

Progettazione del Software

Prova Pratica (A)

12/02/2021 - tempo a disposizione 1h30m

Si vuole realizzare un'applicazione *client-server* per il monitoraggio di una serie di macchine virtuali (o VM) predefinite.

Un server gestisce le macchine virtuali e ne monitora le informazioni relative all'utilizzo di CPU, RAM, disco e rete al fine di renderle disponibili per i client, i quali possono connettersi al server e visualizzarle all'utente in tempo reale. Il client permette, tramite l'interfaccia grafica, la selezione della macchina virtuale di interesse da un insieme predefinito da parte dell'utente e l'invio della richiesta delle informazioni al server, la ricezione, e la visualizzazione di tali informazioni tramite delle componenti grafiche. Il server è *multithreading* ed accetta connessioni da parte di più client. Ciascun client è in grado di interagire con il server indipendentemente dagli altri client (le letture possono avvenire in parallelo).

La comunicazione è basata unicamente su scambio di stringhe. Tutte le stringhe sono inviate da client a server e viceversa utilizzando il carattere di fine linea come separatore.

Durante l'esame il server sarà raggiungibile al seguente indirizzo:

- Indirizzo IP: 80.211.232.219
- Porta: 4400

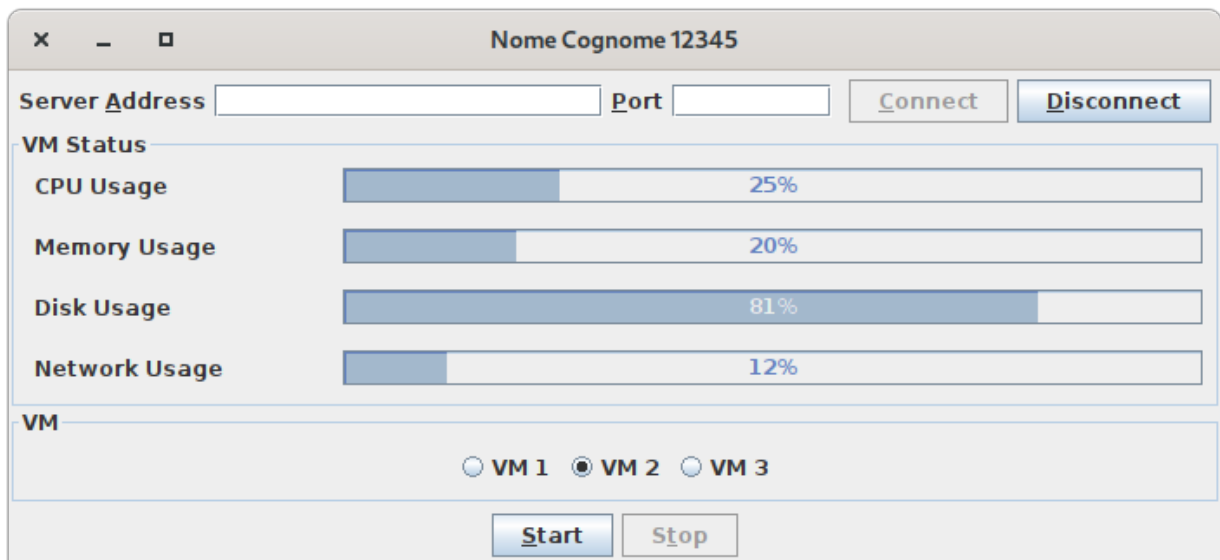


Figura 1: Interfaccia del client da realizzare

Si richiede la realizzazione del client, con interfaccia grafica e networking, in grado di comunicare con il server.

L'interfaccia grafica del client dovrà essere composta da una finestra che abbia come titolo *nome cognome matricola* dello studente, da due campi testuali editabili per l'indirizzo e la porta del server, da quattro barre di progresso, ciascuna con la rispettiva etichetta ad indicare il valore misurato, e da sette pulsanti che realizzino le seguenti funzioni: 1) *Connect*: permette di connettersi al server utilizzando indirizzo e porta specificati nei campi testuali; 2) *Disconnect*: permette di chiudere la connessione al server; 3) *Start*: permette di inviare la richiesta di download delle misurazioni al server; 4) *Stop*: permette di interrompere la ricezione dal server durante il download; 5) *VM1-3*: 3 istanze di `JRadioButton` per la selezione della VM di interesse. I 3 pulsanti devono fare parte di un unico gruppo in modo tale che uno solo di essi alla volta possa essere selezionato (si consultino le documentazioni di `JRadioButton` e `ButtonGroup`).

Le barre di progresso sono realizzate da 4 istanze della classe `JProgressBar`, con valori minimo e massimo rispettivamente 0 e 100, in quanto le misurazioni ricevute dal server indicano la percentuale di saturazione delle risorse allocate alla VM per ciascuna misurazione.

Suggerimento: Per far mostrare il valore attuale all'interno di ciascuna istanza di `JProgressBar` come da Figura 1 è necessario invocare su di essa il metodo `setStringPainted(true)`.

Si implementi il seguente comportamento dell'interfaccia:

- All'avvio solamente il pulsante *Connect* deve essere abilitato.
- Alla pressione del pulsante *Connect*, il client invia una richiesta di connessione al server utilizzando *indirizzo e porta indicati negli appositi campi*.
- Una volta stabilita correttamente la connessione, il client deve disabilitare il pulsante *Connect*, ed abilitare i pulsanti *Start*, *VM1-3* e *Disconnect*. Alla pressione del pulsante *Start*, il client deve inviare al server una stringa nella forma:

`start:vm<X>`

rimpiazzando ad `<X>` il numero della VM corrispondente al radio button selezionato. Ad esempio, considerando lo stato dell'interfaccia in Figura 1, alla pressione del pulsante *Start* il client invierà la seguente stringa al server:

`start:vm2`

Il client dovrà inoltre abilitare il pulsante *Stop*, e disabilitare i pulsanti *Start*, *VM1-3* e *Disconnect* in quanto non deve essere possibile inviare una nuova richiesta o disconnettersi dal server mentre è in corso il download. Alla pressione del pulsante *Stop*, durante il download, il client dovrà inviare la stringa "stop" al server, il quale provvederà ad interrompere l'invio delle misurazioni.

- Alla ricezione della stringa "start:vmX", il server inizierà ad inviare, ad intervalli regolari, le misurazioni relative alla VM richiesta, nel seguente formato:

`cpu:mem:disk:net`

dove *cpu*, *mem*, *disk*, *net* sono valori interi nell'intervallo [0, 100] rappresentanti l'utilizzo in percentuale rispettivamente delle risorse CPU, RAM, throughput del disco e dell'interfaccia di rete. Ad esempio, la stringa

`50:20:10:5`

indica che la VM in questione sta attualmente utilizzando il 50% della CPU, il 20% della RAM, e rispettivamente il 10% e 5% del throughput del disco rigido e della rete allocati.

Il server continuerà ad inviare periodicamente le informazioni per un tempo massimo di 30 secondi, o fintanto che il client non richieda l'interruzione del download tramite la stringa "stop". Al termine dell'invio, sia esso per raggiungimento del tempo massimo o per richiesta

del client, il server invierà la stringa “*” ad indicare l’avvenuta interruzione, e rimarrà in attesa di nuovi comandi.

Qualora la VM richiesta dal client non sia tra quelle predefinite, il server invierà la stringa “Not Found”, terminerà l’invio con “*” e rimarrà in attesa di nuove richieste. Per qualsiasi altra stringa ricevuta al di fuori di quelle indicate, il server risponderà con la stringa “error”, ed interromperà immediatamente la connessione con il client.

Alla ricezione di ciascuna stringa, il client deve effettuare le seguenti operazioni:

- Alla ricezione di una stringa `cpu:mem:disk:net`, il client deve tradurre tali valori in interi, e visualizzarli tramite le barre di progresso nell’interfaccia grafica (passandoli al metodo `setValue(int)` di ciascuna di esse).
- Alla ricezione della stringa “*”, il client deve disattivare il pulsante *Stop*, e riattivare i pulsanti *Start*, *VM1-3* e *Disconnect* in quanto deve essere possibile selezionare una nuova VM ed avviare un nuovo download senza dover disconnettere o riavviare il client.

Suggerimento: Al fine di semplificare la fase di debug si suggerisce di far stampare a terminale tutte le stringhe ricevute dal server.

- Alla pressione del pulsante *Disconnect*, il client deve inviare la stringa “disconnect”, chiudere tutti i canali di comunicazione generati in fase di connessione, e deve inoltre disabilitare i pulsanti *Start*, *Stop*, *VM1-3* ed abilitare nuovamente il pulsante *Connect* in quanto deve essere possibile instaurare una nuova connessione senza che sia necessario il riavvio del client.