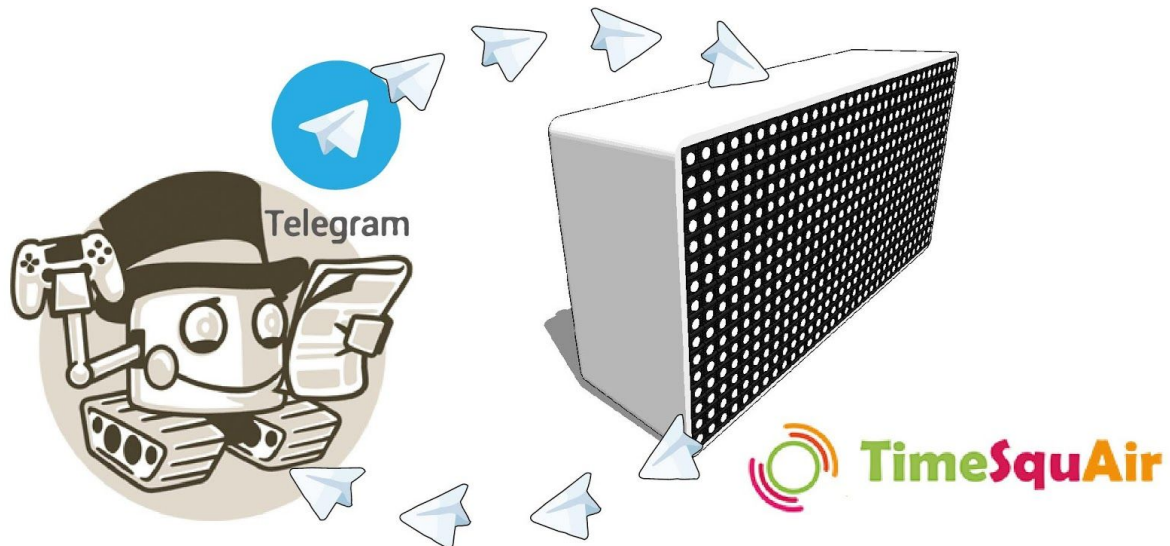


Telegram Bot with TimeSquAir : the bedrock



Willing to play with Bots? To play with IoT? Curious to combine IoT and Bots for magical connected use cases?

Here we propose you a terrific bedrock to build your own Bot, thanks to the blending of two powerful ingredients:

- Telegram Bots (link): Telegram is a messaging system, with a great Bot API. May be the greatest.
- TimeSquAir (link): TimeSquAir is a device enabling the creation of connected use cases without needing specific software development skills: creating IoT scenarios is a piece of cake with TimeSquAir!

Our promises:

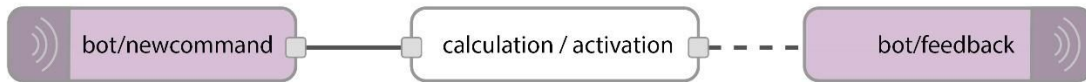
- You will build your own Bot usages in record time!
- You will link your Bot to IoT, hardware, sensors, IA, connectivity... ..naturally, without any extra effort..

The only 2 first steps require your attention and 4 minutes of your time. The other ones are just here to get a better understanding of what's actually going on, if required.

- Build your Bot use case in less than 1 minute!
- Install the TimeSquAir Telegram Bot bedrock in less than 3 minutes
- Get some explanation or details, only if you want!

Build your Bot use case in less than 1 minute!

Video



Adding a new Bot use case to the TimeSquAir Telegram Bot Bedrock is a matter of seconds! You won't need to subscribe to any extra system.

Don't worry, if you don't have the TimeSquAir Telegram Bot bedrock yet, getting it up and running is explained in the coming chapter.

Here's the structure of your own Telegram Bot handled by your own TimeSquAir:

- a MQTT input node, which topic is bot/newcommand (where newcommand is the new command!)
- the flow with your desired calculation/activation
- potentially a MQTT output node if you want a feedback on Telegram

The video shows how to add a configurable stopwatch with the command /count and the number of seconds as an argument.

Install the TimeSquAir Telegram Bot bedrock in less than 3 minutes!

Video

Follow the following steps, as shown in the video. And if you don't have a Telegram Bot yet, don't worry, it is described in the following chapter.

- Copy the flow given in appendix (TimeSquAir Telegram Bot bedrock flow) into your desktop notepad.
- Import it into a TimeSquAir new tab.
- Change the tokens in the "Set Globals" function on top of the flow (the "Bot token" is given by Telegram when you create your Bot, and the "Jedi token" is what you want, but we suggest to take a part of the "Bot token")
- After activation, launch the flow "Click "Go" to get the MQTT router topic in the Debug panel" to get in the debug panel the specific MQTT
- Change the topic of the MQTT input node of the "Bot's commands router" with the path copied on the previous point.
- Launch the flow "Launch this flow to link automatically the Telegram commands from your BOT to a MQTT service".

You're done! Just deploy this new flow, and your TimeSquAir Telegram Bot bedrock is up and running. The two examples make the following:

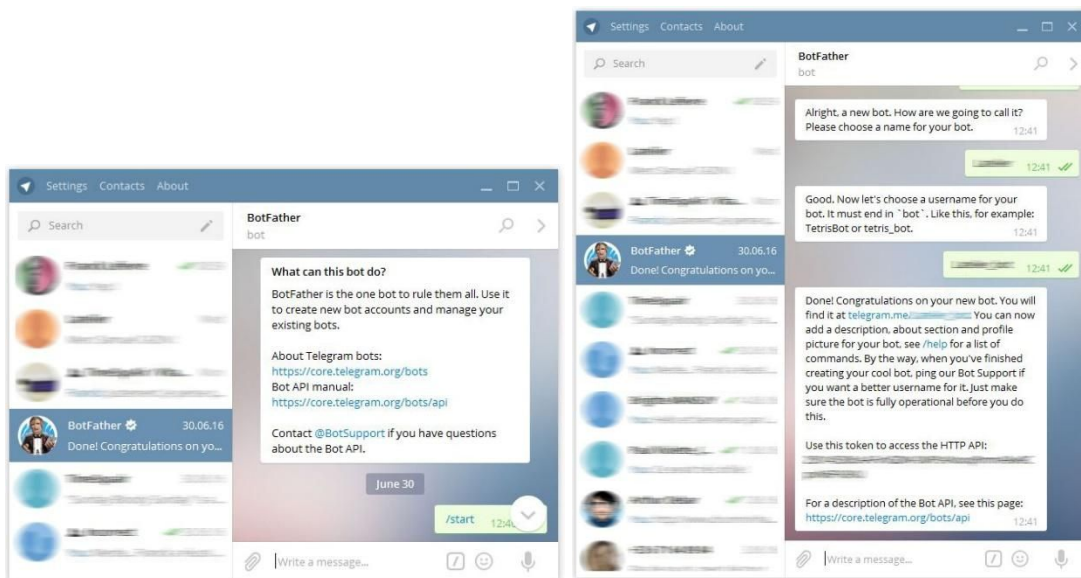
- If you type /radio_on it will switch on the Radio Phenix stream on your TimeSquAir, and you can switch it off by typing /radio_off

- If you type /display followed by a message, this message is displayed to your TimeSquAir's LED panel, and you receive a feedback on Telegram.

Some more details, just if you need or want

Starting from now, we only give additional information! We can here go through the flow, in order to explain and detail the structure. But first, let's explain how to create a new Telegram Bot!

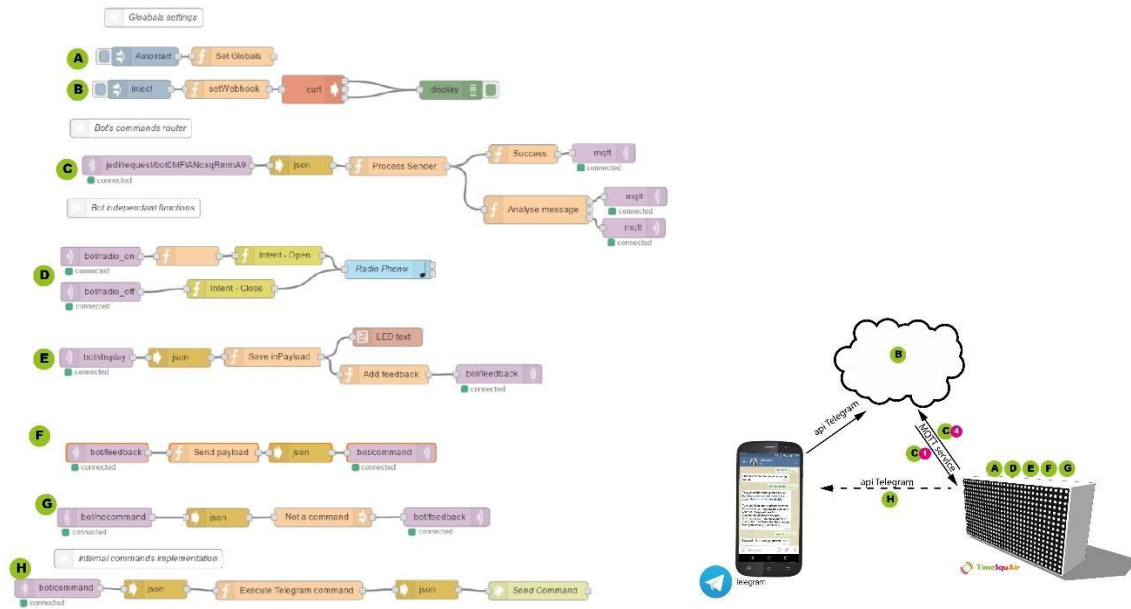
Create a new BOT on Telegram



To create a new Bot on Telegram is "simple as hello"!

- First of all, on your Telegram application, connect to « BotFather », the BOT that offers the interface to create a new BOT (find more about the creation of new bots through BotFather here: <https://core.telegram.org/bots>)
- Type "/newbot"
- Type the Name of the BOT you are being creating
- Type the Username of the BOT, finishing by "bot"
- Then, get the full token the BotFather is associating to your newborn BOT. For our example, the given token is: 250145306:AAFvhGDW-0MftANoxqRmmA9w9CpxN6XXXX

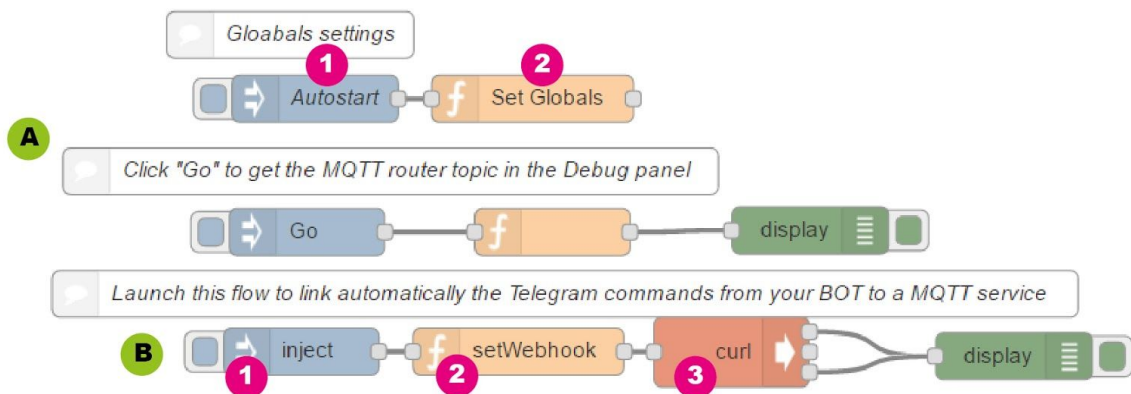
The bedrock flow



The bedrock flow is composed of 4 main parts:

- Settings (A & B)
- The router (C)
- The functions (D, E, F & G)
- The feedback implementation (H)

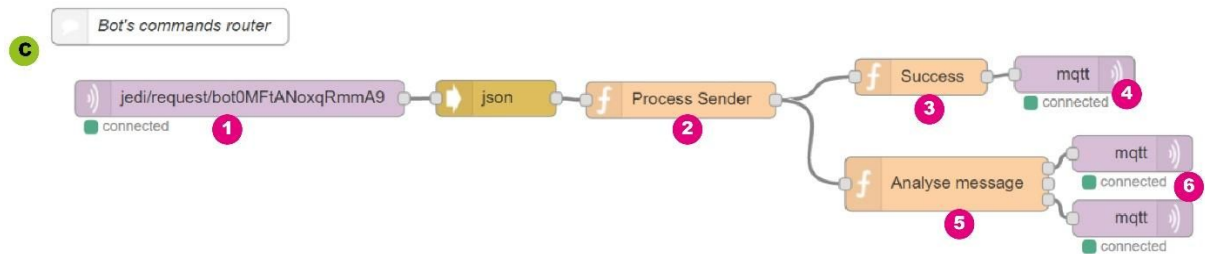
The settings (A)



- A- Let's configure the settings that are specific to you Bot (mainly the tokens). The purpose here is to do it through global parameters, which will be setup at each start of your TimeSquair.
The second flow automatically calculate the MQTT topic to configure the MQTT input node of flow C.
- B- Let's also build the flow that will systematically transform every Telegram command of your Bot into a MQTT service. The Curl function is used to build this link. This flow actually

needs to be run once, but running it each time TimeSquAir starts guaranty that the webhook has not been forgotten by the Telegram’s servers ;-). The Debug node at the end of the flow is of course not necessary, but it gives the status of the “webhook” of your Bot if required.

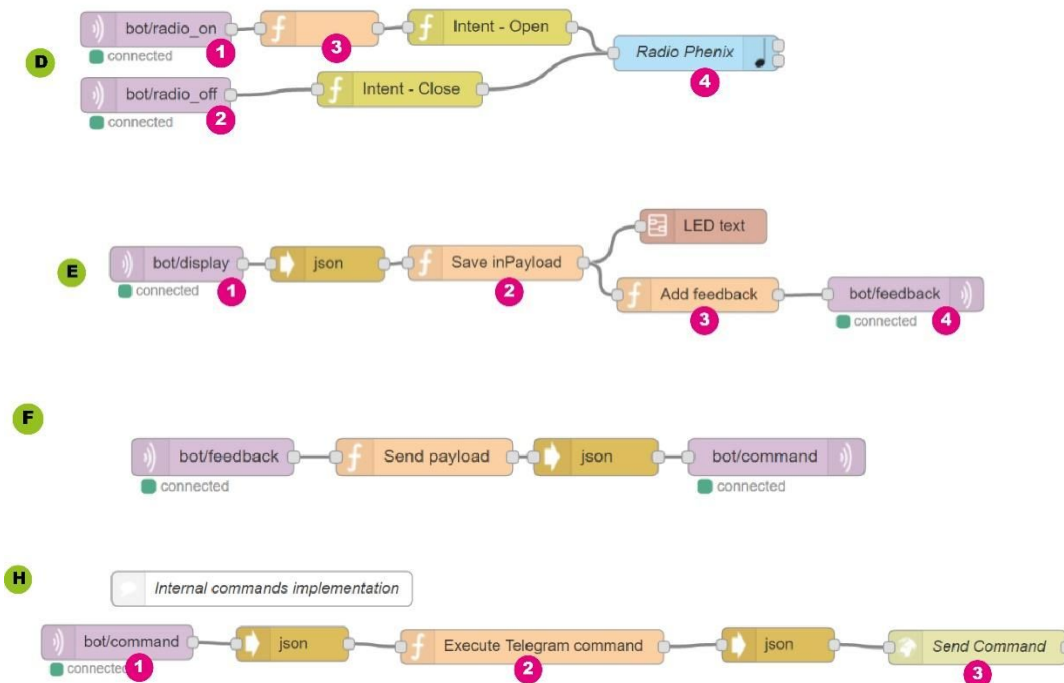
The router (C)



The Command’s router handles the incoming MQTT message (built from the telegram Bot request) and calculates the possible incoming commands and arguments for further processing.

- A- The MQTT output “4” is the answer sent in balance of the input “1”. The two other MQTT outputs are dedicated to the internal Bot functions (D, E and F), the first when there is one command (starting with the character /, with or without argument) and the second when the message received from the Bot is not a command.

The functions (D, E, F & G)



This part is where you simply add new functions of your Bot.

It always starts with the incoming MQTT input, which topic corresponds to the related command. The command might come with arguments.

- A- This is the example 1, proposing to launch and stop a web radio on TimeSquAir
- B- This is the example 2, that will display a message on TimeSquAir's LED panel.
- C- If a feedback to Telegram is needed, it is pre-calculated here
- D- For this basic Bot recipe, we just want to send back a "Not a command" message if the TimeSquAir receives a string that don't start with "/"

The feedback implementation (H)



- A- This flow sends information to your Telegram mobile application using the proper HTTP command from the Telegram's API..

Coming soon

Follow us and discover soon new cool stuff around Telegram bots and TimeSquAir like:

- Add Zwave to your Telegram Bot (switch a light and get temperature)
- Connect your Telegram Bot to Houndify services (get the answer to any topic)
- Handle subscription and user's rights.
- Send private specific contents to subscribers
- Take advantage of custom Telegram keyboards
- Receive automated pictures from a surveillance camera
- Create your own collaborative music jukebox

Appendix: The TimeSquAir Telegram Bot bedrock flow

```
[{"id":"Topic.LedMatrix","type":"queue-topic","origin":"/root/thethingbox/node_modules/node-red/flow/flows.json","topic":"LEDMatrix"},{"id":"MQTT.Localhost","type":"mqtt-broker","origin":"/root/thethingbox/node_modules/node-red/flow/flows.json","broker":"localhost","port":"1883","clientId":""},{"id":"99f30803.a91d48","type":"subflow","name":"LED text (5)","in":[{"x":54,"y":66,"wires":[{"id":"535ca4ae.1ef56c"}]}],"out":[],"origin":"/root/thethingbox/node_modules/node-red/flow/flows.json"},{"id":"535ca4ae.1ef56c","type":"function","origin":"/root/thethingbox/node_modules/node-red/flow/flows.json","name":"Format text if provide from calendar","func":"if (msg.hasOwnProperty(\"data\")){\n  if(msg.data.hasOwnProperty(\"kind\")){\n    if(~msg.data.kind.indexOf(\"calendar\")){\n      msg.payload = msg.payload.title + \" : \" + msg.payload.start;\n    }\n  }\n  return msg;\n}","outputs":1,"x":260,"y":65.58331298828125,"z":"99f30803.a91d48","wires":[["b221155a.b32388"]]},{"id":"b221155a.b32388","type":"function","origin":"/root/thethingbox/node_modules/node-red/flow/flows.json","name":"is time ?","func":"var re =\nmsg.payload.match(/((([01][0-9]|2[0-3]):[0-5][0-9])/);\\nif ( re !== null && re.length !==0 && re[0] ==\nmsg.payload ){\\n  /*\\n  var date = new Date();\\n  var h = Number(re[0].split(\":\")[0]);\\n  var m\n= Number(re[0].split(\":\")[1]);\\n  var h24 = h;\\n  if(~msg.payload.indexOf(\"pm\")){\\n\nh24+=12;\\n  }\\n  if(Number(date.getHours()) > h24 || Number(date.getMinutes()) > m ){\\n\nvar newTime = h+\":\"+m;\\n  msg.payload.replace(re[0],newTime);\\n  }*/\\n  \\n  return\n[msg,null];\\n}\\n}\\nreturn
```



```
} \n      } \n    }; \n      \nmsg.topic = topic ; \n \nreturn
msg; \n,"outputs":1,"noerr":0,"x":823.3808975219727,"y":347.7777738571167,"z":"293ee063.1298
4","wires":[["fd09e254.a78c5"]],{"id":"747e8e21.11976","type":"function","name":"Analyse
message","func":"var text = msg.botmessage.text ; \nvar command = \"\" ; \nvar args = \"\"
; \n \nif(text === undefined) \n { \n msg.topic = \"bot/filein\" ; \n msg.payload =
msg.botmessage ; \n return msg; \n } \n \nif(text.substring(0,1) == \"/\") \n { // The first word is
a command \n var firstSpace = text.indexOf(\" \") ; \n if(firstSpace == -1) \n { // There is a
command with no arg \n command = text.substring(1) ; \n } \n else \n { // There are
additional arg(s) \n command = text.substring(1, firstSpace) ; \n args =
text.substring(firstSpace+1) ; \n } \n msg.topic = \"bot/\" + command ; \n
msg.botmessage.text = args ; \n msg.payload = msg.botmessage ; \n return [msg, null, null]; \n
} \n \ncommand = \"nocommand\" ; \nargs = text ; \nmsg.topic = \"bot/\" + command
; \nmsg.botmessage.text = args ; \nmsg.payload = msg.botmessage ; \nreturn [null, null,
msg]; \n \n,\"outputs\":\"3\",\"noerr\":0,\"x\":844.8570709228516,\"y\":436.4285764694214,\"z\":\"293ee063.
12984\",\"wires":[["2a851773.081fd8"],[],["f42a08fa.6d50f8"]],{"id":"4da0dad5.436f14","type":"m
qtt
in","name":"","topic":"jedi/request/botAAFJmtCzHriipM9zE4od","broker":"530bf301.2c68dc","x":
255.71532440185547,"y":365.6998624801636,"z":"293ee063.12984","wires":[["dff1b01a.650ad"]
],{"id":"5d7e9cf7.a4f674","type":"http request","name":"Send
Command","method":"POST","ret":"txt","url":"","x":847.1948089599609,"y":1044.024676322937,
"z":"293ee063.12984","wires":[[]],{"id":"a13e3ee1.55b12","type":"mqtt
in","name":"","topic":"bot/command","broker":"MQTT.Localhost","x":81.0569839477539,"y":103
9.6307554244995,"z":"293ee063.12984","wires":[["2ee362bc.02c56e"]],{"id":"9355ac64.00191","
type":"function","name":"Execute Telegram command","func":"/*\nWe wait the following
payload's attributes:\n - recipient (mandatory) : Telegram recipient's ID \n - command (optional)
: Telegram's command (default = sendMessage) \n - text (optional) : Text to send (default =
none) \n - reply_markup (optional) : Cf. Telegram's doc (default = none) \n*/\n\nvar
BotSecretToken = context.global.telegram.token ; \n/\n\nvar inPayload = msg.payload ; \nvar
command = \"sendMessage\" ; \n \nif( inPayload.command !== undefined) \n { \n command =
inPayload.command ; \n } \n \nmsg.headers = { \"Content-Type\" : \"application/json\" }
; \nmsg.url = \"https://api.telegram.org/bot\" + BotSecretToken + \"/\n\" + command ; \nmsg.payload
= {} ; \nmsg.payload.chat_id = inPayload.recipient ; \n \nif(inPayload.text) \n msg.payload.text =
inPayload.text ; \nelse \n msg.payload.text = \"-\" ; \n \nif(inPayload.reply_markup) \n
msg.payload.reply_markup = inPayload.reply_markup ; \n \nreturn
msg; \n \n,\"outputs\":1,\"noerr\":0,\"x\":450.66766357421875,\"y\":1043.4319829940796,\"z\":\"293ee063.
12984\",\"wires":[["5e2c42d1.2ab5fc"]],{"id":"5e2c42d1.2ab5fc","type":"json","name":"","x":669.4
985542297363,"y":1041.705563545227,"z":"293ee063.12984","wires":[["5d7e9cf7.a4f674"]],{"id
":"2ee362bc.02c56e","type":"json","name":"","x":237.19243621826172,"y":1040.995873451233,"
z":"293ee063.12984","wires":[["9355ac64.00191"]],{"id":"f48987b4.744e28","type":"info","name
":"Internal commands
implementation","info":"","icon":"comment.png","langs":{"key":"en-US","title":"Internal
commands
implementation","body":"","isDefault":true}},{"currentLanguage":"fr-FR","x":210.0495376586914,"
y":992.8530216217041,"z":"293ee063.12984","wires":[],{"id":"1b33db31.072f05","type":"info",
"name":"Bot independant
functions","info":"","icon":"comment.png","langs":{"key":"en-US","title":"Bot independant
functions","body":"","isDefault":true}},{"currentLanguage":"fr-FR","x":202.54853057861328,"y":43
3.62622356414795,"z":"293ee063.12984","wires":[],{"id":"950eb51b.8b14b8","type":"mqtt
in","name":"","topic":"bot/nocommand","broker":"MQTT.Localhost","x":156.16500854492188,"y"
```



```
:926.7776880264282,"z":"293ee063.12984","wires":[[{"id":"d7e5dfb9.1955b"}]],{"id":"d7e5dfb9.1955b",
,"type":"json","name":"","x":336.3613586425781,"y":926.9498109817505,"z":"293ee063.12984",
"wires":[[{"id":"2ddad3d1.26944c"}]],{"id":"b981faec.fb4738","type":"function","name":"Set
Globals","func":"context.global.telegram = {} ;\ncontext.global.telegram.token = \"Enter your own
token\" ;\n\ncontext.global.jedi = {} ;\ncontext.global.jedi.token =
context.global.telegram.token.substr(10, 20) ;\n\n\ncontext.global.database = {}
;\ncontext.global.database.name = \"TSA_trial_bot\" ;\n\nreturn
msg;","outputs":1,"noerr":0,"x":360.38092041015625,"y":81.99999999999999,"z":"293ee063.129
84","wires":[[{"id":"682bb8cc.b677f8","type":"info","name":"Gloabals
settings","info":"","icon":"comment.png","langs":{"key":"en-US","title":"Gloabals
settings","body":"","isDefault":true},"currentLanguage":"fr-FR","x":234.38092041015625,"y":20,"
z":"293ee063.12984","wires":[]},{id":"1e768163.7b7b3f","type":"inject","name":"","topic":"","pa
yload":"","payloadType":"none","repeat":"","crontab":"","once":false,"x":211.38092041015625,"y
":244.00000286102295,"z":"293ee063.12984","wires":[[{"id":"52d5085a.490ec8"}]],{"id":"52d5085a.49
0ec8","type":"function","name":"setWebhook","func":"\n// Note: we consider here that the name
of the topic used by Jedi is this sub-part of the token:\nvar topic = context.global.jedi.token
;\n\nmsg.payload = \"-F \\\"url=https://api.timesquair.io/api/jedi/bot\" +\n  topic + \"\\\"\" +
\" \" +\n  \"\\\"https://api.telegram.org/bot\" + context.global.telegram.token +
\"/setWebhook\\\"\" ;\n\nreturn(msg)
;\n","outputs":1,"noerr":0,"x":351.3808898925781,"y":244.00000286102295,"z":"293ee063.1298
4","wires":[[{"id":"45e11bab.314564"}]],{"id":"45e11bab.314564","type":"exec","command":"curl","ad
dpay":true,"append":"","useSpawn":"","name":"","x":488.6031494140625,"y":244.222201347351
07,"z":"293ee063.12984","wires":[[{"id":"6afdf80f.cf6f68"},[{"id":"6afdf80f.cf6f68"}]],{"id":"6afdf80f.cf6f68
","type":"display","active":true,"x":670.3808441162109,"y":246.99997997283936,"z":"293ee063.1
2984","wires":[]},{id":"1f6fa5ec.6fce6a","type":"mqtt
in","name":"","topic":"bot/display","broker":"MQTT.Localhost","x":142.9150161743164,"y":671.9
898862838745,"z":"293ee063.12984","wires":[[{"id":"41f07ef6.83c57"}]],{"id":"41f07ef6.83c57","type":
"json","name":"","x":275.9149932861328,"y":671.9898767471313,"z":"293ee063.12984","wires":[
{"id":"315d05e1.8ca7da"}]],{"id":"315d05e1.8ca7da","type":"function","name":"Save
inPayload","func":"msg.inPayload = msg.payload ;\n\nmsg.payload = msg.payload.text
;\n\nreturn
msg;","outputs":1,"noerr":0,"x":423.9149932861328,"y":670.9898767471313,"z":"293ee063.1298
4","wires":[[{"id":"c3862bc6.231258","type":"subflow:9
9f30803.a91d48","name":"","x":606.9150238037109,"y":638.9898328781128,"z":"293ee063.1298
4","wires":[]},{id":"1bca5108.13bcdff","type":"function","name":"Add
feedback","func":"msg.payload = \"the following message has been displayed on TimeSquAir: \" +
msg.payload ;\n\nreturn
msg;","outputs":1,"noerr":0,"x":601.9150238037109,"y":698.9898557662964,"z":"293ee063.1298
4","wires":[[{"id":"f13be8e5.987dd8"}]],{"id":"2ddad3d1.26944c","type":"set","property":"payload","val
ue":"Not a
command","name":"","x":504.6649932861328,"y":926.4899091720581,"z":"293ee063.12984","wi
res":[[{"id":"271a0551.9a321a"}]],{"id":"1782d23b.8d277e","type":"mqtt
in","name":"","topic":"bot/radio_on","broker":"MQTT.Localhost","x":149.9150161743164,"y":510.
9898862838745,"z":"293ee063.12984","wires":[[{"id":"af7eab74.e14f38"}]],{"id":"8499c49.fb6dd38","t
ype":"function","name":"Send payload","func":"\nvar text = msg.payload ;\n\nmsg.payload = {}
;\nif(msg.inPayload !== undefined)\n  msg.payload.recipient = msg.inPayload.sender.id ;\nelse
if(msg.sender !== undefined)\n  msg.payload.recipient = msg.sender.id ;\nelse\n
msg.payload.recipient = context.global.chat_id ;\n\nmsg.payload.text = text ;\n\nreturn
msg;","outputs":1,"noerr":0,"x":330.29759216308594,"y":823.6665449142456,"z":"293ee063.129
```

```
84","wires":[[{"a52e9a70.e53f58"}]],{"id":"a52e9a70.e53f58","type":"json","name":"","x":468.2975
311279297,"y":823.6664838790894,"z":"293ee063.12984","wires":[[{"26e30f09.7f4ae"}]],{"id":"26
e30f09.7f4ae","type":"mqtt
out","name":"","topic":"bot/command","qos":"","retain":"","broker":"MQTT.Localhost","x":613.29
75921630859,"y":823.6665449142456,"z":"293ee063.12984","wires":[]},{"id":"271a0551.9a321a",
"type":"mqtt
out","name":"","topic":"bot/feedback","qos":"","retain":"","broker":"MQTT.Localhost","x":701.04
75311279297,"y":926.1666669845581,"z":"293ee063.12984","wires":[]},{"id":"f13be8e5.987dd8",
"type":"mqtt
out","name":"","topic":"bot/feedback","qos":"","retain":"","broker":"MQTT.Localhost","x":784.79
75311279297,"y":698.4166746139526,"z":"293ee063.12984","wires":[]},{"id":"d90a041c.e45e68",
"type":"mqtt
in","name":"","topic":"bot/feedback","broker":"MQTT.Localhost","x":160.29759216308594,"y":82
3.6665449142456,"z":"293ee063.12984","wires":[[{"8499c49.fb6dd38"}]],{"id":"1e571074.7304d",
"type":"mqtt
in","name":"","topic":"bot/radio_off","broker":"MQTT.Localhost","x":149.79757690429688,"y":56
7.1666746139526,"z":"293ee063.12984","wires":[[{"3b2bc0c3.68b42"}]],{"id":"3e869194.217c8e",
"type":"Open","x":433.7141571044922,"y":509.3333387374878,"z":"293ee063.12984","wires":[[{"
2d47f09f.872a2"}]],{"id":"3b2bc0c3.68b42","type":"Close","x":357.2141571044922,"y":562.33333
87374878,"z":"293ee063.12984","wires":[[{"2d47f09f.872a2"}]],{"id":"2d47f09f.872a2","type":"sou
nd","name":"Radio
Phenix","sound":"http://live.radio-campus.org:8000/caen.mp3","x":608.7141571044922,"y":531.3
333387374878,"z":"293ee063.12984","wires":[[[]]],{"id":"af7eab74.e14f38","type":"function","n
ame":"","func":"msg.payload = null;\n\nreturn
msg;","outputs":1,"noerr":0,"x":288.71421813964844,"y":510.33327770233154,"z":"293ee063.12
984","wires":[[{"3e869194.217c8e"}]],{"id":"2a851773.081fd8","type":"mqtt
out","name":"","topic":"","qos":"0","retain":"","broker":"MQTT.Localhost","x":1001.71423339843
75,"y":415.3333158493042,"z":"293ee063.12984","wires":[]},{"id":"f42a08fa.6d50f8","type":"mq
t
out","name":"","topic":"","qos":"0","retain":"","broker":"MQTT.Localhost","x":999.914871215820
3,"y":464.98991680145264,"z":"293ee063.12984","wires":[]},{"id":"82824e96.8e8d8","type":"go",
"name":"","x":215.15237426757812,"y":167.34286785125732,"z":"293ee063.12984","wires":[[{"36
a00eb8.3d1e52"}]],{"id":"36a00eb8.3d1e52","type":"function","name":"","func":"msg.payload =
\"jedi/request/bot\" +context.global.jedi.token ;\nreturn
msg;","outputs":1,"noerr":0,"x":373.1523742675781,"y":167.34286785125732,"z":"293ee063.129
84","wires":[[{"53eae1c9.a6574"}]],{"id":"53eae1c9.a6574","type":"display","active":true,"x":539.1
523742675781,"y":166.34286785125732,"z":"293ee063.12984","wires":[]},{"id":"3467422d.d3f9e
e","type":"info","name":"Click \"Go\" to get the MQTT router topic in the Debug
panel","info":"","icon":"comment.png","langs":{"key":"en-US","title":"Click \"Go\" to get the
MQTT router topic in the Debug
panel","body":"","isDefault":true}},{"currentLanguage":"fr-FR","x":269.0095520019531,"y":128.485
7177734375,"z":"293ee063.12984","wires":[]},{"id":"d8e4f1e3.5c42d","type":"info","name":"Laun
ch this flow to link automatically the Telegram commands from your BOT to a MQTT
service","info":"","icon":"comment.png","langs":{"key":"en-US","title":"Launch this flow to link
automatically the Telegram commands from your BOT to a MQTT
service","body":"","isDefault":true}},{"currentLanguage":"fr-FR","x":381.38092041015625,"y":207,"
z":"293ee063.12984","wires":[]}]}
```