

Progetto

Nella seconda prova il Nao collabora con noi Ciceroni nel fare delle domande a potenziali visitatori del Mediamuseum, in questa circostanza rappresentati anch'essi da noi dal momento che non potevamo ricorrere a veri visitatori, per rendere la visita più coinvolgente. Per raggiungere questo scopo abbiamo utilizzato anche l'Arduino, che si illumina di verde nel momento in cui il visitatore fornisce la risposta esatta. Le domande che il Nao rivolge ai visitatori riguardano la vita di Ennio Flaiano, le sue opere e i suoi aforismi e lo stesso Nao è il personaggio di un suo film, ossia Kunt di "Un Marziano a Roma".

Programmazione Nao

Per far sì che il Nao interagisse con i Ciceroni e i visitatori, lo abbiamo programmato attraverso il software Coreographe. Abbiamo dato come primo input per il Nao una casella "Animated say" in cui saluta. Abbiamo deciso di iniziare con questo tipo di casella per il dialogo e non con la casella "Dialog", che abbiamo usato per tutti gli altri dialoghi, in modo che il Nao potesse parlare senza aver ricevuto prima un input vocale da parte nostra, ma autonomamente all'avvio del programma. Alla casella "Animated say" abbiamo inserito come output la "Set language", necessaria per impostare la lingua quando si va ad utilizzare una casella "Dialog", che è infatti il suo output. È proprio nella casella "Dialog" che sono presenti tutti i dialoghi della prova, ad eccezione del saluto iniziale (come detto in precedenza). Per poter pronunciare le frasi inserite all'interno della casella "Dialog", il robot ha bisogno che qualcuno pronunci una o più parole che gli vengono date come input, in modo da poter rispondere. Abbiamo scelto le parole da usare in base alla facilità con cui il robot le comprendeva; infatti, il robot reagisce solo se la sua soglia di comprensione supera il 50%. A questa casella di dialogo abbiamo collegato come output altre caselle che permettessero al Nao di muoversi oltre a parlare. Per i movimenti abbiamo utilizzato i boxes: "Wipe", che permette al robot di realizzare un gesto con il braccio che faccia sembrare si stia asciugando la fronte; "Hands", che gli permette di aprire e chiudere le mani; e "Breath", che attiva l'animazione della respirazione.

Finalità del progetto

La finalità del progetto è la promozione del patrimonio culturale del Mediamuseum. Inoltre, per far conoscere questo museo a persone che ne sentono parlare per la prima volta, abbiamo pensato alla creazione di un QR code che contenga un link ad un video dove il Nao pone delle domande sulle sale e sui contenuti, a cui i ragazzi dovrebbero rispondere usando i propri telefoni, il che rende la visita ancora più coinvolgente.

Comunicazione NAO e Arduino

Programmazione NAO

Per far comunicare NAO e Arduino abbiamo bisogno di una serie di blocchi di programmazione; innanzitutto serve un "Set Reco. Lang." per determinare il linguaggio, nel nostro caso l'italiano. Sono poi inseriti i blocchetti "Speech Reco." (in cui scegliamo le parole che il NAO dovrà riconoscere) collegato con un output di tipo stringa al successivo, il "Socket". La programmazione di quest'ultimo è un po' più complessa, in quanto non appare predefinito nella libreria ma va creato manualmente. Si deve quindi prendere un blocco "python" e inserire all'interno di esso due parametri, Yun IP Address (192.168.240.1 nel nostro caso) e la Porta (7891). Una volta fatto ciò bisogna scrivere le linee di programmazione del "Socket", inserendo come segnali di accensione e spegnimento le parole specificate in "Speech Reco.", nel nostro caso "giusto" e "indizio". Queste due parole vengono associate a dei numeri, 1 (giusto) e 2 (indizio).

Programmazione Arduino.

A questo punto entra in gioco la programmazione di Arduino, codificata tramite l'omonimo software. In questo caso quindi, mettendo in comunicazione NAO e Arduino, non appena il robot sente la parola "giusto" manda il segnale numerico "1" a Arduino. Nella sua programmazione, "1" corrisponde al comando "digitalWrite(LED, HIGH);" e quindi "strcpy(echo_message, "Acceso da yun");", che dunque farà accendere il LED. Allo stesso modo, quando il NAO sente "indizio" tramite il Socket invierà il segnale "2" all'Arduino, che corrisponde a "digitalWrite(LED, LOW);" e quindi "strcpy(echo_message, "Spento da yun");" per spegnere il LED.