

Pràctiques

A continuació trobaras la llista de **tasques optatives** per aquest tema.

Com s'ha indicat al tema anterior, el teu repositori de tasques de *Bitbucket* ha d'estar compartit amb el professor via el mail e.nicloscamarasa@edu.gva.es en aquest cas.

Per últim, i de caràcter general, **els programes cal executar-los com a fils per evitar el llançament de múltiples programes**, aleshores **el mateix programa inclourà els diferents clients i servidor**.

Qualsevol dubte, podeu consultar-me a les tutories col·lectives, a l'individual o al fòrum del tema a la plataforma Aules.

Tasca 3.4

Realitza un programa a Java (nom del paquet *Task3_04_ClonezillaServer*) que simule el comportament del ClonezillaServer amb Multicast usant fils d'execució on:

El funcionament de comunicació serà: - El fil servidor Multicast, esperarà 10 segons abans de procedir a l'enviament de les dades. El que enviarà serà una imatge del disc dur que la visualitzarem en percentatges de disc dur enviat i que ho mostrarem per pantalla. En finalitzar l'enviament es mostrarà per pantalla eixe fet i procedirà al tancament dels sockets,...

- Els fils clients Multicast (almenys 3 fils) el que faran serà subscriure's a la difusió multicast mostrant totes les dades rebudes i informant de la finalització del mateix en arribar al 100%. En acabar enviament, es mostrarà per pantalla la finalització de la recepció de dades, la cancel·lació de la subscripció i tancament dels ports.

Simula 3 clients que es connecten en els primers 10 segons previs a enviament del servidor i el tercer client, es connectarà al servidor passats ixos 10 segons inicials enviant les dades per el port que consideres.

```
mermaid
sequenceDiagram
    autonumber
    Note right of ServidorMulticast: Esperant 10 segons
    Client1->> ServidorMulticast: Subscrivint-se a IP:224.15.81.2 al port 5000
    ServidorMulticast --> Client1: Enviament: 10% imatge disc dur...
    Client2->> ServidorMulticast: Subscrivint-se a IP:224.15.81.2 al port 5000
    ServidorMulticast --> Client1: Enviament: 20% imatge disc dur...
    ServidorMulticast --> Client2: Enviament: 20% imatge disc dur...
    Client3->> ServidorMulticast: Subscrivint-se a IP:224.15.81.2 al port 5000
    ServidorMulticast --> Client1: Enviament: 20% imatge disc dur...
    ServidorMulticast --> Client2: Enviament: 20% imatge disc dur...
    ServidorMulticast --> Client3: Enviament: 20% imatge disc dur...
```

Exemple execució:

SERVIDOR	CLIENT 1	CLIENT 2
SERVIDOR MULTICAST esperant 10 segons		
	CLIENT1. Subscrivint-se a IP:224.15.81.2 al port 5000	
Enviament: 10% imatge disc dur...		CLIENT2. Subscrivint-se a IP:224.15.81.2 al port 5000

SERVIDOR	CLIENT 1	CLIENT 2
Enviament: 20% imatge disc dur...	Dades rebudes: 10% imatge disc dur...	
Enviament: 30% imatge disc dur...	Dades rebudes: 20% imatge disc dur...	
Enviament: 40% imatge disc dur...	Dades rebudes: 30% imatge disc dur...	Dades rebudes: 30% imatge disc dur...
Enviament: 50% imatge disc dur...	Dades rebudes: 40% imatge disc dur...	Dades rebudes: 40% imatge disc dur...
Enviament: 60% imatge disc dur...	Dades rebudes: 50% imatge disc dur...	Dades rebudes: 50% imatge disc dur...
Enviament: 70% imatge disc dur...	Dades rebudes: 60% imatge disc dur...	Dades rebudes: 60% imatge disc dur...
Enviament: 80% imatge disc dur...	Dades rebudes: 70% imatge disc dur...	Dades rebudes: 70% imatge disc dur...
Enviament: 90% imatge disc dur...	Dades rebudes: 80% imatge disc dur...	Dades rebudes: 80% imatge disc dur...
Enviament: 100% imatge disc dur...	Dades rebudes: 90% imatge disc dur...	Dades rebudes: 90% imatge disc dur...
Finalitzat enviament	Dades rebudes: 100% imatge disc dur...	Dades rebudes: 100% imatge disc dur...
	Dades rebudes correctament	Dades rebudes correctament
	Cancel·lant subscripció	Cancel·lant subscripció

Tasca 3.5

Implementa la versió Client/Servidor on el servidor requerirà la validació d'usuari i contrasenya per a poder accedir (nom del paquet *Task3_05_ValidacioTCP*).

El funcionament del programa serà:

- El servidor escoltarà peticions a un determinat port, per exemple el 9999 i quedarà esperant fins que el client és valide o fins que passen 30segons.
- Quan un client vol connectar-se al servidor, enviarà un primer missatge en el que s'enviarà una paraula que correspondrà a un nom d'usuari proporcionat per teclat.
- El servidor processarà l'usuari i contestarà:
 - Si l'usuari NO existeix, mostrarà per pantalla «*Usuari XXXXX NO existeix*» i sol·licitarà al client l'introducció d'un usuari per teclat.
 - Si l'usuari existeix, mostrarà per pantalla «*Usuari XXXXX existeix*» i sol·licitarà al client l'introducció d'una contrassenya.
- El client introduirà l'informació sol·licitada al punt anterior i l'enviarà al servidor el missatge format per «*PASSWD contrassenya*»
- Si les dades son correctes, el servidor contestarà al client amb el missatge «*Usuari XXXXX connectat correctament*» i mostrarà per pantalla «*Usuari XXXXX validat correctament*» però si la contrassenya és incorrecta, mostrarà per pantalla «*Usuari XXXXX no validat, contrassenya incorrecta*» i respondrà al client amb el missatge «*Usuari XXXXX, login incorrecte*»

Considera que l'unic usuari vàlid és aquell en el que el nom usuari és *laura* i la contrasenya vàlida és *l123*.*

Exemple execució:

SERVIDOR TCP	CLIENT TCP 1	CLIENT TCP 2
[SERVIDOR] escoltant al port 9999		
	[CLIENT] Port del servidor: 9999	
	[CLIENT]Usuari a enviar al servidor: <i>pepe</i>	[CLIENT] Port del servidor: 9999
[SERVIDOR]Usuari pepe NO existeix		[CLIENT]Usuari a enviar al servidor: <i>laura</i>

SERVIDOR TCP	CLIENT TCP 1	CLIENT TCP 2
[SERVIDOR]Usuari laura existeix	[CLIENT]Usuari pepe NO existeix	
	[CLIENT]Proporciona un usuari vàlid: <i>laura</i>	[CLIENT]Usuari laura existeix
[SERVIDOR]Usuari laura existeix	[CLIENT]Usuari laura existeix	[CLIENT]Contrassenya per usuari laura: /123
[SERVIDOR]Usuari laura validat correctament	[CLIENT]Contrassenya per usuari laura: /223	
[SERVIDOR]Usuari laura no validat, contrassenya incorrecta		[CLIENT]Usuari laura connectat correctament
	[CLIENT]Usuari laura, login incorrecte	