



PROGETTO DI LABORATORIO DI SISTEMI
OPERATIVI 2024/2025
APPLICAZIONE CLIENT-SERVER CHE GESTISCE
AREE DI GIOCO DEL TRIS

NOME GRUPPO: TICtAcTEAM

MURANO	MONTE	LIBERTI
STEFANO	CRISTIAN	SIMONE
N86004383	N86004326	N86004252

APRILE 2025

Indice

1	Documentazione	3
1.1	Scelte di Implementazione	3
1.2	Requisiti e Relative Risoluzioni	4

Capitolo 1

Documentazione

1.1 Scelte di Implementazione

La scelta implementativa dell'applicazione Client-Server, verte sull'utilizzo di un Server in linguaggio C, rispettando il vincolo imposto dal cliente che ha commissionato tale applicazione, la scelta per il client, verte sul linguaggio Python, che fa parte dei linguaggi di scripting, tale scelta è stata ponderata basandoci su quelle che sono le peculiarità di tale linguaggio, sotto diversi punti come:

1. **Semplicità e leggibilità:** Python è noto per la sua sintassi chiara e leggibile, il che rende più facile scrivere e mantenere il codice. Questo è particolarmente utile quando si lavora in team o si devono apportare modifiche nel tempo.
2. **Ampia libreria di moduli:** Python offre una vasta gamma di librerie e framework, come Flask e Django per il server, e Requests per il client. Questi strumenti possono semplificare notevolmente lo sviluppo e ridurre il tempo necessario per implementare funzionalità comuni.
3. **Supporto per la programmazione asincrona:** Con librerie come `asyncio`, Python consente di gestire operazioni di rete in modo efficiente, migliorando le prestazioni delle applicazioni Client-Server, specialmente quando si tratta di gestire molte connessioni simultanee.
4. **Portabilità:** Python è un linguaggio multiplatforma, il che significa che le applicazioni possono essere eseguite su diversi sistemi operativi senza modifiche significative al codice.
5. **Comunità attiva:** La comunità di Python è molto attiva e offre un ampio supporto. Ci sono molte risorse, tutorial e forum dove puoi trovare aiuto e condividere esperienze.
6. **Integrazione con altre tecnologie:** Python si integra bene con altre tecnologie e linguaggi, il che facilita l'interazione con database, API e servizi esterni.

7. **Rapidità di sviluppo:** Grazie alla sua sintassi semplice e alle librerie disponibili, Python consente uno sviluppo rapido, il che è ideale per prototipazione e iterazione veloce.

Per la gestione della comunicazione tra Server e Client, dato che vengono creati in due linguaggi di programmazione differenti, soprattutto per il loro paradigma computazione, tale risoluzione avviene mediante l'utilizzo nel Server della libreria in linguaggio C chiamata:

CJSON.c: che ha lo scopo di prelevare la stringa del buffer popolata nel server, convertirla in stringa JSON in modo tale da comunicare con il Client in Python che attraverso una sua libreria converte e spacchetta il JSON per ottenere il i dati richiesti.

1.2 Requisiti e Relative Risoluzioni

Tale applicazione Client-Server, prevede delle direttive specifiche inerenti al funzionamento, ovvero che ogni giocatore che per scelte di struttura viene autenticato attraverso un suo username, una volta entrato nell'area a sua disposizione, può creare una o più partite, può partecipare ad un'unica partita per volta. L'utente che crea l'ambiente partita può scegliere di accettare o rifiutare le richieste che vengono effettuate da altri client per partecipare alla partita, e tale problematica è stata ovviata attraverso l'utilizzo di Thread che ne gestiscono le richieste, questo per ogni Client, anche perché tale applicazione presenta uno stile Multi-Client dove i Client non solo comunicano con il Server ma soprattutto tra Client. Una volta che il Client crea una partita dove ogni partita ha un suo identificativo, l'entità partita viene gestita attraverso degli attributi di stato, ovvero:

- **Terminata:** Stato che rappresenta una partita conclusa ed entrambi i client non intendono proseguire.
- **In Attesa:** Stato che rappresenta una partita creata che attende altre comunicazione, ovvero quando una partita viene creata, il suo stato viene messo automaticamente in Attesa, perché in tal caso bisogna far sì che altri Client che vogliono partecipare alla partita possano inviare la richiesta di partecipazione.
- **In Corso:** Stato che rappresenta una partita avviata, anzi dal momento in cui il Client Owner della partita accetta una richiesta da parte un Client Guest, allora lo stato della partita muta in In Corso. E non viene permesso ad altri Client di inviare la richiesta per quella partita.
- **Creazione:** "ANCORA DA DEFINIRE"

Per quanto riguarda la gestione su gli esiti della partita, viene definita in modo che ad ogni nuova mossa sia del Client Owner sia del Client Guest si controlla in base alla sua mossa, la verticale, l'orizzontale e diagonale primaria e secondaria.

Nel caso in cui la partita termina, i giocatori possono scegliere rigiocare oppure no. Tra i requisiti Opzionali che il nostro gruppo ha deciso di eseguire, vi è, che al termine di una partita, il perdente dovrà uscire dalla partita, e se a perdere la partita è il Client Guest della partita allora esso viene espulso dalla partita, e poi il Client Owner della partita decide se proseguire con una nuova partita e quindi accettare una richiesta, oppure uscire anche lui. Nel caso in cui a Vincere sia il Client Guest della partita, il Client Owner viene espulso in quanto perdente, e il Client Guest diventerà il nuovo Owner per quella partita ed ha sua volta potrà scegliere se fare un'altra partita e quindi accettare una delle richieste da altri Client, oppure uscire anche esso, NOTA:(Quando sia Owner sia Guest decidono di non giocare più, escono dalla partita e la partita viene eliminata). Nel caso di pareggio della partita, sia Client Owner sia Client Guest della partita possono scegliere se rigiocare, ovviamente in caso uno dei due o entrambi non vogliano, si applicano le regole definite in precedenza.