

Documentazione

ArpavldroBot

(Bot per Telegram)

Data di ultima modifica: 14/11/2016

Indice

1 Introduzione.....	3
2 I file e le cartelle che compongono il bot.....	4
3 Installazione.....	5
3.1 Requisiti.....	5
3.2 Come installare le componenti richieste.....	5
3.2.1 Curl.....	5
3.2.2 php-xml.....	5
3.3 Come creare il database.....	5
3.3.1 Creare il database.....	5
3.3.2 Creare le tabelle e le procedure.....	6
3.3.3 Creare l'utente del database per il bot.....	6
4 Amministrare il bot.....	7
4.1 Avviare il bot.....	7
4.2 Spegner il bot.....	7
4.2 Resettare il database.....	7
5 Struttura del codice.....	8
6 Manutenzione adattiva.....	9
6.1 Come aggiungere dei nuovi livelli di criticità.....	9
6.2 Come cambiare il testo di un messaggio.....	9
6.3 Come cambiare i percorsi ai file.....	9
6.4 Come cambiare il token del bot.....	9
6.5 Cambiare la descrizione o la foto del bot.....	9
7 Manutenzione perfetta.....	11
7.1 Come aggiungere un comando.....	11
7.2 Come rimuovere un comando.....	11

1 Introduzione

Questo documento descrive alcuni aspetti del bot per Telegram denominato ArpavIdroBot, di seguito “il bot”.

Di seguito nel documento sono descritti i seguenti aspetti del bot:

- I file e le cartelle che compongono il bot.
- Come installare il bot.
- Come avviare e spegnere il bot.
- Struttura del codice.
- Come effettuare la manutenzione adattiva.
- Come effettuare la manutenzione perfetta.

2 I file e le cartelle che compongono il bot

Tutti i file necessari per far funzionare il bot si trovano nella cartella denominata “ArpavIdroBot”.

Questa è la struttura dei file:

ArpavIdroBot (cartella radice)

ArpavIdroBot.log – File contenente i log generati dal bot.

ArpavIdroBot.php – Contiene il codice del bot.

ArpavIdroBotLauncher.php – Parte del codice che eseguire il bot.

connect_DB.php – Parte del codice necessaria per connettersi al database

descrizione_comandi.txt - Descrizioni dei comandi per il bot @BotFather di Telegram.

livelli.xml – Livelli di criticità per le stazioni.

start.sh – Script da invocare per avviare il bot in background.

stop.sh – Script da invocare per fermare l'esecuzione del bot.

mysql_database (cartella)

createDB.sql – Contiene istruzioni sql per creare le tabelle e le procedure.

createProcedures.sql

createTables.sql

dropDB.sql - Contiene istruzioni sql per eliminare le tabelle e le procedure.

dropProcedures.sql

dropTables.sql

3 Installazione

Questo capitolo descrive la procedura da seguire per installare il bot.

3.1 Requisiti

Il bot funziona con la seguente configurazione del sistema.

Una configurazione differente non necessariamente preclude il corretto funzionamento del bot.

- **Sistema operativo:** Ubuntu 16.04
- **Interprete PHP:** PHP 7.0
- **Database:** Mysql 5.7.13

3.2 Come installare le componenti richieste

E' necessario che le seguenti componenti siano installate nel sistema.

3.2.1 Curl

Per Ubuntu 16.04 basta digitare il seguente comando per installarlo:

`sudo apt-get install php-curl`

3.2.2 php-xml

Per Ubuntu 16.04 basta digitare il seguente comando per installarlo:

`sudo apt-get install php7.0-xml`

3.3 Come creare il database

3.3.1 Creare il database

Nella shell del sistema posizionarsi nella cartella mysql_database del bot.

Avviare mysql con un utente avente un alto grado di privilegi.

Esempio di comando per l'utente "root":

`mysql -u root -p`

Creare il database denominato "BotIdro" con il seguente comando:

CREATE DATABASE BotIdro;

Posizionarsi nel database appena creato:

USE BotIdro;

3.3.2 Creare le tabelle e le procedure

Digitare il seguente comando per creare le tabelle e le procedure necessarie:

SOURCE createDB.sql

3.3.3 Creare l'utente del database per il bot

Per creare un utente per il bot digitare:

CREATE USER `BotIdro`@`localhost` IDENTIFIED BY 'password';

Dove “BotIdro” è il nome dell'utente e “password” è la parola chiave.

Per dare all'utente appena creato i privilegi necessari digita il comando:

GRANT SELECT,INSERT,UPDATE,DELETE,EXECUTE ON BotIdro.* TO 'BotIdro'@'localhost';

Infine digita il seguente comando per rendere effettive le modifiche ai privilegi:

FLUSH PRIVILEGES;

Infine inserisci le credenziali per l'accesso al database nel file “connect_DB.php”.

4 Amministrare il bot

4.1 Avviare il bot

Per avviare il bot posizionarsi nella cartella principale del bot (ArpavIdroBot) e digitare il seguente comando:

```
bash ./start.sh
```

Dopo l'esecuzione di questo comando verrà abilitato l'avvio automatico del bot al riavvio del computer.

Questo comando imposta anche la gestione del file di log “ArpavIdroBot.log”, in modo che la sua grandezza non cresca eccessivamente.

4.2 Spegnare il bot

Per terminare l'esecuzione del bot posizionarsi nella cartella principale del bot (ArpavIdroBot) e digitare il comando:

```
bash ./stop.sh
```

Dopo l'esecuzione di questo comando verrà disabilitato l'avvio automatico del bot al riavvio del computer.

Inoltre la grandezza del file di log “ArpavIdroBot.log” non sarà più gestita.

4.2 Resettare il database

Se c'è il bisogno di ripulire il database si possono usare, dal terminale di mysql dopo essersi posizionati nella cartella “mysql_database”, i seguenti comandi:

```
SOURCE dropDB.sql
```

```
SOURCE createDB.sql
```

5 Struttura del codice

La maggior parte del codice si trova nel file ArpavIdroBot.php

Le funzioni sono suddivise in categorie.

L'inizio di una categoria è segnato dalla stringa:

```
//[nome categoria] – BEGIN//
```

La fine di una categoria è segnata dalla stringa:

```
//[nome categoria] – END//
```

Le categorie sono seguenti:

CORE – Le funzioni che interagiscono con Telegram.

Queste funzioni si occupano di:

Riconoscere i comandi digitati dall'utente e di invocare le funzioni associate ai comandi riconosciuti.

COMMANDS – Funzioni associate ai comandi del bot.

Queste funzioni sono invocate automaticamente dal CORE.

STRINGS TO USER – Funzioni che stampano i testi da mostrare all'utente.

SUPPORT FUNCTIONS – Funzioni di supporto di carattere generale.

ARRAYS FUNCTIONS – Funzioni che restituiscono array.

NOTIFICATION FUNCTIONS – Funzioni relative alla funzionalità di notifica del bot.

La funzionalità di notifica ha le seguenti caratteristiche:

Manda le notifiche agli utenti iscritti alle stazioni.

Le notifiche sono relative al cambio del livello dell'acqua.

Per alcune stazioni sono definiti dei livelli di criticità dell'acqua.

Le stazioni generano le notifiche al passaggio del livello dell'acqua da un livello di criticità ad un altro.

6 Manutenzione adattiva

Di seguito è spiegato come adattare il bot ai cambiamenti dell'ambiente operativo ed ad alcune altre modifiche richieste.

6.1 Come aggiungere dei nuovi livelli di criticità

E' sufficiente aggiungere i dati relativi ai nuovi livelli nel file "livelli.xml".

6.2 Come cambiare il testo di un messaggio

Le funzioni che generano i messaggi visualizzati dagli utenti si trovano nella categoria STRINGS TO USER nel file "ArpavIdroBot".

Usa la funzionalità di ricerca del tuo editor di testo per ricercare la frase che vuoi modificare.

6.3 Come cambiare i percorsi ai file

Le seguenti funzioni generano gli URL per l'accesso ai file di dati, e possono essere modificate.

getWaterLevelsURL – Genera l'indirizzo URL del file XML di una singola stazione.

getCriticalityLevelsURL – Genera il percorso al file XML contenente i livelli di criticità.

getHydroStationsURL – Genera l'URL al file XML contenente l'elenco delle stazioni idrometriche.

GetHydroStationGraphicURL – Genera l'URL al grafico degli andamenti di una stazione.

6.4 Come cambiare il token del bot

Il token è una stringa segreta che permette al bot di comunicare con Telegram.

Per generare un nuovo token usa il bot @BotFather messo a disposizione da Telegram.

Scrivi il comando /token e segui le istruzioni. Alla fine ti verrà dato il nuovo token associato al bot.

Sostituisci il token precedente, che si trova nel file "ArpavIdroBot.php" nella variabile \$botToken.

6.5 Cambiare la descrizione o la foto del bot

Per cambiare la descrizione o la foto del bot usa il bot @BotFather messo a disposizione da Telegram.

I comandi sono:

/setuserpic – Per cambiare la foto del bot.

/setdescription – Per cambiare la descrizione del bot.

7 Manutenzione perfetta

Di seguito è spiegato come modificare alcune funzionalità del bot.

7.1 Come aggiungere un comando

Per aggiungere un nuovo comando al bot bisogna:

1) Aggiungere la funzione denominata **command_[nome del comando]** nella categoria COMMANDS. Inserisci in questa funzione anche la logica del comando.

2) Aggiungere nella categoria STRINGS TO USER le funzioni:

descrizione_[nome del comando] – Restituisce una stringa di descrizione dettagliata del comando.

syntax_[nome del comando] – Restituisce un esempio di sintassi corretta del comando.

3) Aggiungere il comando e la sua descrizione nel file “descrizione_comandi.txt”.

4) Usare il comando **/setcommands** nel bot @BotFather di Telegram per aggiornare le descrizioni dei comandi.

7.2 Come rimuovere un comando

Per rimuovere un comando seguire i passi del paragrafo precedente rimuovendo il comando desiderato invece di aggiungerlo.