BOT TELEGRAM

INTRODUZIONE

Questo software è stato realizzato utilizzando Visual Studio Code e il linguaggio di programmazione Python. È parte integrante del progetto “Neighborhood Delivery” creato per l’esame di IoT & 3D Intelligent Systems dal gruppo Kaido, Merolla e Calvano.

Il software viene eseguito su EC2, gestita dal sistema di Amazon AWS ed è uno degli script che compongono la parte Cloud del progetto. È collegato al database MySQL presente sulla stessa macchina e riceve ed invia messaggi tramite Telegram a tutti gli utenti del sistema, in base alle loro richieste o ad avvenimenti precisi. È in ascolto sulla porta 5001.

FUNZIONAMENTO

Lo script Python è realizzato utilizzando Flask e rimane in ascolto di ricevere ordini da utenti tramite la chat di Telegram connessa al Bot oppure dallo script “Find Neighbour” (si veda documentazione appropriata).

Quando il software riceve un comando tramite chat, risponde ricavando informazioni dal Database oppure inserendo informazioni su di esso (in base al contenuto del messaggio dell’utente). È inoltre in grado di inviare messaggi agli utenti interessati della consegna di un pacco utilizzando il Bot di Telegram.

CODICE E COMANDI PRINCIPALI

Il codice, scritto in Python, è scomposto in varie funzioni che corrispondono ai vari step necessari per concludere la richiesta fatta lato utente finale.

Qui di seguito sono elencati i principali comandi utilizzabili tramite chat Telegram con il Bot.

* ***/config***

Comando che consente di configurare un nuovo dispostivo/casa da parte dell’utente. Grazie allo scambio di informazioni tra bot e utente, all’interno del database vengono inseriti nome, cognome, posizione, chatID Telegram e dispositivo acquistato.

* ***/reset***

Comando che consente di eliminare un dispositivo/casa dal database, qualora non fosse più richiesto il servizio offerto. La procedura è guidata mediante il dialogo con il Bot.

* ***/cancel***

In qualunque momento della conversazione, utilizzando questo comando, è possibile interrompere l’operazione svolta, annullandola.

Infine, come detto precedentemente, il codice riceve una richiesta http di tipo POST da parte dello script “Find Neighbour” quando diventa necessario inviare messaggi agli utenti interessati di una consegna.

Il primo messaggio è inviato all’effettivo proprietario del pacco, e gli segnala che esso verrà recapitato ad un suo vicino in un determinato indirizzo. Il secondo messaggio è inviato al nuovo “destinatario” del pacco, che viene avvisato dell’imminente arrivo del corriere per effettuare la consegna “spostata”.

DIPENDENZE

Il codice Python utilizza alcune librerie esterne per svolgere i suoi compiti.

1. **pymysql**, per comunicare con il database, reperire le informazioni ed aggiornarlo.
2. **requests**, per ricevere la richiesta POST dello script “Find Neighbour”.
3. **Telegram**, per gestire la comunicazione Bot-utente.
4. **Flask**, per consentire le connessioni allo script stesso dall’esterno.
5. **Geopy**, per manipolare i dati relativi alla posizione dell’utente.