

1. Betriebssystem

12

001	Unterschiede Raspberry Pi 4 – Raspberry Pi 400.	12
002	Betriebssystem mit dem Raspberry Pi Imager installieren	13
003	Verschiedene Raspberry-Pi-OS-Varianten	14
004	Raspberry Pi 4 einschalten	14
005	Raspberry Pi 400 einschalten.	14
006	Raspberry Pi erstmals einrichten	15
007	WLAN einrichten	15
008	Standardpasswort ändern	16
009	Updates herunterladen und installieren	16
010	Warnsymbole auf dem Bildschirm	16
011	Die Tastatur des Raspberry Pi 400	17
012	Wichtige Unterschiede gegenüber Windows bei Dateinamen und Verzeichnissen	17
013	Speicherkarte komplett sichern	18
014	Speicherkarte aus Image-Datei duplizieren	19
015	Speicherkarte ohne PC duplizieren	19
016	Weitere Betriebssysteme im Raspberry Pi OS Imager	20
017	Andere Betriebssysteme installieren	20

2. Praxiseinsatz

21

018	Der Raspberry-Pi-OS-Desktop	21
019	Der Dateimanager PCManFM	21
020	Dateimanager im Zwei-Fenster-Modus	22
021	Wichtige Befehle auf der Linux-Kommandozeile	23
022	Der Superuser root	24
023	Dateimanager mit root-Rechten nutzen	24
024	USB-Sticks an- und abdocken	25
025	Desktop auf zwei Monitoren einrichten	26
026	Eigene Hintergrundbilder für den Desktop	27
027	Fan-Art als Hintergrundbild	30
028	Raspberry Pi ausschalten und neu starten	30
029	Taskleiste am unteren Bildschirmrand	30
030	Blaue Fensterrahmen zurückholen	31

031	Texteditoren für Konfigurationsdateien.....	32
032	Bildschirmschoner abschalten	37
033	Desktop-Verknüpfungen für Programme aus dem Startmenü anlegen	38
034	Die wichtigsten Standardverzeichnisse in Linux.....	39
035	LXTerminal mit weißem Hintergrund	41
036	Uhrzeit ohne Internetverbindung einstellen.....	41
037	Freien Speicherplatz anzeigen	42
038	Große Verzeichnisse finden	42
039	Speicherkarte aufräumen, wenn der Raspberry Pi nicht mehr bootet...	43
040	Betriebssystem und vorinstallierte Programme updaten	44
041	Screenshots vom Raspberry-Pi-Desktop erstellen	44
042	Wissenschaftlicher Taschenrechner.....	45
043	Drucker einrichten.....	46
044	Drucken aus einem Linux-Programm	49
045	Netzwerkdrucker mit CUPS einrichten.....	49

3. Softwaretipps

53

046	Empfohlene Programme der Raspberry Pi Foundation	53
047	Software als Linux-Paket installieren.....	53
048	Software über grafische Paketverwaltung installieren.....	54
049	Archive packen und entpacken	55
050	Ziffernblock aktivieren.....	55
051	Firefox synchronisiert mit dem PC	56
052	Werbe-Pop-ups in Firefox blockieren	56
053	Moderne Browser-Alternative Vivaldi	57
054	Werbung in Vivaldi blockieren	58
055	Standardbrowser ändern	58
056	DNT (Do Not Track) – ein Politikum im Internet	59
057	LibreOffice nachinstallieren	59
058	Deutsches Sprachpaket zu LibreOffice nachinstallieren	60
059	Tipps zu LibreOffice.....	61
060	Microsoft-Office-Dokumente mit LibreOffice nutzen.....	61
061	Texte in LibreOffice Writer formatieren	62
062	Zahlen in LibreOffice Calc mit Grafiken veranschaulichen	62

063	Funktionen in LibreOffice Calc	63
064	Grafiken in LibreOffice Draw bearbeiten	64
065	Datenbanken in LibreOffice Base	64
066	Präsentationen in LibreOffice Impress	64
067	Mathematische Formeln in LibreOffice Math	65
068	PDF-Dokumente auf dem Raspberry Pi betrachten	65
069	PDF-Dokumente in LibreOffice bearbeiten	65
070	Schlanke Alternative: AbiWord	66
071	Diaschau im Vollbildmodus	66
072	Einfache Bildbearbeitung mit Mirage	67
073	Professionelle Bildbearbeitung mit GIMP	67
074	GIMP-Erweiterungen nutzen	68
075	E-Mail mit Claws Mail	69
076	GMX und Web.de in Claws Mail nutzen	71
077	Filterregeln für eingehende E-Mails in Claws Mail	72
078	Das Adressbuch in Claws Mail	73
079	HTML-Mails in Claws Mail	73
080	E-Mails mit Thunderbird	74
081	Gmail in Thunderbird nutzen	74
082	GMX und Web.de in Thunderbird nutzen	75
083	Externe Inhalte in Thunderbird blockieren	76
084	Filterregeln für eingehende E-Mails in Thunderbird	77
085	Junk-Filter in Thunderbird	78
086	Thunderbird-Kalender mit Google-Kalender synchronisieren	79
087	Bildschirmlupe	81
088	Videos abspielen	82
089	YouTube-Videos mit H.264 flüssiger abspielen	82
090	ASCII-Art	82
091	Pi Presents für Präsentationen	84
092	Eigene Präsentationen mit Pi Presents erstellen	85
093	Minecraft	87
094	Windows-Spieleklassiker für Raspberry Pi umgesetzt	89
095	Textadventures – interaktive Fiktion	89
096	Klassische Point-and-Click-Adventures	90
097	Code the classics	92
098	DOSBox – PC-Emulator	93
099	Der Emulator Sinclair ZX Spectrum	94

100	MAME – Sammlung von Emulatoren	95
101	FS-UAE – Amiga-Spiele auf dem Raspberry Pi 4 und 400.....	96
102	Spiele für FS-UAE herunterladen und installieren	97
103	Eigene Amiga-Spiele in FS-UAE spielen	98
104	RetroPie einrichten	99
105	Spiele in RetroPie installieren	99
106	Recalbox	100
107	KODI-Mediacenter mit LibreELEC einrichten	101
108	Fernbedienung des Fernsehers über HDMI-CEC nutzen	103
109	Medienquellen hinzufügen	103
110	Musik im KODI-Mediacenter abspielen	104
111	Fotos im KODI-Mediacenter	104
112	Fotos aus Cloud-Speichern betrachten	105
113	Videos in KODI abspielen	105
114	Webradio in KODI	106
115	TV-Mediatheken in KODI nutzen	106
116	YouTube in KODI nutzen	107
117	Wetter im KODI-Mediacenter	108
118	Tastaturbelegung auf Deutsch umstellen	108
119	Smartphone als Fernbedienung für KODI	109
120	KODI im Browser auf einem anderen PC fernsteuern	110

4. Hardware und Netzwerk

111

121	Was unterscheidet die verschiedenen Raspberry-Pi-Modelle?	111
122	Die Anschlüsse des Raspberry Pi 4	111
123	Die Anschlüsse des Raspberry Pi 400	112
124	Netzteil anschließen	112
125	Blaue und schwarze USB-Anschlüsse	112
126	Tastatur und Maus persönlich einstellen	113
127	Audio einrichten	113
128	MP3 aus dem Dateimanager abspielen	114
129	DVD, Blu-ray, SVCD und Audio-CD abspielen	114
130	Raspberry-Pi-Modell anzeigen	115
131	Mehr Leistung durch Speichertuning	115
132	Aktuelle Speicherauslastung anzeigen	116

133	Taskmanager und Prozessorlast in der Taskleiste anzeigen	117
134	CPU-Temperatur in der Taskleiste.....	118
135	Das Raspberry Pi Touch Display	118
136	Bildschirmtastatur für Touchscreen-Display.....	120
137	Raspberry-Pi-Kamera anschließen.....	121
138	Pi-NoIR-Kamera.....	122
139	Fotografieren mit der Raspberry-Pi-Kamera.....	122
140	Video mit der Raspberry-Pi-Kamera	123
141	USB-Sticks mounten	124
142	Die Linux-Dateisysteme.....	125
143	Komfortable Festplatteneinrichtung mit GParted	126
144	CloudMe auf dem Raspberry Pi nutzen	127
145	Cloud von GMX und Web.de auf dem Raspberry Pi nutzen	128
146	OneDrive-Cloud-Speicher über Rclone verbinden	129
147	IP-Adresse und MAC-Adresse des Raspberry Pi anzeigen	131
148	Feste IP-Adresse über den Router vergeben	132
149	Feste IP-Adresse auf dem Raspberry Pi eintragen	132
150	Vom PC über SSH auf dem Raspberry Pi anmelden.....	133
151	Daten per SSH vom PC auf den Raspberry Pi kopieren – und umgekehrt.....	134
152	Dateiübertragung mit dem Total Commander	135
153	Dateiverwaltung mit dem Midnight Commander	135
154	SSH-Client für Smartphones und Tablets.....	136
155	Windows-PC über VNC vom Raspberry Pi fernsteuern	137
156	Raspberry Pi vom Windows-PC über VNC steuern	139
157	Zweiten Raspberry Pi vom Raspberry Pi 400 über VNC steuern	142
158	Notebook als Tastatur und Monitor für Raspberry Pi nutzen	142
159	VNC vom Tablet verbinden.....	143
160	Externe Festplatten für Serverbetrieb mounten.....	145
161	Webserver installieren	145
162	Webserver über No-Ip ins Internet bringen	147
163	Portweiterleitung auf dem Router einrichten	148
164	LionWiki – eigene kleine Wissensdatenbank einrichten	149
165	Fotogalerie auf dem Webserver	151
166	Samba-Server installieren.....	153
167	Samba-Server im Windows-Netzwerk nutzen	155
168	DLNA-Streamingserver MiniDLNA einrichten	157

169	Medien auf Windows-PCs streamen	158
170	Medien auf Smartphones oder Tablets streamen.....	158
171	MiniDLNA-Statusanzeige im Netzwerk	159
172	RaspController – Statusanzeige und Fernsteuerung auf dem Smartphone.....	159
173	Raspcontrol – Statusanzeige im Browser.....	160
174	Pi-hole blockiert Werbung im Netzwerk	161
175	Pi-hole im Netzwerk einrichten.....	164
176	Pi-hole administrieren	165

5. Programmierung

166

177	Python als Taschenrechner.....	166
178	Python-Programme nutzen	166
179	Python-Programme über die Kommandozeile starten.....	167
180	Kompletten Funktionsumfang der Thonny Python IDE freischalten....	167
181	Klassische Python-IDE nachinstallieren	167
182	Python – Ausgabe auf dem Bildschirm	168
183	Python – Variablen.....	169
184	Python – Eingabe durch den Benutzer.....	169
185	Python – Bedingungen mit if.....	169
186	Python – Bedingungen mit and und or verknüpfen	171
187	Python – Schleifen mit for.....	171
188	Python – Schleifen mit while	171
189	Python – Funktionen	172
190	Python – Funktionen mit Parametern.....	172
191	Python – Funktionen mit Rückgabewert.....	172
192	Boolesche Wahr- und Falsch-Werte.....	173
193	Einfaches Programmierbeispiel: 99 Bottles of Beer	173
194	Farbige Texte im Terminalfenster	177
195	Uhrzeiten in Python verarbeiten	178
196	Analoguhr mit Pygame programmieren	179
197	Zusätzliche Python-Pakete nachinstallieren.....	185
198	Programmieren in Scratch 3.....	185
199	Kultspiel Flappy Bird in Scratch 3 nachbauen	188

6. Elektronik

193

200	Elektronik über GPIO anschließen	193
201	Elektronische Schaltungen ohne Löten aufbauen	194
202	GPIO-Port über Flachbandkabel herausführen	195
203	LEDs an die GPIO-Ports anschließen	195
204	Erweiterungsplatinen und bestehende Schaltungen am Raspberry Pi 400 nutzen	198
205	LED-Lauflicht	198
206	Taster am GPIO-Anschluss	200
207	LED-Würfel	201
208	Fußgängerampel mit LEDs und Taster	203
209	LEDs mit PWM dimmen	205
210	PWM mit Python	205
211	RGB-LEDs	208
212	RGB-LEDs in Python steuern	209
213	Farbverlauf auf RGB-LEDs	210
214	Töne mit Piezo-Summer erzeugen	212
215	Servos steuern	213
216	Sieben-Segment-Anzeige zur Darstellung von Ziffern	215
217	Sieben-Segment-Anzeige mit Python ansteuern	217
218	Mehrstellige Sieben-Segment-Anzeigen	219
219	Mehrstellige Sieben-Segment-Anzeige mit Python ansteuern	221
220	Uhr auf der Sieben-Segment-Anzeige	223
221	IP-Adresse auf der Sieben-Segment-Anzeige	225
222	Python-Programme auf dem Raspberry-Pi-OS-Desktop automatisch starten	229
223	LED-Punktmatrix-Anzeigen	229
224	LED-Punktmatrix-Anzeige mit Python ansteuern	231
225	Zeichensatz für LED-Punktmatrix-Anzeige	232
226	Zeichen auf LED-Punktmatrix-Anzeige ausgeben	236
227	Zeichenorientiertes LCD-Display anschließen	239
228	Zeichenorientiertes LCD-Display mit Python ansteuern	241
229	IP-Adresse und Uhr auf dem LCD-Display	244
230	Vierzeiliges LCD-Display ansteuern	246
231	Statusanzeige für Speicherkarte oder USB-Sticks	247
232	Zeichensatz des LCD-Displays anzeigen	250

233	Umlaute durch passende Zeichen ersetzen	251
234	i2c-Geräte adressieren und finden.	252
235	Der Portexpander MCP23017	253
236	Lauflicht mit dem Portexpander (MCP23017)	254
237	CPU-Lastanzeige mit Portexpander (MCP23017)	257
238	Binäruhr mit Portexpander (MCP23017)	259
239	i2c-LCD-Module (MCP23017) ansteuern	263
240	Der Portexpander PCF8574/PCF8574A	267
241	i2c-LCD-Module (PCF8574) ansteuern	269
242	i2c-Zusatzmodule für LCD-Displays ansteuern	271
243	Sensorkontakte selbst bauen	273
244	Sensorkontakte mit Python nutzen.	274
245	Gamepad mit Sensorkontakten für bekannte Spiele	275
246	Die GPIOzero Bibliothek	278
247	GPIO-Ausgänge mit Scratch 3	280
248	GPIO-Ereignisse mit Scratch 3 auswerten	281
249	LED-Würfel mit Scratch	282
250	Scratch-3-OneGPIO-Erweiterung	284
251	LEDs mit der Scratch-3-OneGPIO-Erweiterung schalten	285
252	PWM mit der Scratch-3-OneGPIO-Erweiterung	286
253	Servos mit der Scratch-3-OneGPIO-Erweiterung bewegen	286
254	Erweiterungsplatine Sense HAT	287
255	Der Sense HAT Emulator	288

Am Server ohne grafische Oberfläche werden Festplatten auf klassischem Weg über die Kommandozeile partitioniert und gemountet.

1. Feststellen, unter welchem Namen die angeschlossene Festplatte in der Linux-Verzeichnisstruktur eingetragen ist:

```
sudo fdisk -l
```

In den meisten Fällen wird eine USB-Festplatte physikalisch als `/dev/sda` oder `/dev/sdb` bezeichnet, die erste darauf befindliche Partition als `/dev/sda1` oder `/dev/sdb1`.

2. Prüfen, ob die Festplatte bereits gemountet ist: `mount`
3. Sollte hier zum Beispiel die Festplatte `/dev/sda1` auftauchen, muss sie für eine saubere Neuinstallation zunächst unmountet werden. Der automatisch angelegte Mountpunkt unterhalb von `/media/pi` kann bei jeder Festplatte anders heißen, meistens handelt es sich um lange Zahlenkombinationen:

```
sudo umount /media/pi/009944a0d
```

4. Festplatte mit dem Linux-ext4-Dateisystem formatieren:

```
sudo mkfs.ext4 /dev/sda1
```

5. Festplatte neu mounten:

```
sudo mkdir /media/pi/usb1
```

```
sudo mount /dev/sda1 /media/pi/usb1
```

6. Dem Benutzer `pi` den Besitz an der Festplatte übertragen:

```
sudo chown -R pi /media/pi/usb1
```

7. Mountpunkt in `/etc/fstab` eintragen:

```
sudo nano /etc/fstab
```

Am Ende die neue Festplatte entsprechend der gemounteten Daten eintragen.

<code>proc</code>	<code>/proc</code>	<code>proc</code>	<code>defaults</code>	<code>0</code>	<code>0</code>
<code>PARTUUID=d35b20a0-01</code>	<code>/boot</code>	<code>vfat</code>	<code>defaults</code>	<code>0</code>	<code>2</code>
<code>PARTUUID=d35b20a0-02</code>	<code>/</code>	<code>ext4</code>	<code>defaults,noatime</code>	<code>0</code>	<code>1</code>
<code>/dev/sda1</code>	<code>/media/pi/usb1</code>	<code>ext3</code>	<code>defaults</code>	<code>0</code>	<code>0</code>

Der Raspberry Pi kann zur Entwicklung eigener Webseiten als geräuschloser und sehr stromsparender Webserver im lokalen Netzwerk laufen. Dazu – einfacher als unter Windows – einen Webserver im lokalen Netzwerk direkt auf dem Raspberry Pi 4 oder 400 installieren. Der Webserver **Lighttpd** (www.lighttpd.net) ist eine schlanke Alternative zum mächtigen Apache-Webserver.



1. Basisverzeichnis für den Webserver anlegen:
`sudo mkdir /var/www`
2. Benutzer und Benutzergruppe für den Webserver anlegen:
`sudo addgroup www-data`
`sudo adduser --ingroup www-data www-data`
3. Den neuen Benutzer zum Eigentümer des Verzeichnisses `/var/www` machen:
`sudo chown -R www-data:www-data /var/www`
4. Standardnutzer `pi` der Benutzergruppe `www-data` hinzufügen:
`sudo adduser pi www-data`
5. Der ganzen Gruppe Schreibrecht auf das Datenverzeichnis des Webserver und alle Unterverzeichnisse gewähren:
`sudo chmod -R 775 /var/www`
6. Um die neuen Rechte zu übernehmen, mit dem Menüpunkt **Abmelden/Abmelden** einmal ab- und wieder anmelden.
7. Webserver Lighttpd installieren: `sudo apt install lighttpd`
8. php7.3-Module installieren (kompletten Befehl in einer Zeile eingeben):
`sudo apt install php7.3-fpm php7.3-mbstring php7.3-mysql php7.3-curl php7.3-gd php7.3-xml php7.3-zip php7.3-ldap php7.3-mcrypt php7.3-imagick php7.3-intl php7.3-openssl php7.3-bcmath php7.3-soap php7.3-readline php7.3-sysvshm php7.3-redis php7.3-igmp php7.3-ldap2 php7.3-ldap3 php7.3-ldap4 php7.3-ldap5 php7.3-ldap6 php7.3-ldap7 php7.3-ldap8 php7.3-ldap9 php7.3-ldap10 php7.3-ldap11 php7.3-ldap12 php7.3-ldap13 php7.3-ldap14 php7.3-ldap15 php7.3-ldap16 php7.3-ldap17 php7.3-ldap18 php7.3-ldap19 php7.3-ldap20 php7.3-ldap21 php7.3-ldap22 php7.3-ldap23 php7.3-ldap24 php7.3-ldap25 php7.3-ldap26 php7.3-ldap27 php7.3-ldap28 php7.3-ldap29 php7.3-ldap30 php7.3-ldap31 php7.3-ldap32 php7.3-ldap33 php7.3-ldap34 php7.3-ldap35 php7.3-ldap36 php7.3-ldap37 php7.3-ldap38 php7.3-ldap39 php7.3-ldap40 php7.3-ldap41 php7.3-ldap42 php7.3-ldap43 php7.3-ldap44 php7.3-ldap45 php7.3-ldap46 php7.3-ldap47 php7.3-ldap48 php7.3-ldap49 php7.3-ldap50 php7.3-ldap51 php7.3-ldap52 php7.3-ldap53 php7.3-ldap54 php7.3-ldap55 php7.3-ldap56 php7.3-ldap57 php7.3-ldap58 php7.3-ldap59 php7.3-ldap60 php7.3-ldap61 php7.3-ldap62 php7.3-ldap63 php7.3-ldap64 php7.3-ldap65 php7.3-ldap66 php7.3-ldap67 php7.3-ldap68 php7.3-ldap69 php7.3-ldap70 php7.3-ldap71 php7.3-ldap72 php7.3-ldap73 php7.3-ldap74 php7.3-ldap75 php7.3-ldap76 php7.3-ldap77 php7.3-ldap78 php7.3-ldap79 php7.3-ldap80 php7.3-ldap81 php7.3-ldap82 php7.3-ldap83 php7.3-ldap84 php7.3-ldap85 php7.3-ldap86 php7.3-ldap87 php7.3-ldap88 php7.3-ldap89 php7.3-ldap90 php7.3-ldap91 php7.3-ldap92 php7.3-ldap93 php7.3-ldap94 php7.3-ldap95 php7.3-ldap96 php7.3-ldap97 php7.3-ldap98 php7.3-ldap99 php7.3-ldap100`
9. fastcgi-Modul aktivieren:
`sudo lighty-enable-mod fastcgi`
`sudo lighty-enable-mod fastcgi-php`
10. Konfigurationsdatei des Webserver bearbeiten:
`sudo nano /etc/lighttpd/conf-available/15-fastcgi-php.conf`
Die Datei muss so aussehen:

```
# -*- depends: fastcgi -*-
# /usr/share/doc/lighttpd/fastcgi.txt.gz

fastcgi.server += ( ".php" =>
    (
        "socket" => "/var/run/php/php7.3-fpm.sock",
        "broken-scriptfilename" => "enable"
    )
)
```
11. Alternativ die Datei aus dem Download kopieren:
`sudo cp ~/15-fastcgi-php.conf /etc/lighttpd/conf-available`
12. Die neue Konfiguration in Lighttpd laden:
`sudo service lighttpd force-reload`
13. Einfache Startseite für den Webserver erstellen, die die PHP-Konfiguration überprüft und anzeigt. Diese Datei `index.php` besteht nur aus einer einzigen Zeile:
`<?php phpinfo() ?>`

14. Alternativ die Datei aus dem Download kopieren:

```
cp ~/index.php /var/www/html
```

15. Der Webserver steht nun lokal auf dem Raspberry Pi 4/400 und auch im lokalen Netzwerk zur Verfügung. Auf einem PC die IP-Adresse des Raspberry Pi im Browser eingeben, zum Beispiel: <http://192.168.178.37>

162

Webserver über No-IP ins Internet bringen

Dynamische DNS-Dienste (DDNS) ersparen das manuelle Herausfinden und Weitergeben der eigenen IP-Adresse, um aus der Ferne auf den eigenen Server zugreifen zu können. Hier bekommt man einen virtuellen Servernamen, der regelmäßig mit der eigenen Adresse aktualisiert werden muss.

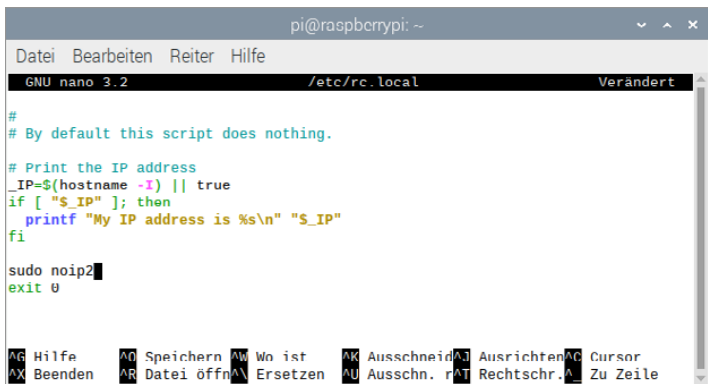
Der bekannteste derartige Dienst, DynDNS, hat seine kostenlosen Angebote leider eingestellt. **No-IP** ist ein interessanter Nachfolger.

1. Kostenloses Benutzerkonto und virtuellen Host bei [no-ip.com](https://noip.com) anlegen.
2. No-IP-Client für Raspberry Pi herunterladen und installieren:

```
wget http://www.noip.com/client/linux/noip-duc-linux.tar.gz
tar xf noip-duc-linux.tar.gz
rm noip-duc-linux.tar.gz
mv noip-* noip
cd noip
sudo make install
```



3. Eventuelle Warnungen können ignoriert werden. Zum Schluss den Log-in-Namen und das Passwort von [no-ip.com](https://noip.com) angeben. Gewünschten (eigenen) Host auswählen.
4. Update-Intervall festlegen. In den meisten Fällen sollte die Standardvorgabe von 30 Minuten ausreichen, da die meisten Internetprovider nur einmal am Tag, üblicherweise nachts, die IP-Adressen neu vergeben.
5. Die Frage *Do you wish to run something at successful update* mit dem vorgeschlagenen **N** beantworten.
6. No-IP-Client starten: `sudo noip2`
Der Webserver auf dem Raspberry Pi ist ab sofort unter dem in der Konfiguration angegebenen Hostnamen von außen zu erreichen.
7. Um den No-IP-Client beim Booten des Raspberry Pi automatisch zu starten, in der Datei `/etc/rc.local` vor der Zeile `exit 0` eine neue Zeile `sudo noip2` einfügen:
`sudo nano /etc/rc.local`



```
pi@raspberrypi: ~  
Datei Bearbeiten Reiter Hilfe  
GNU nano 3.2 /etc/rc.local Verändert  
#  
# By default this script does nothing.  
# Print the IP address  
_IP=$(hostname -I) || true  
if [ "$_IP" ]; then  
  printf "My IP address is %s\n" "$_IP"  
fi  
  
sudo noip2  
exit 0  
␣ Hilfe ␣ Speichern ␣ Wo ist ␣ Ausschneid ␣ Ausrichten ␣ Cursor  
␣ Beenden ␣ Datei öffn ␣ Ersetzen ␣ Ausschn. ␣ Rechtschr. ␣ Zu Zeile
```

163 Portweiterleitung auf dem Router einrichten

Der Router muss noch wissen, auf welchen Computer im Netzwerk eine von außen kommende Anfrage nach einer Webseite weitergeleitet werden soll.

- 1. Routerkonfiguration im Browser aufrufen.
- 2. Einen Bereich **Port Mapping** oder **Portweiterleitung** suchen.
- 3. Weiterleitung für den HTTP-Port 80 auf die lokale IP-Adresse des Raspberry Pi eintragen.
- 4. Auf der FRITZ!Box unter **Internet/Freigaben** mit einem Klick auf **Gerät für Freigaben hinzufügen** den Raspberry Pi auswählen und über **Neue Freigabe** die Freigaben anlegen.

Die Tabelle zeigt die verwendeten Ports für die wichtigsten Internetdienste:

Dienst	Port(s)	Dienst	Port(s)
FTP	20/21	SFTP	115
SSH	22	NTP	123
Telnet	23	IMAP	143
SMTP	25/587	IRC	194
WHOIS	43	HTTPS	443
DNS	53	Citadel	504
HTTP	80	Rsync	873
POP3	110		

LionWiki – eigene kleine Wissensdatenbank einrichten

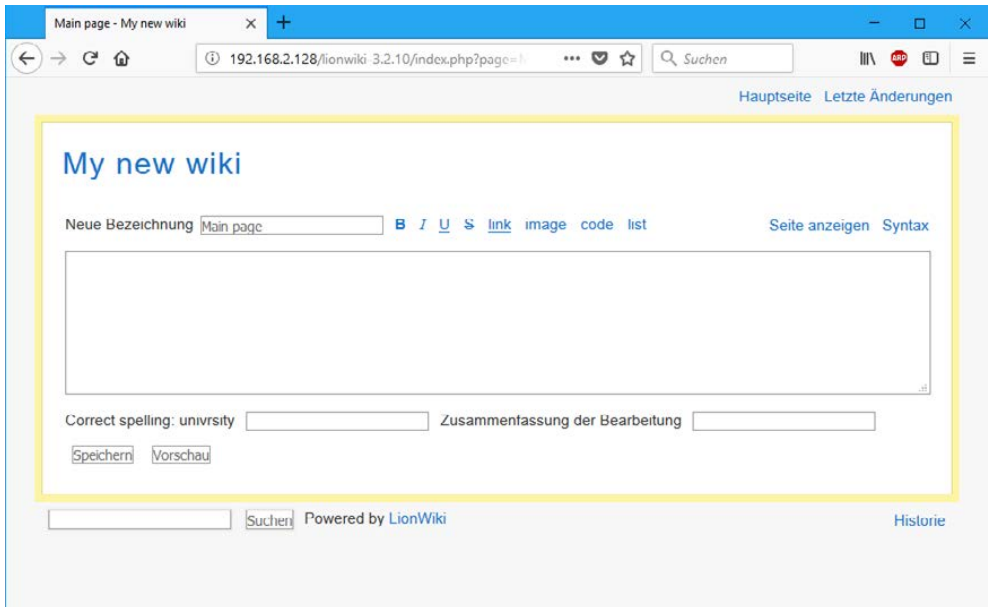
LionWiki (lionwiki2t.sourceforge.io) ist ein System zum Erstellen von Informationsseiten im Wikipedia-Stil, das mit sehr geringer Systemleistung zurechtkommt und nicht einmal eine Datenbank auf dem Webserver benötigt.

1. LionWiki installieren (Webserver muss auf dem Raspberry Pi laufen):

```
wget http://lionwiki.0o.cz/download/3.2.11/lionwiki-3.2.11.zip
sudo unzip lionwiki-3.2.11.zip -d /var/www/html
sudo mv /var/www/html/lionwiki-3.2.11 /var/www/html/lionwiki
sudo chmod -R 777 /var/www/html/lionwiki/var
```

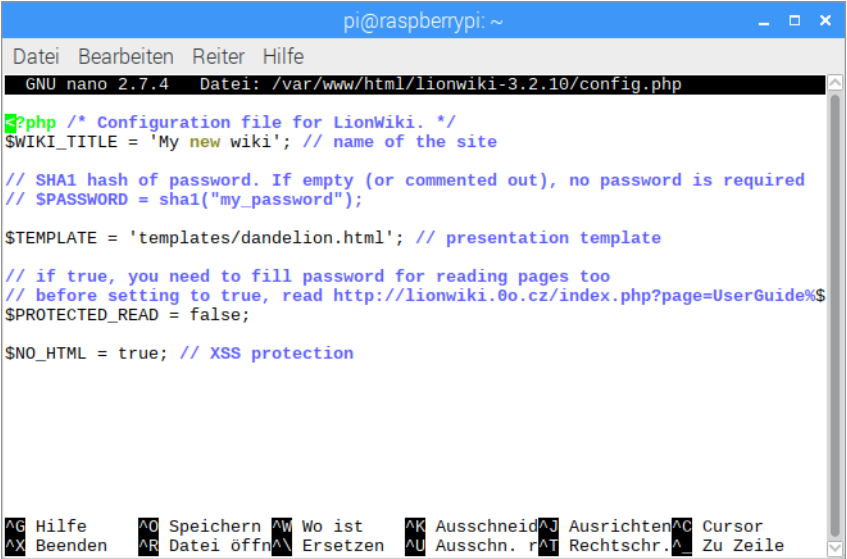


2. LionWiki im Webbrowser auf dem PC aufrufen. Dabei die IP-Adresse durch die des Raspberry Pi ersetzen: <http://192.168.2.124/lionwiki>



3. Seitentitel angeben und den Text der neuen Seite in das große Editorfeld schreiben. Zum Einfügen von Links, Bildern und Listenelementen stehen Symbole zur Verfügung.
4. Bei der ersten Bearbeitung muss ganz unten eine einfache Frage beantwortet werden. Es ist eine Sicherheitsabfrage, die automatisierte Spam-Kommentare verhindern soll. Sie erscheint immer mal wieder, aber nicht bei jeder Bearbeitung.
5. Nach dem Speichern ist die neue Seite über den Link [Seite anzeigen](#) oben rechts zu sehen.

6. Zusätzlich zum vorgegebenen gelb-weißen Template *Dandelion* liefert LionWiki im Verzeichnis *templates* der LionWiki-Installation weitere Templates zur Auswahl. Das gewünschte Template in der Datei */var/www/html/lionwiki/config.php* in der Zeile *\$TEMPLATE* eintragen.



```
pi@raspberrypi: ~
Datei Bearbeiten Reiter Hilfe
GNU nano 2.7.4 Datei: /var/www/html/lionwiki-3.2.10/config.php

<?php /* Configuration file for LionWiki. */
$WIKI_TITLE = 'My new wiki'; // name of the site

// SHA1 hash of password. If empty (or commented out), no password is required
// $PASSWORD = sha1("my_password");

$TEMPLATE = 'templates/dandelion.html'; // presentation template

// if true, you need to fill password for reading pages too
// before setting to true, read http://lionwiki.0o.cz/index.php?page=UserGuide%$
$PROTECTED_READ = false;

$NO_HTML = true; // XSS protection

AG Hilfe      AO Speichern  AW Wo ist     AK Ausschneid  AJ Ausrichten  AC Cursor
AX Beenden    AR Datei öffn  AE Ersetzen  AU Ausschn. r  AT Rechtschr.  AA Zu Zeile
```

Die Tabelle zeigt die wichtigsten Syntaxelemente der Wiki-Beschreibungssprache:

Syntax	Bedeutung
!	Überschrift H2
!!	Überschrift H3
!!!	Überschrift H4
*	Liste 1.Ebene
**	Liste 2.Ebene
***	Liste 3.Ebene
#	Nummerierte Liste 1.Ebene
##	Nummerierte Liste 2.Ebene
###	Nummerierte Liste 3.Ebene
""Text""	Fett (drei Apostrophe)

Syntax	Bedeutung
"Text"	Kursiv (zwei Apostrophe)
""Text""	Fett + kursiv (fünf Apostrophe)
'--Text--'	Durchgestrichen (Apostroph und 2x -)
'__Text__'	Unterstrichen (Apostroph und 2x _)
{small}Text{/small}	Kleine Schrift
{sup}Text{/sup}	Hochgestellt (Exponent)
{sub}Text{/Sub}	Tiefgestellt (Index)
[Link]	Link auf andere Wiki-Seite, externe Links werden automatisch erkannt
[Linktext Link]	Text mit hinterlegtem Link auf andere Wiki-Seite

165

Fotogalerie auf dem Webserver



Die Software **PhotoShow** (github.com/thibaud-rohmer) ermöglicht interaktive Fotogalerien auf einem Webserver und benötigt dabei so wenig Leistung, dass ein Raspberry Pi mit Lighttpd als Webserver völlig ausreicht.

Ein weiterer Vorteil gegenüber anderen Lösungen ist, dass PhotoShow lediglich PHP benötigt, aber keine Datenbank. Die Fotos können interaktiv im Browser hochgeladen oder einfach in die entsprechenden Verzeichnisse kopiert werden.

1. PhotoShow herunterladen und installieren:

```
sudo apt-get install php7.0-xml
wget github.com/thibaud-rohmer/PhotoShow/archive/master.zip
sudo unzip ~/master.zip
sudo mv ~/PhotoShow-master /var/www/html/photoshow
sudo chown -R www-data:www-data /var/www/html/photoshow
mkdir ~/photos
mkdir ~/generated
```



2. Zugriffsrechte für die Benutzer *pi* und *www-data* einrichten:

```
sudo chown -R www-data:www-data ~/photos
sudo chown -R www-data:www-data ~/generated
sudo chmod -R 775 ~/photos
sudo chmod -R 775 ~/generated
sudo adduser pi www-data
```

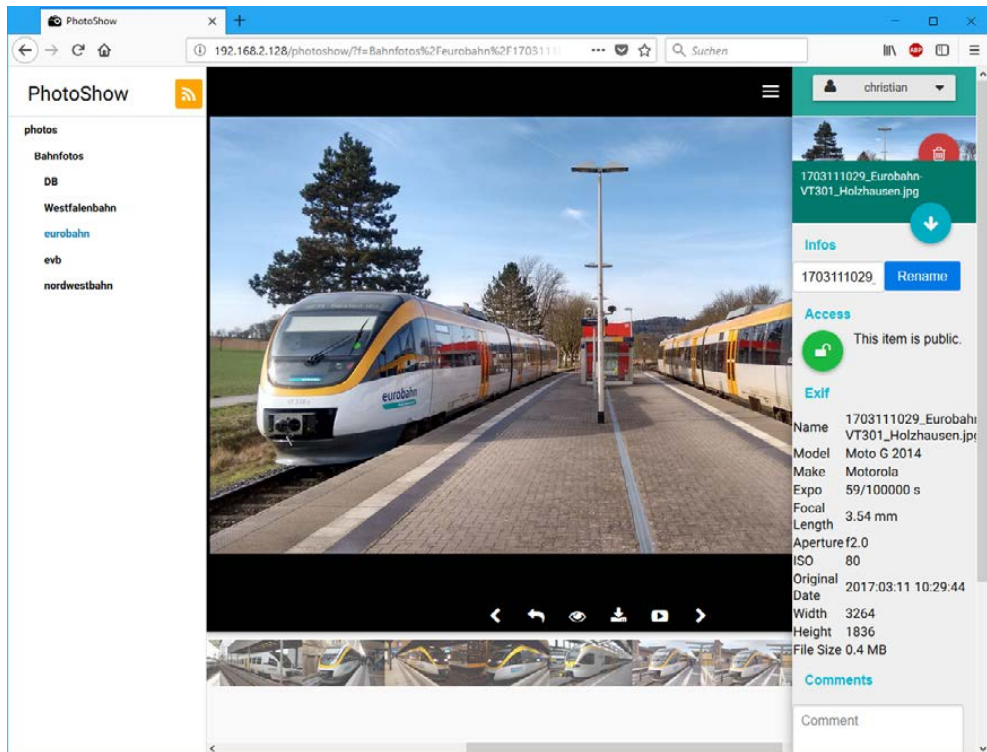
3. Datenverzeichnisse in die Konfiguration von PhotoShow eintragen:

```
sudo nano /var/www/html/photoshow/config.php
```

Hier die vorgegebenen Zeilen für `photos_dir` und `ps_generated` ändern:

```
$config->photos_dir = "/home/pi/photos";  
$config->ps_generated = "/home/pi/generated";
```

4. PhotoShow im Browser auf dem PC aufrufen. Dabei die IP-Adresse durch die des Raspberry Pi ersetzen: <http://192.168.2.124/photoshow>
5. Beim ersten Start erscheint ein Formular, in dem das Hauptbenutzerkonto angelegt werden muss. PhotoShow verwendet eine eigene, von Linux unabhängige Benutzerverwaltung.



6. Angemeldete Benutzer haben die Möglichkeit, Alben und Unterordner für Fotos anzulegen sowie Bilder hochzuladen. Bei jedem Album kann man festlegen, ob es privat oder öffentlich sein soll.
7. Bilder müssen nicht unbedingt einzeln über das Uploadformular hochgeladen werden, sondern können auch direkt in Unterverzeichnisse des `photos`-Verzeichnisses kopiert werden. Allerdings gehören neu angelegte Ordner dem Benutzer `pi` und lassen sich in der Webansicht von PhotoShow nicht interaktiv bearbeiten, sondern nur betrachten. Um das Problem zu lösen, nach dem manuellen Anlegen neuer Ordner im Dateisystem die Rechte neu zuweisen:

```
sudo chown -R www-data:www-data ~/photos
sudo chmod -R 0775 ~/photos
```

8. Ein anonymen Besucher der Fotogalerie kann sich oben rechts über den Button **Anmelden** ein eigenes Benutzerkonto anlegen. Das allein gibt ihm noch keinerlei Rechte, außer dass er Kommentare jetzt mit seinem Namen schreibt und nicht mehr anonym.
9. Der Administrator kann diesem neuen Nutzer jetzt verschiedene Rechte geben. Dazu sind unter **Edit Groups** im **Admin**-Menü drei Benutzergruppen vordefiniert. Benutzer einfach per Drag-and-drop in eine Gruppe ziehen oder mit dem Häkchen links neben dem Benutzernamen wieder aus einer Gruppe herausnehmen.
 - **root** – Nutzer in dieser Gruppe haben Administrator-Berechtigung und dürfen alles: private Alben betrachten, Alben anlegen, Fotos hochladen, Einstellungen bearbeiten, Benutzer verwalten.
 - **uploaders** – Nutzer in dieser Gruppe dürfen Fotos hochladen, aber nichts an den Einstellungen ändern.
 - **user** – Nutzer in dieser Gruppe dürfen nur Fotos betrachten.

166

Samba-Server installieren

Ein Samba-Server stellt das von Microsoft für Windows-Freigaben und Arbeitsgruppen entwickelte SMB-Protokoll (**S**erver-**M**essage-**B**lock-Protokoll) auf einem Linux-PC oder Raspberry Pi zur Verfügung. Damit lassen sich ein Raspberry Pi 4 und 400 in ein Windows-Netzwerk integrieren.

1. Pakete installieren:


```
sudo apt update
sudo apt install samba samba-common-bin
```
2. Bei der Installation wird der Samba-Dienst automatisch in die Datei **/etc/init.d** eingetragen, damit er in Zukunft beim Start des Raspberry Pi direkt mit gestartet wird.
3. Samba-Benutzer anlegen. Dabei muss ein Passwort festgelegt werden, das später zur Anmeldung auf dem Windows-PCs benötigt wird.


```
sudo smbpasswd -a pi
```
4. Verzeichnisse für die gemeinsam genutzten Daten freigeben:


```
sudo nano /etc/samba/smb.conf
```
5. Den folgenden Textblock am Ende hinzufügen:


```
[RaspberryPi]
path = /home/pi
writable = yes
```



- Der Titel in eckigen Klammern, hier `[RaspberryPi]`, gibt den Namen der Freigabe im Windows-Netzwerk an.
 - Der Eintrag `path`, hier `/home/pi`, gibt das freigegebene Verzeichnis auf dem Raspberry Pi an.
 - Der Parameter `writable` legt fest, dass Benutzer aus dem Netzwerk Daten auf dem Raspberry Pi schreiben dürfen.
6. Wird im Windows-Netzwerk eine andere Arbeitsgruppe als die Standard-**WORKGROUP** verwendet, diese in der Zeile `workgroup` im Bereich `[global]` eintragen.
7. Samba-Server neu starten: `sudo /etc/init.d/samba restart`

Der Name der Arbeitsgruppe findet sich auf einem Windows-PC unter **System/Info**: Tastenkombination **[Win]+[Pause]** drücken, im nächsten Fenster in der Mitte auf **Domäne oder Arbeitsgruppe** klicken.

