



Université du Québec  
à Trois-Rivières

Projet de Session

PRÉSENTÉ À

M BOUCIF AMAR BENSABER

COURS

RÉSEAUX D'ORDINATEURS 1

PAR

**GANSONRE ISMAEL**

*SEMANOU RICHAIR,*

*DIALLO MARIAMA, DIALLO*

*FATIMA,*

*AUDREY BONI*

*ET RAMI BERLAT*

25 AVRIL 2022

## Contents

Description du logiciel .....	3
Instructions spéciales .....	3
Description des fichiers. ....	3
S_lec : .....	3
S_ecr : .....	3
L_ecr : .....	4
L_lec : .....	4
Description des classes.....	4
Les classes de paquet .....	4
La classe Constantes .....	4
La classe ConnexionTransport .....	5
La classe transport .....	5
La classe EntitéeReseau .....	5

## Description du logiciel

Notre logiciel se veut être une simulation de connexion et de transfert de données vers un ordinateur distant. Le système local est divisé en une couche transport et une couche réseau. La liaison de données et l'ordinateur distant sont simulés complètement par la couche réseau. Ces composantes seront expliquées plus en détail ultérieurement. Chacune de ces couches communique avec la couche qui lui est supérieure et la couche qui lui est inférieure. La couche transport lit une liste d'instruction dans un fichier, les passe ensuite à des threads de la couche réseau qui fabriquent des paquets et les envoient au distant qui est en fait simulé par réseau. Distant prend les paquets, inscrit la donnée reçue dans un fichier et répond adéquatement à la couche réseau, en plus de garder un log des transactions. Réseau décide ensuite de conserver ou fermer les connexions avec le distant au besoin, et indique à transport lorsqu'une connexion est fermée. Transport note aussi les ouvertures et fermetures de connexion dans un fichier.

## Instructions spéciales

Il y a peu d'instructions spéciales pour exécuter le programme, il suffit de regarder le terminal pour avoir une confirmation de chaque pression sur le bouton et de le lancer à partir de la classe main et de choisir l'option demo file qui va se charger dans la vue S\_lec puis d'appuyer sur le bouton Commencer pour tout se faire tout seul. On peut ensuite constater ce qui s'est passé à l'aide des différents fichiers et de leur vue dans l'interface.

Il est normal que le programme prenne une dizaine de secondes à s'effectuer, car il doit parfois attendre qu'un thread ne finisse.

Les fichiers textes sont situés sous :

```
src/Couche/Session/S_lec.txt ;Couche/Session/S_ecr.txt;Couche/LiaisonDonnee/L_ecr.txt;Couche/LiaisonDonnee/L_lec.txt
```

## Description des fichiers.

Les différents fichiers textes que nous utilisons sont les suivants :

S\_lec :

Contient les différentes instructions que nous souhaitons envoyer au distant.

S\_ecr :

Contient les logs de la couche transport. Les trames sont encore une fois séparées en deux, commençant par le nom du programme, suivi du log en tant que tel.

Les deux opérations que note transport sont création d'une connexion, et terminaison ou encore échec d'une connexion.

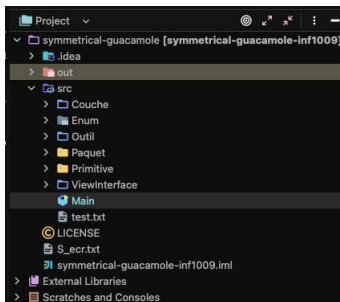
L\_ocr :

Ce fichier correspond à ce que reçoit l'application sur la machine distante, Il n'inscrit que le contenu des paquets de data reçus, précédés du numéro de la connexion qui les a envoyés.

L\_lec :

Ce fichier correspond au fichier log de la liaison de données. Le distant note toutes les transactions dans ce fichier. Il commence encore une fois par le type de paquet suivit par le numéro de la connexion, et de la raison.

## Vue d'ensemble du projet



## Description des classes

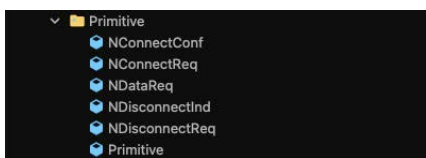
### Les classes de paquet

Tous les types de paquet sont représentés par une classe distincte qui hérite de la classe générale Paquet se se trouvent dans le dossier Paquet



### La classe des Primitives

La classe Primitives contient toutes les constantes nécessaires au bon fonctionnement du programme et se trouvent dans le dossier Primitives



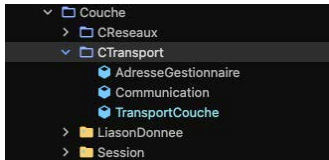
La classe main

Le programme  
commence par  
L'ouverture de  
l'interface  
utilisateur

Le dossier CTransport

Il contient les classes

nécessaires pour le transport



class AdresseGestionnaire

permet de gestion des adresse

class Communication

qui presenter une

communication entre deux

extremite,

les deux extremite peuvent

etre deja connecte ou non

connecte.

```
import Couche.CTransport.TransportCouche;
import ViewInterface.guiix;
/* Gansorre Ismael */
public class Main {
    /* Gansorre Ismael */
    public static void main(String[] args) {
        javax.swing.SwingUtilities.invokeLater(() -> {
            new guiix().setVisible(true);
            TransportCouche transport = new TransportCouche();
            transport.DemarrerCommunication();
        });
    }
}
```

