# Server

Abbiamo utilizzato la libreria socket di Python per gestire connessioni multiple dai client. Ogni client viene inserito in una lista, che viene letta durante l’invio dei dati e durante la disconnessione per sapere quali sono i client connessi. I pacchetti sono in formato JSON, organizzati in chiavi e valori. Ogni pacchetto in ingresso viene letto e vengono eseguiti i comandi richiesti al suo interno (ad es. muovendo i motori o cambiando la velocità). I client, invece, vengono aggiornati periodicamente sullo stato del rover e dei sensori con altri pacchetti JSON.

# Client

Il client ha il compito di leggere i valori dal server ed aggiornare l’interfaccia utente di conseguenza. Ha inoltre dei metodi specifici per l’invio dei pacchetti di movimento. È stata anche disposta una utility parallela per la ricerca in LAN dei server, che scansiona la rete e connette il client al server in caso di disconnessione.

# GUI

L’interfaccia utente utilizza il client per aggiornare i valori dei sensori e inviare comandi alla pressione dei pulsanti. Permette di inserire il nome host e connettersi, ha un joystick virtuale per il movimento e degli slider per le velocità. Infine, può inviare il comando che attiva o disattiva la modalità Machine Learning. L’interfaccia è scritta con le librerie PyQt5.