

# Protocole client-serveur de Zoltan

## 1. Ouvrir une session

### 1.1. Authentification

Le Client se connecte au serveur et envoi son login:

L:Login

et attend la réponse du serveur. La réponse du serveur est [serveurZoltan] LoginOk si le login existe dans la base de données du serveur.  
Sinon la connexion est annulée.

Ensuite le client envoi le mot de passe:

P:Password

et attend la réponse du serveur. La réponse du serveur est [serveurZoltan] PassOk si le mot de passe existe dans la base de données du serveur.  
Sinon la connexion est annulée.

### 1.2. Envoi de liste de contact

Après l'authentification, le client demande sa liste de contact en envoyant:

UpdateContactList

Le serveur envoi la liste de contact:

R:login:adress:port:connected:status: (1 personne dans la liste)

R:login1:adress1:port1:connected1:status1|login2:adress2:port2:connected2:status2:

(chaque personne de la liste est séparés par | )

Si le client ne possède aucun contact dans sa liste alors, le serveur envoi:

R:

Lorsque le client reçoit sa liste, il renvoi au serveur:

ZListOk

### 1.3.Demande d'autorisation

Après avoir reçu **ZListOk**, le serveur regarde si le client a eu des demandes d'ajout de contact. Si il en trouve le serveur envoi:

**\*:login**  
(nom du contact qui veut être ajouter)

Le client reçoit la demande et il a le choix autoriser l'ajout du contact dans sa liste, suivant sa réponse, il renvoi au serveur:

-Si sa réponse est positive:

**A:login**

-Le serveur renvoi dans ce cas:

**[serveurZoltan] Votre login a été ajoute à la liste de contact de client**  
et la nouvelle liste de contact mis a jour comme expliqué dans 1.2

-Par contre le serveur envoi au client (connecté) qui a demander l'ajout :

**[serveurZoltan] login a accepté de vous ajouté dans sa liste.**  
et la nouvelle liste de contact mis a jour comme expliqué dans 1.2

-Si sa réponse est négative :

**N:login**

Le serveur renvoi dans ce cas au client qui a refusé l'ajout:

**[serveurZoltan] Vous avez refusé l'ajout de client**

Par contre le serveur envoi au client (connecté) qui a demander l'ajout :

**[serveurZoltan] login a refusé de vous ajouter dans sa liste**

L'opération 1.3 est recommencée pour chaque utilisateur qui veut être ajouté à la liste de contact.

## 2. Gestion des utilisateurs

### 2.1. Ajout de Contact

Le Client qui veut ajouter un contact envoie au serveur :

**C:login** (login est le nom du contact à ajouter)

Le Serveur envoie un message au client:

Si le login existe déjà dans sa liste de contact :

[serveurZoltan] login a déjà été ajouté à votre liste de contact

Si le login n'existe pas :

[serveurZoltan] Le contact que vous voulez ajouter n'existe pas

Si le login existe et qu'il n'est pas déjà dans sa liste :

Le serveur regarde si le client a déjà effectué cette demande d'ajout :

Si oui, il envoie :

[serveurZoltan] La demande d'autoriser l'ajout de login a déjà été enregistrée sur le serveur

Sinon, il envoie :

[serveurZoltan] La demande d'autoriser l'ajout de login a été enregistré sur le serveur, celui-ci pourra l'autoriser lorsqu'il sera connecté à Zserveur

### 2.2. Supprimer un Contact

Lorsque le client veut supprimer un contact, il envoie au serveur :

**S:login**

Le Serveur envoie un message au client :

-Si le contact qu'il veut supprimer n'existe pas dans sa liste il envoie :

[serveurZoltan] login a déjà été supprimé de votre liste de contact

-Si le contact existe :

[serveurZoltan] login a été supprimé de la liste de contact

Le serveur envoie la nouvelle liste de contact de login comme expliqué dans 1.2.

### 2.3. Statut

Le Client qui modifie son statut envoi au serveur :

**I:Statut** (par défaut le client a le choix: En Ligne,Occupé,Travail,Absent)

Le serveur informe les clients (connectés) de sa liste de contact en leur envoyant leurs nouvelles listes de contact, comme expliqué dans le 1.2

### 2.4. Recherche de Contacts

Lorsque le client recherche un contact, il envoi soit:

-Le nom du contact recherché :

**R:contact**

Le serveur renvoi le contact recherché :

**T:login1:ip1:port1:connected1:status1:**

-Une ou plusieurs lettres :

**R:co**

Le serveur renvoi les contacts commençant par les lettres donnés :

**T:login1:ip1:port1:connected1:status1:login2:ip2:port2:connected2:status2:**

(...)

-Aucun contact :

**R:**

Le serveur renvoi la liste de tous les utilisateurs du logiciel :

**T:login1:ip1:port1:connected1:status1:login2:ip2:port2:connected2:status2:(...)**

### 2.5. Déconnexion d'un client

Lorsqu'un utilisateur se déconnecte, le Serveur envoi l'information a ses contacts qui sont encore connectés.

**-:login**

### 2.6. Partage de fichiers

Un client peut partager une liste de fichier pour chaque personne de sa liste de contact ;  
Pourdemander d'envoyer la liste des fichiers partagés:

**G:?**

Pour envoyer la liste de fichier:

**G:fichier1|fichier2|fichier3|**

note:seul le nom du fichier est envoye.

## 3. Chat

### 3.1.Global

Lorsque le client discute dans le chat global, il envoie au serveur:

M:message

Le serveur le reçoit et récupère le login de la personne qui envoie le message, et ajoute des crochets autour de son login et le place devant le message et puis l'envoie à tous les clients connectés :

[login]message

### 3.2.Privé

Lorsque le client discute dans le chat privé, il envoie à un autre client :

M:<login>message

## 4. Creation de compte

### 4.1.Demande de creation de compte

Lorsque le client demande au serveur de lui créer un compte, il envoie :

Z:login|pass

Le serveur vérifie si le login existe dans la liste des utilisateurs.

Si le login est présent, il renvoie au client:

Z:Le login est déjà utilisé par un autre utilisateur

Le serveur vérifie si le login ne contient pas de caractères spéciaux :

Si il en contient un il renvoie au client:

Z:Votre login contient des caractères non valide

Le serveur vérifie si le nombre de caractères du login n'est pas compris entre 2-16 :

Z:le nombre de caractères de votre login doit être compris entre 2-16

Une fois que le serveur a créé le compte, il renvoie :

Z:Votre compte a été créé

## 5. Envoi de fichier

Demande d'envoi de flux.

**Y:filePath|Size|pos**

*filePath* contient le nom complet (avec le chemin) du fichier present sur l'emeteur.

Demande de mise en pause

**W:filePath**

Demende d'envoi de fichier

**O:fileName**

Envoi des infos precedent l'envoi du flux

**F:FilePath|Size|Pos**