# **ΕΓΚΑΤΑΣΤΑΣΗ APACHE SERVER στο Raspberry Pi 4**

Για να μπορέσουμε να εκτελέσουμε την εφαρμογή μας, θα πρέπει να εγκαταστήσουμε τον Apache Web Server και την PHP. Για να γίνει αυτό, ανοίγουμε ένα τερματικό και δίνουμε τις ακόλουθες εντολές:

**1ον Ενημέρωση του Λ.Σ.**

sudo apt-get update

sudo apt-get upgrade

sudo apt-get update

**2ον Εγκατάσταση Apache Web Server:**

sudo apt-get install apache2 -y

**3ον Δικαιώματα στα αρχεία του server:**

sudo chown -R pi:www-data /var/www/html/

sudo chmod -R 770 /var/www/html/

**4ον Έλεγχος λειτουργίας του Apache:**

wget -O check\_apache.html <http://127.0.0.1>

**5ον Αποθήκευση αρχείου λειτουργίας Apache:**

cat ./check\_apache.html

**6ον Φάκελος εγκατάστασης αρχείων για site:**

**/var/www/html**

**Όταν θέλουμε να δημιουργήσουμε Ιστοσελίδες και το περιεχόμενο τους, τα τοποθετούμε σε αυτόν το γονικό φάκελο για αρχεία Ιστοσελίδων**

# **PHP Installation – Εγκατάσταση PHP στο Raspberry Pi**

**1ον Εγκατάσταση PHP :**

sudo apt install php php-mbstring

**2ον Δημιουργία αρχείου PHP:**

echo "<?php phpinfo ();?>" > /var/www/html/index.php

**MySQL DataBase for our server**

**1ον Εγκατάσταση MySQL :**

sudo apt-get install mysql-server php-mysql

or

sudo apt-get install mariadb-server

**2ον Έλεγχος λειτουργίας ότι εγκαταστάθηκε η MySQL:**

sudo mysql --user=root

**3ον Έλεγχος χρήστη για την Βάση Δεδομένων μας, όπου υπάρχει η λέξη ‘password’αλλάζουμε με το δικό μας password για τον χρήστη root**

DROP USER 'root'@'localhost';

CREATE USER 'root'@'localhost' IDENTIFIED BY 'password';

GRANT ALL PRIVILEGES ON \*.\* TO 'root'@'localhost';

Save Changes to DataBase:

FLUSH PRIVILEGES;

**4ον Προσθήκη του PhpMyAdmin:**

sudo apt install phpmyadmin

**5ον Έλεγχος εγκατάσταης PhpMyAdmin:**

<http://127.0.0.1/phpmyadmin>

sudo ln -s /usr/share/phpmyadmin /var/www/html/phpmyadmin

<http://127.0.0.1/phpmyadmin>

**6ον Για να έχουμε πρόσβαση στο Web Server μας απόμακρυσμένα θα πρέπει να δημιουργήσουμε ένα DySNS name και να το κάνουμε port forwarding:**

## **Install Node-Red on Raspberry Pi 4**

**1ον Κάνουμε Εγκατάσταση του Node – RED σβήνοντας την έκδοση που έχει το Raspberry Pi και εγκαθιστώντας ξανά από την αρχή μια νέα:**

bash <(curl -sL https://raw.githubusercontent.com/node-red/linux-installers/master/deb/update-nodejs-and-nodered)

**Εικόνα που περιέχει κείμενο

Περιγραφή που δημιουργήθηκε αυτόματα**

### **Running locally**

Για να το τρέξουμε τοπικά στο μηχάνημα μας, μπορούμε να χρησιμοποιήσουμε την εντολή σε command prompt **node-red** , για να τρέξουμε το Node-RED σε ένα τερματικό. Μπορούμε να σταματήσουμε την διαδικασία του, πατώντας τα πλήκτρα Ctrl – C ή κλείνοντας απλά το τερματικό παράθυρο του command prompt.

Λόγω της περιορισμένης μνήμης του Raspberry Pi, θα χρειαστεί να ξεκινήσετε το Node-RED με ένα επιπλέον όρισμα για να πείτε στην υποκείμενη διαδικασία Node.js να ελευθερώσει τη μη χρησιμοποιημένη μνήμη νωρίτερα.

Για να γίνει αυτό μπορούμε να χρησιμοποιήσουμε εναλλακτικά την εντολή στο τερματικό παράθυρο node-red-pi και το όρισμα max-old-space-size

node-red-pi --max-old-space-size=256

### **Running as a service**

Το script εγκατάστασης για το Pi ως υπηρεσία. Αυτό σημαίνει ότι μπορεί να τρέχει στο παρασκήνιο και να μπορεί να ξεκινήσει αυτόματα κατά την εκκίνηση.

Οι ακόλουθες εντολές παρέχονται για να λειτουργούν με την υπηρεσία:

* node-red-start - this starts the Node-RED service and displays its log output. Pressing Ctrl-C or closing the window does not stop the service; it keeps running in the background
* node-red-stop - this stops the Node-RED service
* node-red-restart - this stops and restarts the Node-RED service
* node-red-log - this displays the log output of the service

You can also start the Node-RED service on the Raspbian Desktop by selecting the Menu -> Programming -> Node-RED menu option.

### **Autostart on boot**

Αν θέλουμε να ξεκινά ως υπηρεσία με το boot (εκκίνηση) του Raspberry Pi ή με την επανεκκίνηση, μπορούμε να ενεργοποιήσουμε την υπηρεσία να ξεκινά με την εκκίνηση τρέχοντας στη γραμμή εντολών του τερματικού την εξής εντολή:

sudo systemctl enable nodered.service

Για να απενεργοποιήσουμε την υπηρεσία, τρέχουμε την εντολή στο command prompt του τερματικού:

sudo systemctl disable nodered.service

### **Opening the editor**

Once Node-RED is running you can access the editor in a browser.

If you are using the browser on the Pi desktop, you can open the address: [http://localhost:1880](http://localhost:1880/).

We recommend using a browser outside of the PI and pointing it at Node-RED running on the Pi. However you can use the built in browser and if so we recommend Chromium or Firefox-ESR and *not* Epiphany

When browsing from another machine you should use the hostname or IP-address of the Pi: http://<hostname>:1880. You can find the IP address by running hostname -I on the Pi.

# **Securing Node-RED**

<https://nodered.org/docs/user-guide/runtime/securing-node-red>

## **Εγκατάσταση Mosquitto στο Raspberry Pi**

sudo apt install -y mosquitto mosquitto-clients

Ενεργοποιούμε το service:

sudo systemctl enable mosquitto.service

Αν θέλουμε να απενεργοποιήσουμε το service γράφουμε:

sudo systemctl stop mosquitto.service

sudo systemctl disable mosquitto.service

Έλεγχος έκδοσης

mosquitto -v

Επιστρέφει 1.4.10

Αν δεν βάλουμε το service και θέλουμε να τρέχει ως δαίμονας στο background γράφουμε:

mosquitto -d

Συνδρομή σε Topic

mosquitto\_sub -d -t testTopic

Δημοσίευση στο Topic

Ανοίγουμε ένα ακόμη τερματικό και γράφουμε:

mosquitto\_pub -d -t testTopic -m "Hello world!"

Τότε στο πρώτο τερματικό θα δούμε:

Client mosqsub/21899-smart\_ras sending PINGREQ

Client mosqsub/21899-smart\_ras received PINGRESP

Client mosqsub/21899-smart\_ras received PUBLISH (d0, q0, r0, m0,

'testTopic', ... (12 bytes))

Hello world!

Μπορούμε να ανοίξουμε και τρίτο τερματικό κάνοντας συνδρομή στο ίδιο topic. Αν από το 2ο τερματικό κάνουμε δημοσίευση, τότε αυτή θα εμφανιστεί στα τερματικά 1 και 3.

## **Voice recognition Google**

Για Raspberry PI:

sudo apt-get install libportaudio0 libportaudio2 libportaudiocpp0 portaudio19-dev

Για Debian i386:

sudo apt-get install libportaudio2 libportaudiocpp0 portaudio19-dev

sudo apt-get install flac

Για Python 2.x

pip install pyaudio

**Για Python 3.x**

**pip3 install pyaudio**

**pip3 install SpeechRecognition**

Αρχικά φτιάχνω ένα πρόγραμμα με όνομα test.py το οποίο εμφανίζει τις συσκευές ηχογράφησης:

import speech\_recognition as sr

for index, name in enumerate(sr.Microphone.list\_microphone\_names()):

print("Microphone with name \"{1}\" found for

`Microphone(device\_index={0})`".format(index, name))

Με το alsamixer ρυθμίζω επίπεδα για capture. Από το audacity ελέγχω την ένταση και το ποσοστό

του θορύβου.

Text To Speech (T.T.S)

import os, subprocess, urllib

def getGoogleSpeechURL(phrase):

googleTranslateURL = "http://translate.google.com/translate\_tts?ie=UTF-8&client=tw-ob&tl=el&"

parameters = {'q': phrase}

data = urllib.parse.urlencode(parameters)

googleTranslateURL = "%s%s" % (googleTranslateURL,data)

return googleTranslateURL

def speakSpeechFromText(phrase):

googleSpeechURL = getGoogleSpeechURL(phrase)

print(googleSpeechURL)

#Raspberry PI

#subprocess.call(["mplayer",googleSpeechURL], shell=False, stdout=subprocess.PIPE, stderr=subprocess.PIPE)

#Debian i386

subprocess.Popen(["mplayer", "-really-quiet", googleSpeechURL], shell=False, stdout=subprocess.PIPE,

stderr=subprocess.PIPE)

speakSpeechFromText("Καλημέρα παιδιά. Τι κάνετε;")

Εναλλακτικά για offline μπορώ να χρησιμοποιήσω το espeak το οποίο μιλάει ελληνικά με

ρομποτική φωνή.

sudo apt-get install espeak