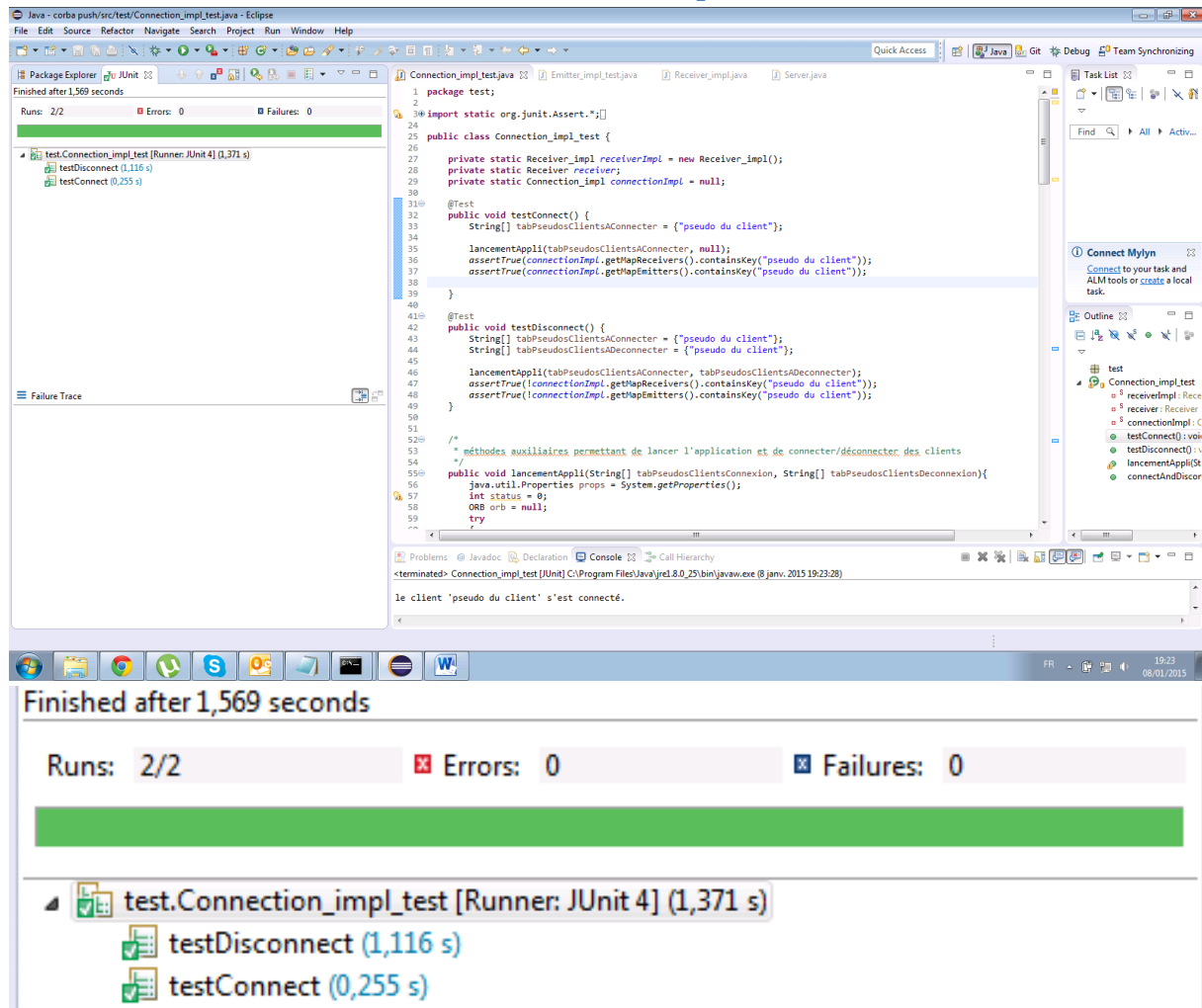


Projet CORBA

Rapport de couverture des tests unitaires (partie PUSH)

Classe servante Connection : Connection_impl



The screenshot displays the Eclipse IDE interface with the following components:

- Package Explorer:** Shows the project structure with the test suite `test.Connection_impl_test` expanded, listing `testDisconnect` (1,116 s) and `testConnect` (0,255 s).
- JUnit Console:** Displays the test results: "Finished after 1,569 seconds", "Runs: 2/2", "Errors: 0", and "Failures: 0".
- Source Editor:** Shows the `Connection_impl_test.java` file with the following code:

```
1 package test;
2
3 import static org.junit.Assert.*;
4
5 public class Connection_impl_test {
6
7     private static Receiver_impl receiverImpl = new Receiver_impl();
8     private static Receiver receiver;
9     private static Connection_impl connectionImpl = null;
10
11     @Test
12     public void testConnect() {
13         String[] tabPseudosClientsAConnecter = {"pseudo du client"};
14
15         lancementAppli(tabPseudosClientsAConnecter, null);
16         assertTrue(connectionImpl.getMapReceivers().containsKey("pseudo du client"));
17         assertTrue(connectionImpl.getMapEmitters().containsKey("pseudo du client"));
18     }
19
20     @Test
21     public void testDisconnect() {
22         String[] tabPseudosClientsAConnecter = {"pseudo du client"};
23         String[] tabPseudosClientsADeconnecter = {"pseudo du client"};
24
25         lancementAppli(tabPseudosClientsAConnecter, tabPseudosClientsADeconnecter);
26         assertTrue(connectionImpl.getMapReceivers().containsKey("pseudo du client"));
27         assertTrue(connectionImpl.getMapEmitters().containsKey("pseudo du client"));
28     }
29
30     /*
31      * méthodes auxiliaires permettant de lancer l'application et de connecter/déconnecter des clients
32      */
33     public void lancementAppli(String[] tabPseudosClientsConnexion, String[] tabPseudosClientsDeconnexion) {
34         java.util.Properties props = System.getProperties();
35         int status = 0;
36         ORB orb = null;
37         try {
38             // ...
39         } catch (Exception e) {
40             // ...
41         }
42     }
43 }
```
- Outline:** Shows the class hierarchy and methods, including `testDisconnect` and `testConnect`.
- Problems:** Shows a message: "le client 'pseudo du client' s'est connecté."

Below the IDE screenshot, a summary of the test results is provided:

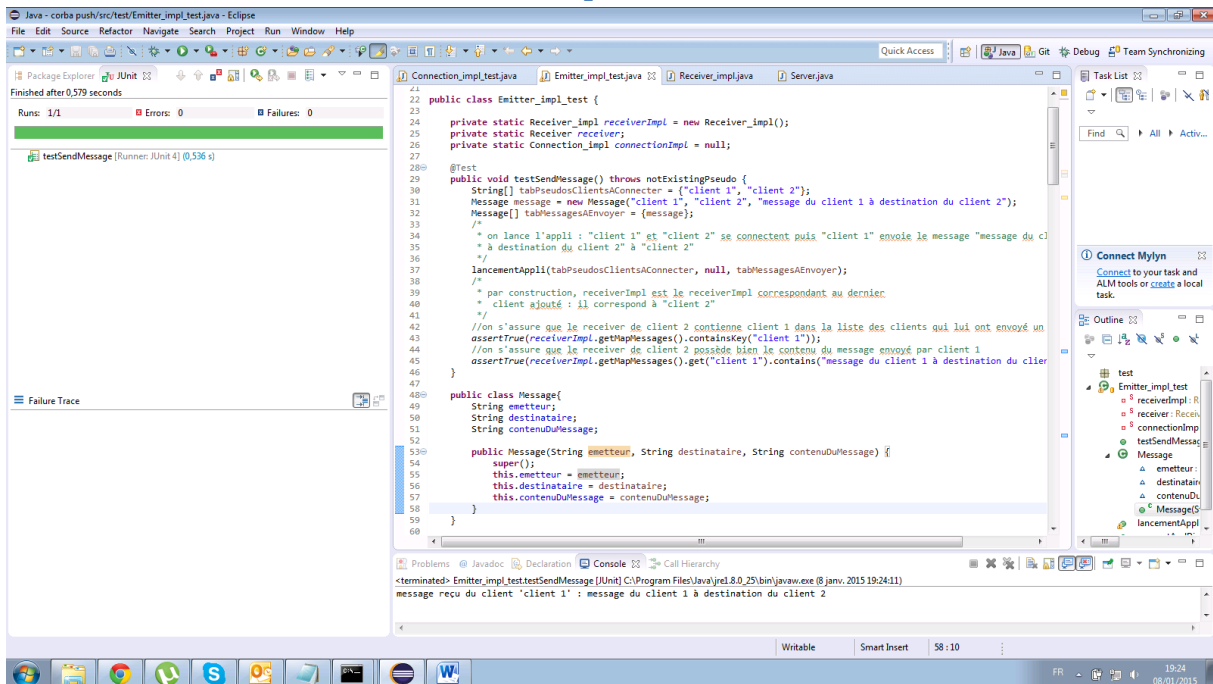
Finished after 1,569 seconds

Runs: 2/2 Errors: 0 Failures: 0

test.Connection_impl_test [Runner: JUnit 4] (1,371 s)

- testDisconnect (1,116 s)
- testConnect (0,255 s)

Classe servante Emitter : Emitter_impl



Finished after 0,579 seconds

Runs: 1/1

Errors: 0

Failures: 0

testSendMessage [Runner: JUnit 4] (0,536 s)

The screenshot shows the Eclipse IDE interface during a JUnit test run. The top toolbar includes standard IDE functions like File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, and Help. The left sidebar contains the 'Package Explorer' showing the project structure, with 'test.Receiver_impl_test [Runner: JUnit 4] (0,001 s)' highlighted. The main editor window displays the source code of 'Receiver_impl_test.java', which includes tests for adding, removing, and receiving messages. The bottom status bar shows 'Runs: 4/4', 'Errors: 0', and 'Failures: 0'. A detailed view of the test results is shown at the bottom, listing 'testAddClient (0,000 s)', 'testRemClient (0,000 s)', 'testReceive (0,001 s)', and 'testInitClients (0,000 s)' with green checkmarks indicating success.

Rapport de revue de qualité du code (partie PUSH)

