# Allgemein

WebExpress ist ein leichtgewichtiger Webserver, welcher für den Einsatz in leistungsarmen Umgebungen (z.B. Rasperry PI) optimiert wurde. Durch die Bereitstellung eines leistungsfähigen PlugIn-Systems und einer umfassenden API, lassen sich Webanwendungen einfach und schnell in einer .Net-Sprache (z.B. C#) entwickeln.

# Software

Die Software steht als Open-Source zur freien Verfügung. Die Softwarequellen können über <https://github.com/ReneSchwarzer/WebExpress> bezogen werden.

Verwendete Bibliotheken:

https://getbootstrap.com/

https://www.chartjs.org

https://jquery.com/

https://summernote.org/

https://popper.js.org/

# Einrichtung Raspberry PI

|  |  |  |
| --- | --- | --- |
| **1.** | Raspbian light auf SD-Karte kopieren. | <https://downloads.raspberrypi.org/imager/imager.exe> |
| **2.** | Anmelden als: pi Passwort rasperry |  |
| **3.** | Raspberry pi konfigurieren   * Password ändern: pi Password hallo * WLAN einrichten: SSID=xxxx, KEY=xxxxx * HOST ändern: webexpress * Zeitzone ändern | pi@raspberrypi:~ $ sudo raspi-config |
| **4.** | SSH aktivieren: Leere Datei mit den Namen ssh in der BOOT-Partition erstellen. | pi@webexpress:cd /boot  pi@webexpress:sudo touch ssh |
| **5.** | ssh auf Windows starten | ssh pi@webexpress |
| **6.** | MC installieren (optional) | pi@webexpress:~ $ sudo apt-get install mc -y |
| **7.** | Alias erstellen in /etc/profile | alias ll='ls -l' |
| **8.** | Net Core installieren (siehe https://dotnet.microsoft.com/download/linux-package-manager/debian10/runtime-current) | pi@webexpress:~ $ wget https://download.visualstudio.microsoft.com/download/pr/ccbcbf70-9911-40b1-a8cf-e018a13e720e/03c0621c6510f9c6f4cca6951f2cc1a4/dotnet-sdk-3.1.201-linux-arm.tar.gz  pi@webexpress:~ $ sudo mkdir /usr/share/dotnet-sdk  pi@webexpress:~ $ sudo tar zxf dotnet-sdk-3.1.201-linux-arm.tar.gz -C /usr/share/dotnet-sdk/ |
| **9.** | Git installieren | pi@webexpress:~ $ sudo apt-get install git |
| **10.** | Repository klonen | pi@webexpress:~ $ git clone <https://github.com/ReneSchwarzer/WebExpress.git> |
| **11.** | Rechte erteilen | pi@webexpress:~ $ chmod +x webexpress/src/core/\*.sh |

# Deploy

|  |  |  |
| --- | --- | --- |
| **1.** | Verzeichnis wechseln | pi@webexpress:~ $ cd webexpress/src/core/ |
| **2.** | Repository aktuallisieren  (optional) | pi@webexpress:~/webexpress $ git pull  (optional) pi@webexpress:~/webexpress/src/core $ git reset --hard HEAD  pi@webexpress:~/webexpress/src/core $ git pull |
| **3.** | Skript deploy.sh ausführen | pi@webexpress:~/webexpress/src/core $ sudo chmod +x \*.sh  pi@webexpress:~/webexpress/src/core $ sudo ./deploy.sh |
| **4.** | Reboot | pi@webexpress:~ $ sudo reboot |

# Update

|  |  |  |
| --- | --- | --- |
| **1.** | Raspbian light aktualisieren | pi@webexpress:~ $ sudo raspi-config |
|  |  |  |

# Reverse Proxy

|  |  |  |
| --- | --- | --- |
| **1.** | Apache Webserver installieren | pi@webexpress:~ $ sudo apt install apache2 -y |
| **2.** | Apache Proxymodule | pi@webexpress:~ $ sudo a2enmod proxy proxy\_http ssl |
|  | Seiten-Konfiguration „vila.conf“ in /etc/apache2/sites-available anlegen | <VirtualHost \*:443>  ServerName <crws.org>  ServerAlias <www.domaincrws.org>  SSLProxyEngine On  ProxyPass / http://sub.domain.tld//vila/vila  ProxyPassReverse / http://vila/vila  SSLEngine on  SSLCertificateFile /etc/apache2/cert/public.pem  SSLCertificateKeyFile /etc/apache2/cert/privkey.pem  SSLCertificateChainFile /etc/apache2/cert/ca.pem  </VirtualHost> |
|  |  | a2ensite vila |
|  |  | pi@webexpress:~ $ sudo systemctl restart apache2 |

# Einkaufsliste

|  |  |  |
| --- | --- | --- |
| **Item** | **Link** | **Preis** |
| Raspberry Pi | https://www.amazon.de/gp/product/B01CD5VC92/ref=ppx\_yo\_dt\_b\_asin\_title\_o00\_s00?ie=UTF8&psc=1 | 40€ |