Travaux Pratique programmation réseau avec QT5

1. OBJECTIF

Utilisation des sockets avec la bibliothèque QT5

Mettre en pratique les sockets en mode événementiel. Utiliser cette propriété pour réaliser un serveur multi-Clients.

Points techniques abordés :

- Codage de l'information
- Programmation sockets avec QT5
- Communication réseau

2. CONDITIONS DE RÉALISATION

Travail individuel sur micro-ordinateurs avec QT Creator

Un serveur et un client sont disponibles pour la mise au point.

Compte rendu et code à rendre.

3. RESSOURCES

Les classes « socket » dans la technologie QT, consulter le site http://doc.qt.io/qt-5/qtnetwork-index.html et plus particulièrement les classes, ***QTcpSocket, QTcpServeur.***

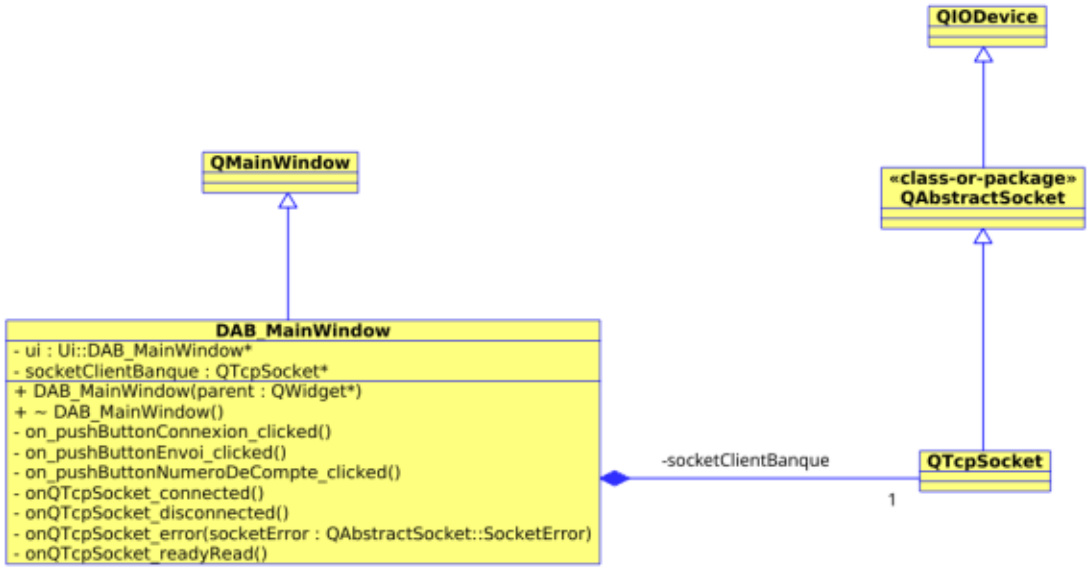
4. LE BESOIN

Les distributeurs automatiques de billets (DAB) permettent de se connecter à la banque. Ils offrent la possibilité d'interroger un compte pour en connaître le solde, de déposer de l'argent et d'en retirer. C'est ces fonctionnalités que nous allons étudier dans un premier temps côté client (sur le DAB) puis côté serveur (à la banque).

5. BANQUE ~ CÔTÉ CLIENT

5.1. Création du projet

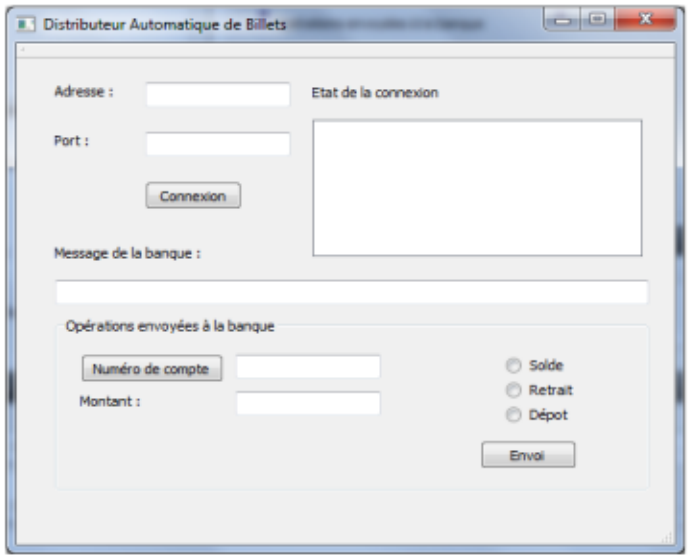
Créez le projet ***ClientBanque*** de type Application graphique en C++ sous QT5 avec QT Creator. La classe principale ce nomme ***DAB_MainWindow*** elle hérite de ***QMainWindow*** comme le montre le diagramme de classes partiel ci-dessous. C'est la seule classe où vous aurez à intervenir.



De même, ajoutez l'attribut réalisant la liaison avec la classe ***QTcpSocket***.

5.2. Création de l'IHM

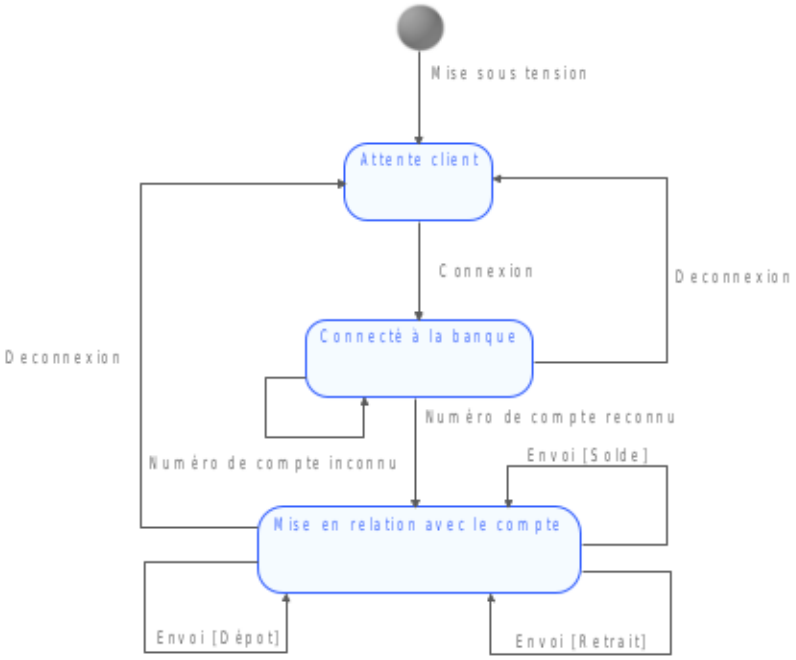
L'interface du client aura l'aspect suivant :



1. Nommez chaque ***Widget*** avec la convention de nommage habituelle.

Tant que la connexion avec la banque n'est pas établie, les opérations vers la banque ne sont pas possibles (les boutons sont désactivés), la zone d'édition pour les messages de la banque est en lecture seule. Le ***listWidget*** contenant l'état de la connexion avec la banque indique les différents états de la connexion et les messages d'erreur système. Le bouton Connexion devient Déconnexion lorsque la connexion est établie.

Fonctionnalités attendues du distributeur automatique de billets :



Comme l'indique le diagramme à états ci-dessus, c'est seulement lorsque le client est connecté à la banque et que son numéro de compte est reconnu que les opérations vers la banque peuvent être réalisées (par sécurité, les opérations sont désactivées sur l'IHM). La déconnexion reste possible à chaque état.

5.3. Utilisation de la classe QTcpSocket

Après l'initialisation de l'attribut représentant le lien entre les classes ***DAB_MainWindow*** et ***QTcpSocket***, reliez les signaux en provenance du socket client, la connexion au serveur, sa déconnexion et la présence de données à lire à des slots chargés du traitement correspondant dans la classe gérant l'IHM (respectez la convention de nommage vue en cour).

Chaque connexion entre signaux et slots doit faire l'objet d'un test avec l'apparition d'une boîte de message indiquant l'erreur et fermant l'application.

L'état de la connexion avec le serveur de la banque est reporté dans le ***listWidget*** prévu à cet effet.

La trame à transmettre pour fournir le numéro de compte se décompose ainsi :

Taille des données	Commande	Valeur
De type quint16	Le caractère 'N' sous la forme d'un Qchar	Un entier correspondant au numéro de compte (type int)

Pour forger cette trame, nous utiliserons un QBuffer et un QDataStream.

Voici un exemple façon de procéder:

```
    quint16 taille=0;
    QChar commande='N';
    int compte=1234;
    QBuffer tampon;
    tampon.open(QIODevice::WriteOnly);
    // association du tampon au flux de sortie
    QDataStream out(&tampon);
    // construction de la trame
    out<<taille<<commande<<compte;
    // calcul de la taille de la trame
    taille=tampon.size()-sizeof(taille);
    // placement sur la premiere position du flux pour pouvoir modifier la taille
    tampon.seek(0);
    //modification de la trame avec la taille reel de la trame
    out<<taille;
    // envoi du QByteArray du tampon via la socket
    socketClientBanque->write(tampon.buffer());
```

1. Réalisez le code nécessaire pour l'envoi du numéro de compte vers la banque le numéro est transmis s'il est supérieur à 0.

Tous les messages en provenance de la banque sont de la forme :

Taille des données	Commande
De type quint16	Une chaîne de caractères sous la forme d'une QString.

Comme il n'est pas possible de connaître à l'avance le nombre d'octets que nous recevrons en même temps, nous devrons procéder par étape en utilisant les flux.

Voici un exemple de traitement de la réponse de la banque:

```
    quint16 taille=0;
    QString message;
// si le nombre d'octets recu est au moins egal a celui de la taille de ce que
l'on doit recevoir
    if (socketClientBanque->bytesAvailable() >= (qint64)sizeof(taille))
    {
// association de la socket au flux d'entree
        QDataStream in(socketClientBanque);
// extraire la taille de ce que l'on doit recevoir
        in >> taille;
// si le nombre d'octets recu est au moins egal a celui de ce que l'on doit
recevoir
        if (socketClientBanque->bytesAvailable() >= (qint64)taille)
        {
            // extraire le message de la banque et le mettre dans message
            in>>message;
        }
    }
```

2. Réalisez le code nécessaire pour recevoir la réponse de la banque et l'afficher dans la zone d'édition prévue à cet effet.

Opérations vers la banque :

Lors de l'affichage de l'IHM du distributeur, on souhaite que le Bouton radio Solde soit coché par défaut.

3. Réalisez le code nécessaire dans la méthode qui vous semble la plus appropriée.

L'appui sur le bouton Envoi doit envoyer à travers le socket le type d'opération souhaité, suivi éventuellement du montant de la transaction suivant le format de trame :

Taille des données	**Opération**	**Montant**
De type quint16	Le caractère 'R' sous la forme d'un Qchar	Un réel (type float)
De type quint16	Le caractère 'D' sous la forme d'un Qchar	Un réel (type float)
De type quint16	Le caractère 'S' sous la forme d'un Qchar	Sans objet

Pour la réponse de la banque si tout va bien vous n'avez rien à faire…