refuser

Cette fonction bloquera l’accès aux services du poste serveur.

* ConsulterListeHotel

Cette fonction va lire l’ensemble des hôtels référencés dans le fichier Hotel.txt et les renvoyer. Un tri s’effectuera à l’aide de fonctions internes du langage C.

* EnregistrerReservation

Cette fonction sauvegardera les informations de réservations dans un fichier appelé Reservations.txt.

* EnregistrerNouvelHotel

Cette fonction enregistrera les informations liées à l’inscription d’un gérant d’hôtel dans le fichier Hotel.txt.

* EnregistrerModificationHotel

Cette fonction va permettre au serveur d’enregistrer les modifications effectuées par un gérant.

* Envoyer

Cette fonction va servir à envoyer des requêtes contenant les informations.

* Recevoir

Cette fonction va analyser les requêtes reçues afin d’y extraire les informations nécessaires.

## IV. Sécurité

Il y a différents éléments de sécurité à prendre en compte qui sont les suivants :

* Le nombre d’accès.

On limitera le nombre de connexion au poste serveur afin d’éviter le flooding.

* La multi-connexion.

Un même acteur touristique ne pourra pas se connecter sur deux postes simultanément.

* L’authentification.

Le gérant d’un hôtel devra au préalable appeler le service afin qu’on lui attribue un identifiant afin d’éviter des faux hôtels et aussi qu’il y ait un seul responsable par hôtel.

# ALGORITHMES





