



2.2 TP Ethernet Serveur

Prés requis :

Ouvrir un projet java avec Eclipse

La communication Ethernet est un mode connecté (socket / prises), ici nous utilisons des flux de communications (stream) en entrée et en sortie (InputStream / OutputStream comme en bluetooth), le protocole de communication est le protocol TCP/IP (Transfert contrôle protocole sur IP (Internet Protocole))

Le protocole de communication nécessite pour initier le dialogue de créer une prise de communication (socket) par conséquent d'utiliser un port de communication ainsi que l'adresse IP du serveur.

Ce protocole de communication est différent du protocole UDP (mode non connecté) qui ne nécessite pas de connaître l'adresse du poste distant (utilisation du broadcasting pour récupérer l'adresse distante).

a) On crée un nouvel objet de connexion de type ServerSocket en précisant le port de communication qu'utilisera le client.

Serveur = new ServerSocket(1234);

b) le serveur se place en attente d'acceptation d'une connexion client et nous retourne un socket sur le client.

Client = Serveur.accept();



c) On crée un flux de sortie sur le socket client pour envoyer des données à celui-ci

FluxSortieEthernet = Client.getOutputStream()

d) On crée un flux d'entrée sur ce socket client pour recevoir des données de celui-ci

FluxEntreeEthernet = Client.getInputStream();

e) Il est donc possible ensuite de lire des tableaux de bytes en provenance du client et de connaître le nombre de bytes reçus

intNbreCarLusEthernet = FluxEntreeEthernet.read(byteTabRequeteRecue);

f) Il est aussi possible d'envoyer des données (tableau de bytes) au client, par exemple des accusés de réception.

FluxSortieEthernet.write(strAccuseReception.getBytes());



2.2.1 Créer un Serveur Ethernet en Java sur PC avec Eclipse

- a) Vous ouvrirez le projet java (**Serveur_Java_TCP_Socket_v1.0ac**) avec l'IDE Eclipse
- b) Vous compléterez le programme avec les informations précédentes
- c) Ici pas besoin d'ajouter de librairies (pas comme en bluetooth)
- d) Vous exécuterez le programme. Si tout se passe correctement vous devriez obtenir ceci.

```
-----  
#                                     Serveur TEST TCP  
#  
# Auteur : Wilfrid Grassi  
# Date : 25/11/2017  
# Version : 1.0  
-----  
  
Activer les accusés de réceptions des requêtes TCP du Client (bidirectionnels) O/N : |
```

Répondez oui (o), cela permettra à votre serveur d'envoyer les accusés de réception vers le client.



```
Graphical Layout  Charged_de_sauvegarde | Problems @ Javadoc Declaration Console
Serveur_Java_TCP_Socket [Java Application] C:\Program Files\Java\jre1.8.0_151\bin\javaw.exe (19 déc. 2017 22:13:35)
-----
#          Serveur TEST TCP
#
# Auteur : Wilfrid Grassi
# Date : 25/11/2017
# Version : 1.0
-----

Activer les accusés de réceptions des requêtes TCP du Client (bidirectionnels) O/N :
o Accusés de réception active.
.....
#
#   Interface(s) réseau et adresse(s) disponible(s) sur ce serveur :
#     [ Carte réseau : Qualcomm Atheros AR946x Wireless Network Adapter : 192.168.1.34 ]
#     [ Carte réseau : VirtualBox Host-Only Ethernet Adapter #2 : 192.168.56.1 ]
#     Port d'écoute : [ 1234 ]
#
#   Serveur en attente de connexion Client...
```

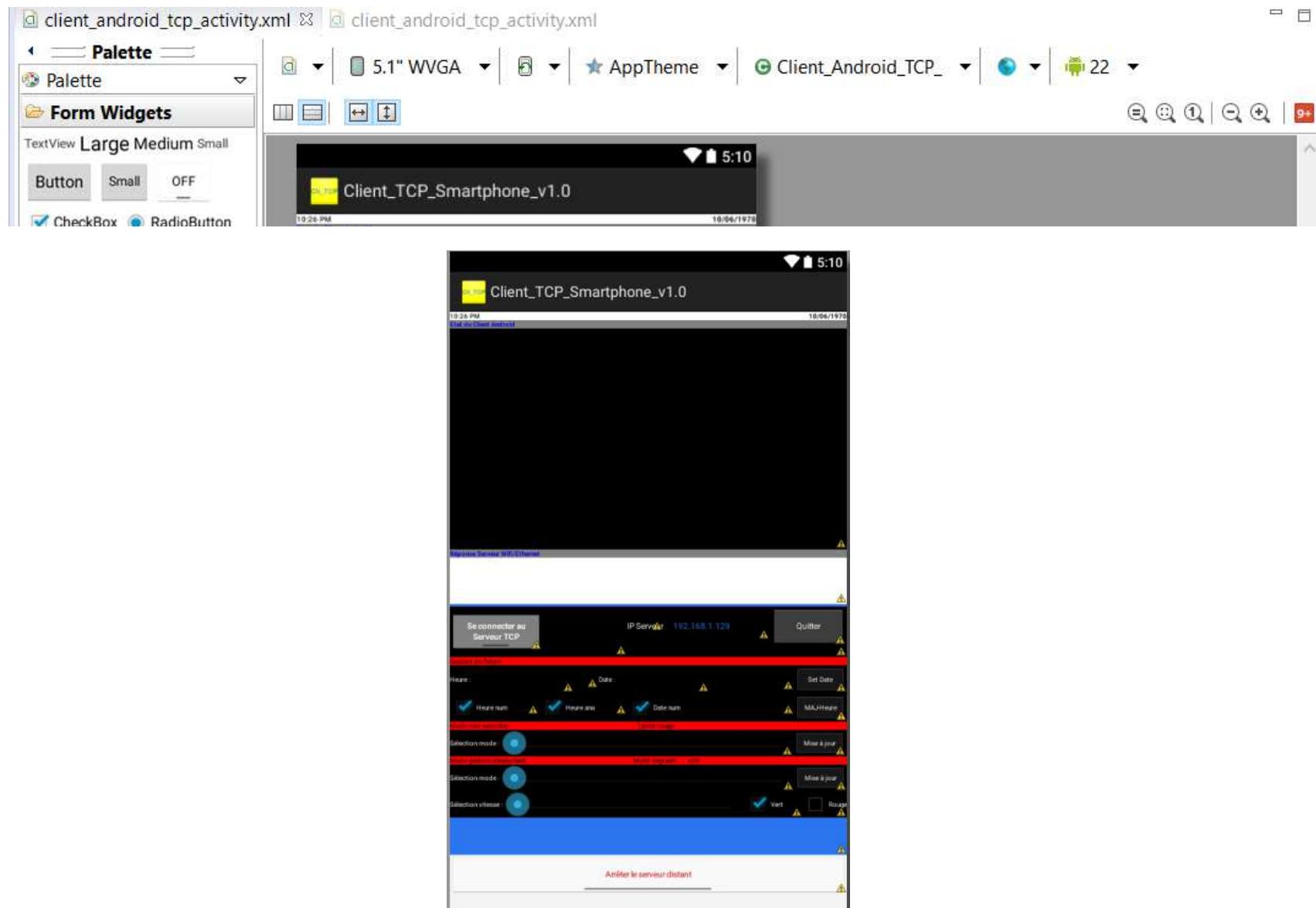
2.2.2 Connecter le serveur Ethernet et un client Ethernet Android

a) Importer les exemples de clients Android Smartphone et tablette

"Client_Android_TCP_Socket_Smartphone_v1.0ac"

"Client_Android_TCP_Socket_Tablette_v1.0ac"

Vérifier les informations d'API (22), AppTheme, 7"WSVGA





- b) Télécharger l'application sur la tablette au moyen de la connexion USB**
- c) Déconnecter la tablette de l'USB**
- d) Exécuter le programme Client sur la tablette**
- e) Etablir une connexion avec le PC (le serveur doit être en fonctionnement) en entrant l'adresse IP du serveur sur l'application tablette**

2.2.3 Exécuter le serveur Ethernet Java sur Raspberry

Pour cela vous utiliserez le partage réseau de votre raspberry et copierez le dossier <Serveur_Java_TCP_Socket_v1.0ac> dans le dossier <Pi3-xx/Pi/Document> puis vous utiliserez le script "StartX_Serveur_Java_TCP_Socket.sh" pour lancer votre programme serveur java

2.2.4 Connecter le serveur Ethernet Raspberry avec le client Android