



Handbuch Manual

Android Software

Version 1.0
Artikel-Nr. / Part-No. 17873



verbindet Menschen und Technik

Inhaltsverzeichnis

1. Haftung / Copyright	3
2. Spezifische Geräteeigenschaften	
2.1 Rotation	4
3. EntwicklerOptionen	
3.1 Entwicklungsumgebung	5
3.2 ADB Treiber	5
3.3 Beschreibung der API für Temperatursensor, GPIO und LED Steuerung	5
3.3.1 Temperatursensor	5
3.3.2 LED und Sensoren	6
3.3.3 GPIO	7
4. Launcher Einstellungen	8
4.1 Einstellungen	10
4.2 Einstellungen Design	11
4.3 Vollständige App-Übersicht	12
4.4 Widgets	12
4.5 Sperren	12
4.6 Helligkeit	12
5. PanelControl	
5.1 Brightness control	13
5.2 Adaptive Brightness	14
6. App Installation	
6.1 Standardinstallation	15
6.2 App Installation ohne Google Playstore	15
6.3 App Installation mit ADB (Android Debug Bridge)	15
7. App updater 2.0	16
8. App-Watchdog	22
9. Gira URL-Endpoints (nur VI.r3)	23
10. Fehlerbehebung	25
Englisch	26



1. Haftung / Copyright

Copyright	<p>tci GmbH, Ludwig-Rinn-Str. 10-14, 35452 Heuchelheim, Deutschland</p> <p>Dieses Handbuch, sowie die Hard- und Software, die es beschreibt, ist urheberrechtlich geschützt und darf ohne ausdrückliche schriftliche Genehmigung der tci GmbH in keiner Weise vervielfältigt, übersetzt oder in eine andere Darstellungsform gebracht werden.</p>
Hinweis	<p>Herausgeber, Übersetzer und Autoren dieser Publikation haben mit größter Sorgfalt die Texte, Abbildungen und Programme erarbeitet. Dennoch können Fehler nicht völlig ausgeschlossen werden. Die tci GmbH übernimmt daher weder eine Garantie noch eine juristische Verantwortung oder Haftung für Folgen, die auf fehlerhafte Angaben zurückgehen.</p> <p>Mitteilungen über eventuelle Fehler werden jederzeit gerne entgegengenommen. Die Angaben in diesem Handbuch gelten nicht als Zusicherung bestimmter Produkteigenschaften. Änderungen, die dem technischen Fortschritt dienen, bleiben vorbehalten.</p>
Haftung	<p>Die tci GmbH haftet nicht für unmittelbare Schäden, die im Zusammenhang mit der Lieferung oder dem Gebrauch der Dokumentation stehen. Wir haften zudem auch nicht für etwaige Fehler in dieser Publikation. Wir verpflichten uns in keiner Weise, die in dieser Dokumentation enthaltenen Informationen auf den aktuellsten Stand zu bringen oder auf dem neuesten Stand zu halten.</p> <p>Alle Rechte vorbehalten</p>

Ltztes Update Mai 2023

2. Spezifische Geräteeigenschaften

2.1 Rotation

Unter Umständen verfügt das System keinen Lagesensor. Somit erfolgt auch keine automatische Rotation. Die Rotation kann über die Android Settings konfiguriert werden. Anwendungen die die aktuelle Rotation ändern sind unter Umständen nicht für das Gerät geeignet.



3. Entwickleroptionen

3.1 Entwicklungsumgebung

Zur Entwicklung von eigenen Anwendungen können alle für die Android Entwicklung geeigneten Umgebungen verwendet werden. Es sind keine speziellen Voraussetzungen nötig.

3.2 ADB Treiber

Ein spezieller für das Gerät geeigneter Treiber kann beim Hersteller angefordert werden. Dieser ist voll mit den ADB Werkzeugen und Anwendungen kompatibel.

3.3 Beschreibung der API für Temperatursensor, GPIO und LED Steuerung

Allgemein: Die Ansteuerung des Temperatursensors, der GPIO's und der LED's erfolgt über das Linux basierte Dateisystem von Android. Alle I/O Operationen können durch einfache lese/schreib Vorgänge in die jeweils zugehörige Datei vorgenommen werden.

Die folgenden Optionen sind Ausstattungsabhängig.

3.3.1 Temperatursensor

Der verbaute Thermistor kann wie oben beschrieben über das Dateisystem ausgelesen werden. Das Ergebnis der Leseoperation ist ein gerasterter Spannungswert zwischen 0-1024. Der so ermittelte Spannungswert muss anschließend in den aktuellen Widerstand des Thermistors umgerechnet werden. Der so ermittelte aktuelle Widerstand des Thermistors kann dann mit der vereinfachten Steinhart-Hart-Gleichung (B or β parameter equation) zur Berechnung der Temperatur verwendet werden.

Beispiel:

Umrechnung des Spannungswerts in den aktuellen Widerstand des Thermistors

Lesen des gerasterten Spannungswerts (rawValue) aus der Datei: (Siehe 3.3.2 ff)

VOLTAGE_INPUT = 1.8

VOLTAGE_MULTIPLIER = VOLTAGE_INPUT / 1024

R1 = 10000

voltageOut = rawValue * VOLTAGE_MULTIPLIER

$R = (\text{voltageOut} * R1) / (\text{VOLTAGE_INPUT} - \text{voltageOut})$

Der Widerstand R kann nun in die Steinhart-Hart-Gleichung eingesetzt werden um die aktuelle Temperatur in Grad Celsius zu berechnen.

Steinhart-Hart-Gleichung (B or β parameter equation)

Die Temperatur in Kelvin, R_0 ist der Widerstand bei Temperatur T_0 ($25^\circ\text{C} = 298.15\text{K}$)

$R_0 = 10 \text{ k}\Omega = 10000$

$B = 3435$

$T_0 = 298.15$

R = current resistance of the thermistor

T = temperature in degrees Kelvin

$$r_{\infty} = R_0 e^{-B/T_0}$$

$$T = \frac{B}{\ln(R/r_{\infty})}$$

Umrechnung Kelvin in Grad Celsius

GradCelsius = T - 273,15

3.3.2 LEDs und Sensoren

Es stehen eine rote und eine grüne LED zur Verfügung. Beide LEDs können unabhängig voneinander gesetzt werden. Der Zugriff erfolgt wieder über das Dateisystem:

Rk3288-VI (Modellabhängig, siehe Typenschild z.B. luna-X-VI.r4)

LED Rot: ksp0630:red

LED Grün: ksp0630:green

LED einschalten: echo 1 > /sys/class/leds/ksp0630:green/brightness

LED ausschalten: echo 0 > /sys/class/leds/ksp0630:green/brightness

Helligkeitssensor: /sys/bus/iio/devices/iio:device0/in_voltage0_raw

Temperatursensor: /sys/bus/iio/devices/iio:device0/in_voltage1_raw

Rk3399-VI (Modellabhängig, siehe Typenschild z.B. luna-x-VI.r5)

LED Grün ein = echo 255 > /sys/class/leds/work/brightness

LED Grün aus = echo 0 > /sys/class/leds/work/brightness

LED Rot ein = echo 255 > /sys/class/leds/standby/brightness

LED Rot aus = echo 0 > /sys/class/leds/standby/brightness

Helligkeitssensor: /sys/bus/iio/devices/iio:device0/in_voltage0_raw

Temperatursensor: /sys/bus/iio/devices/iio:device0/in_voltage2_raw

Rk3288-XI (Modellabhängig siehe Typenschild z.B. F16-XI)

LED Grün ein = echo 1 > /sys/class/leds/egpio_o1/brightness

LED Grün aus = echo 0 > /sys/class/leds/egpio_o1/brightness

LED Rot ein = echo 1 > /sys/class/leds/egpio_o2/brightness

LED Rot aus = echo 0 > /sys/class/leds/egpio_o2/brightness

Rk3188-XI (Modellabhängig siehe Typenschild z.B. E07A-XI)

LED Grün ein = echo 1 > /sys/class/gpio/gpio252/value

LED Grün aus = echo 0 > /sys/class/gpio/gpio252/value

LED Rot ein = echo 1 > /sys/class/gpio/gpio253/value

LED Rot aus = echo 0 > /sys/class/gpio/gpio253/value



Rk3288-VI-solo (Modellabhängig, siehe Typenschild z.B. solo oder E04)

Helligkeitssensor: `/sys/devices/virtual/optical_sensor/lightsensor/ls_level`

Temperatursensor intern: `/sys/bus/iio/devices/iio:device0/in_voltage0_raw`

Temperatursensor extern (optional): `/sys/bus/iio/devices/iio:device0/in_voltage2_raw`

Proximity: `/sys/devices/virtual/optical_sensor/proximity/ps_level`

3.3.3 GPIO

Rk3288-VI (Modellabhängig, siehe Typenschild z.B. luna-X-VI.r4)

Etagenruf = `cat /sys/class/gpio/gpio251/value` (default value = 1)

Relays on= `echo 1 > /sys/class/gpio/gpio48/value`

Relays off= `echo 0 > /sys/class/gpio/gpio48/value`

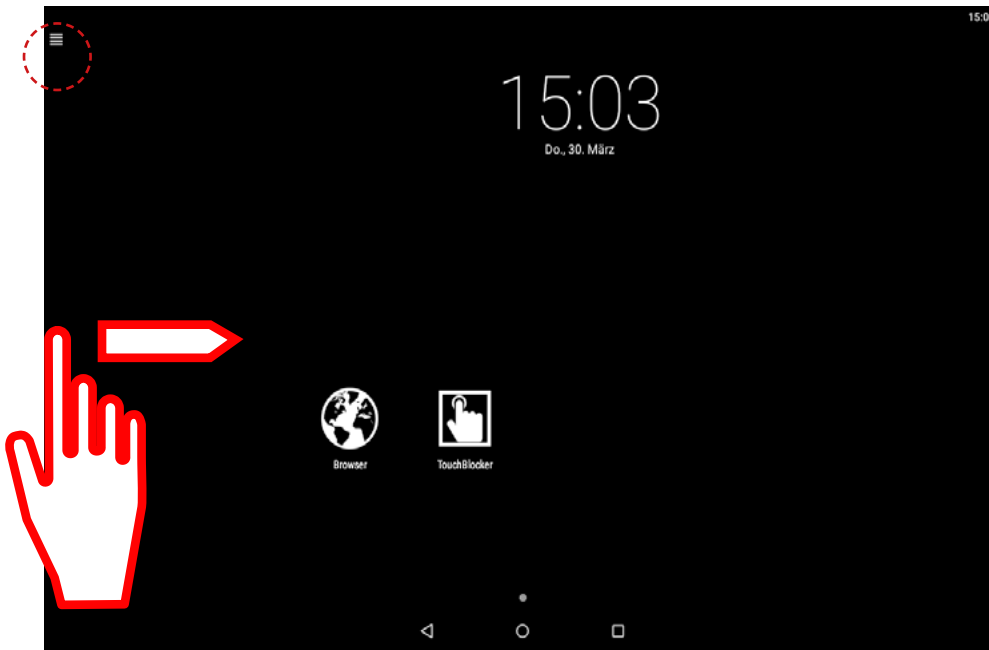
Rk3399-VI (Modellabhängig, siehe Typenschild z.B. luna-x-VI.r5)

Etagenruf = `cat /sys/class/gpio/gpio102/value` (default value = 1)

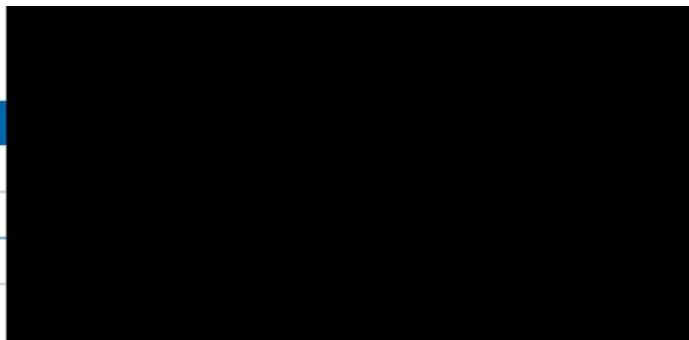
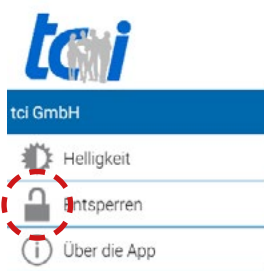
Relays on= `echo 1 > /sys/class/leds/gpio1/brightness`

Relays off= `echo 0 > /sys/class/leds/gpio1/brightness`

4. Launcher Einstellungen

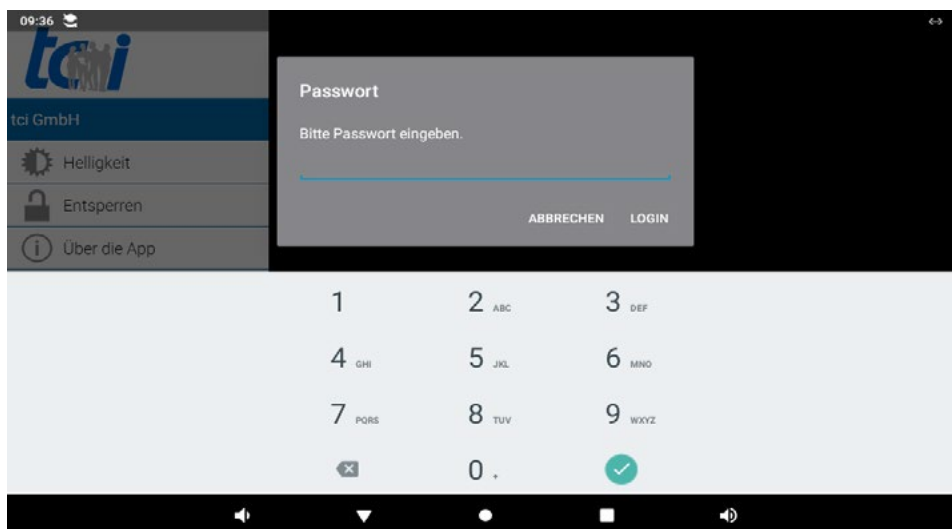


Öffnen sie die Launcher-übersicht mit einer Wischbewegung vom linken Rand zur Mitte des Bildschirms, oder durch betätigen des Menü-Icons in der oberen linken Ecke.



Unlock:

Öffnet den Passwortgeschützten Bereich für den Zugriff auf Apps und Einstellungen.



Es gibt zwei voreingestellte Benutzer:

Benutzer: Endanwender
Passwort: 0000

Benutzer: Integrator
Passwort: 8273

Der Benutzer Endanwender hat gegenüber dem Benutzer Integrator nur eingeschränkte Einstellmöglichkeiten.



Beschränkungen Endanwender

- Nur Zugriff auf Einstellungen für Audio und Display
- Vorinstallierte Apps wie Dev Tools und auch System Apps wie Settings werden aus der „Alle Apps“-Übersicht ausgeblendet und können auch nicht anderweitig aufgerufen werden
- Applikationen können nicht de-/installiert werden

Login als Integrator

Nach dem Login erscheint die Launcherübersicht mit neuen Optionen. Haben Sie sich als Integrator eingeloggt stehen Ihnen folgende Optionen zur Verfügung:



tci GmbH



Helligkeit



Einstellungen



Alle Apps



Widgets

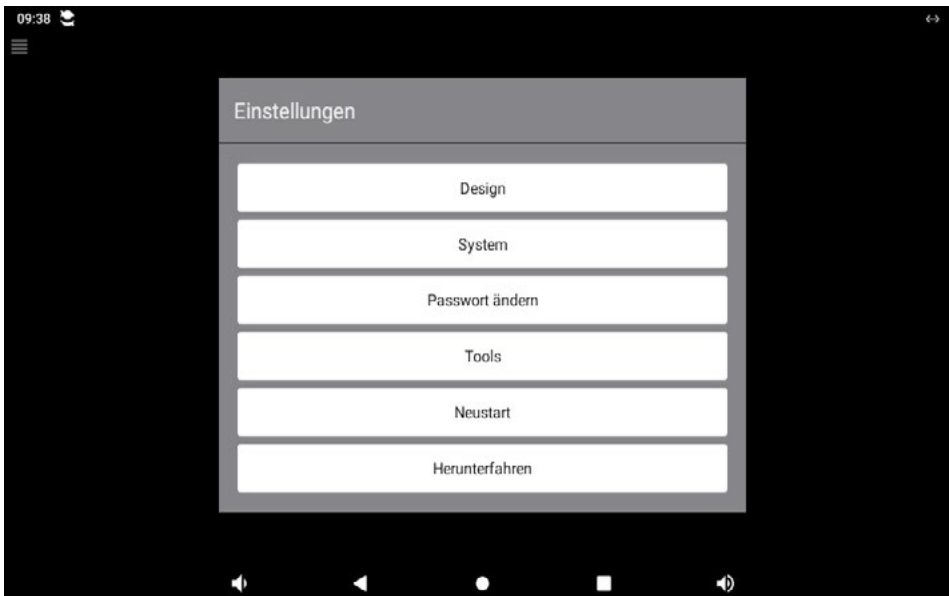


Sperrern



Über die App

4.1 Einstellungen



Design

Änderungen die das Hintergrundbild und sonstiges Erscheinen des Launchers beeinflussen.

System

Führt Sie zu den Standard Android Einstellungen wie Display, Ton & Benachrichtigungen, Ethernet, Speicher, Apps, Sprache, Konten usw. Die Standard Android Optionen werden in diesem Handbuch nicht weiter behandelt.

Passwort ändern

Ändern des Passwortes des derzeit eingeloggten Benutzers.

Tools

- Backup: Erstellt ein Sicherung der derzeitigen Launchereinstellungen.
- Restore: Bei Fehlerhaften Einstellungen kann über die Restore Funktion der Launcher von einem Backup wiederhergestellt werden (Widgets und Hintergrundbild ausgenommen). Auch geeignet zur Inbetriebnahme von mehreren Systemen.
- Launcher Einstellungen zurücksetzen: Entfernt alle Widgets und Apps vom Startbildschirm.

Neustart

Startet das System neu.

Herunterfahren

Schaltete das System aus.



4.2 Designeinstellungen



Logo

Ändert das Logo welches im Launcher in der oberen linken Ecke angezeigt wird.

tci GmbH (Name)

Ändert den angezeigten Namen unterhalb des Logos (1).

Background Color (Titel)

Ändert die Hintergrundfarbe des Textes (2) im Launcher.

Text Farbe (Titel)

Ändert die Textfarbe des frei wählbaren Textes.

Hintergrundbildr

Ändert den Hintergrund in eine einfarbige Fläche. Diese kann durch setzen des Hakens aktiviert oder deaktiviert werden. Bei Verwendung eines Bildes als Wallpapers muss diese Option deaktiviert sein. Ein Wallpaper kann über die Standard Android Einstellungen eingerichtet werden.

Speichern

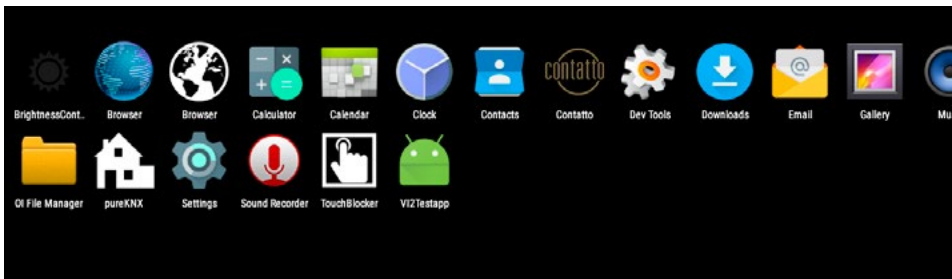
Speichert die eingestellten Texte, Logos und Einstellungen.

Abbrechen

Verwirft die getätigten Einstellungen.

4.3 Vollständige App-Übersicht

Eine Übersicht über alle installierten Applikationen auf dem System. Mit Berühren und halten(Drag & Drop) der gewünschten App kann diese auf die Startseite gezogen werden. Eine App kann durch berühren, halten und auf den Mülleimer ziehen wieder von der Startseite entfernt werden. Zum Verschieben von Apps und Widgets müssen Sie mit einem Passwort eingeloggt sein.



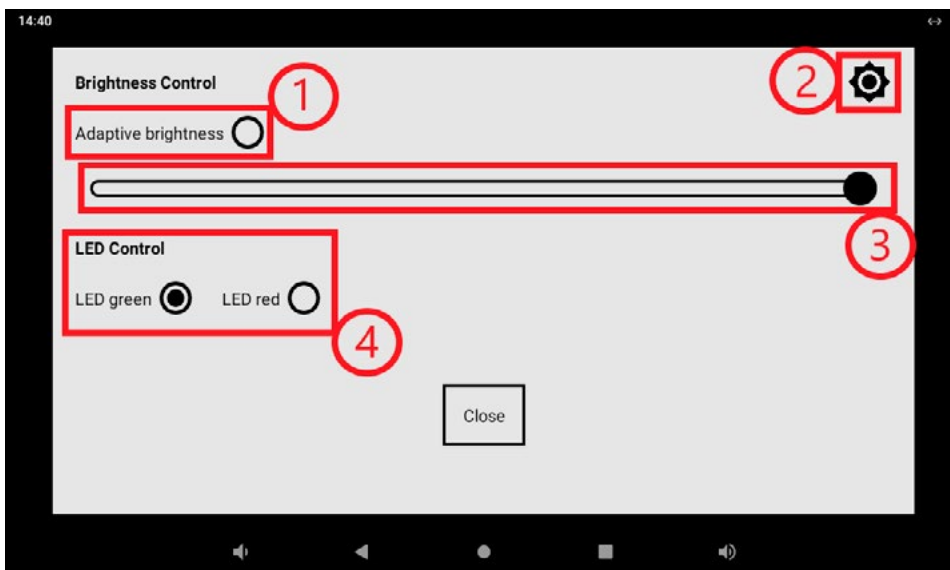
4.4 Widgets

Eine Übersicht über kleine Applikationen für den Startbildschirm wie z.B. Uhrzeit, Kalender und andere nützliche kleine Gadgets. Können durch berühren und halten auf den Startbildschirm gelegt werden. Entfernen der Widgets verhält sich analog zum Entfernen der Apps.

4.5 Sperren

Sperrt das System gegen unbefugten Zugriff auf die Systemeinstellungen. Wenn Einstellungen mit dem Integrator Benutzer getätigt wurden, werden dazu alle laufenden Applikationen beendet.

5. PanelControl



Die folgenden Optionen sind Ausstattungsabhängig

5.1 Brightness control

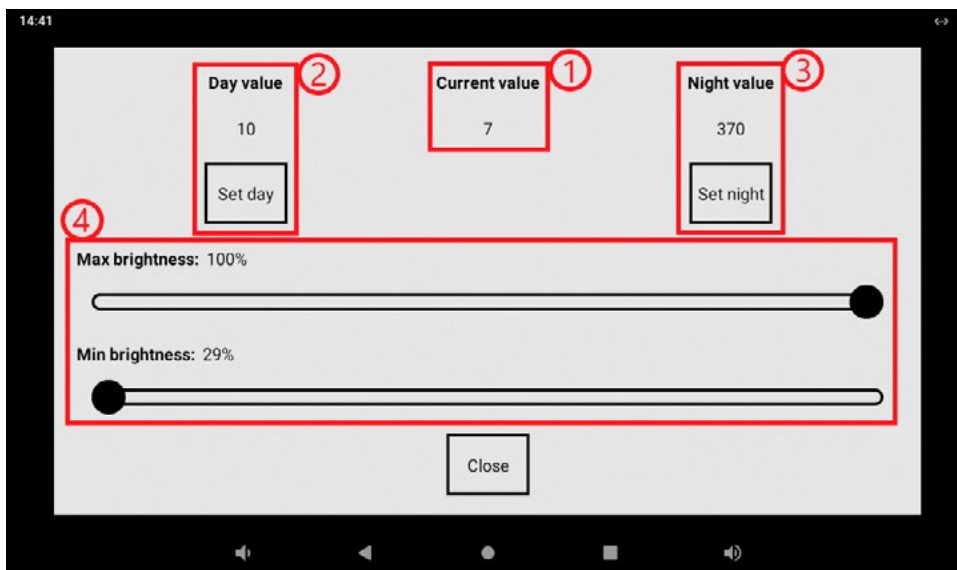
PanelControl ist ein Tool zur Steuerung der Bildschirmhelligkeit.

1. „Adaptive brightness“ (Automatische Helligkeitsreglung). Bei erstmaliger Aktivierung, wird die Bildschirmhelligkeit automatisch an die Umgebungshelligkeit angepasst. Es erfolgt ein Hinweis, dass Sie den Sensor kalibrieren müssen.
2. Mit diesem Symbol öffnen Sie das Menü zur Kalibrierung und Einstellung des Sensors brightness Modus wird die Helligkeit automatisch geregelt.
3. Mit dem Schieberegler stellen Sie die Helligkeit manuell und dauerhaft ein. Die Automatische Helligkeitsregelung ist damit deaktiviert.
4. Mit LED Control können Sie der Status LED am oberhalb des Touches einstellen

5.2 Adaptive brightness

Bitte beachten Sie, dass die Einstellungen im eingebauten Zustand vorgenommen werden sollten.

1. Current value zeigt den aktuellen, sich permanent ändernden Sensor Wert.
2. Day value; hier legen Sie den Tag-Wert fest [Set day], ab diesem Wert stellt sich das Display auf die maximale Helligkeit ein
3. Night value; hier legen Sie den Nacht-Wert fest [Set night], unterschreitet der Sensor Wert diese Angabe, stellt sich das Display auf den niedrigsten Helligkeitsgrad ein. Optimalerweise stellen Sie diese Eingaben auch bei Nacht ein.
4. Über die Schieberegler Max brightness und Min brightness begrenzen Sie die Bildschirmhelligkeit insgesamt ein. Die Bildschirmhelligkeit verändert sich linear zu den herrschenden Lichtverhältnissen.





6. App Installation

6.1 Standardinstallation

Um die Installation einer APK (APK-Datei) vorzunehmen, melden Sie sich als Integrator an.

Kopieren Sie die gewünschte App auf einen USB-Stick. Navigieren Sie in den Einstellungen des Launcher zu „Alle Apps“. Öffnen Sie anschließend die App „Explorer“. Den Inhalt Ihres USB-Sticks finden Sie unter „USB“. Mit einem Klick auf die gewünschte APK-Datei können Sie die Installation starten. Sollte eine APK bereits installiert sein, wird diese aktualisiert.

6.2 App Installation ohne Google Playstore

So gelangen Sie zu den benötigten APK-Dateien (APK = Android Package, Format installierbarer Software auf Android Systemen) auch ohne einen direkten Playstore-Zugang:

1. Bezug der APK Datei direkt vom Hersteller

Viele Hersteller in der Gebäudeautomation verfügen über eigene Apps zur Steuerung von z.B. IP basierten Sprechanlagen. Oft ist der einfachste Weg die benötigten Dateien direkt beim Hersteller anzufordern.

2. Bezug aus Playstore unter Zuhilfenahme eines Android Smartphones oder Tablets

Wenn Sie über ein weiteres Android Gerät verfügen, wie z.B. ein Smartphone oder Tablet mit Playstore-Zugang, können Sie die benötigten APK-Dateien über einen weiteren Weg erlangen. Im Playstore sind diverse Apps erhältlich, die es ermöglichen auf dem System installierte Apps zu extrahieren, um so die benötigte APK Datei zu erhalten (z.B. die App „SaveAPK“).

- Installieren Sie auf Ihrem Smartphone / Tablet die gewünschte App die Sie auf dem tci-System installieren möchten.
- Installieren Sie zusätzlich eine App die einen Export von APK-Dateien erlaubt (wie z.B. „SaveAPK“)
- Starten Sie die App „saveAPK“ und wählen Sie die zu exportierende App und den Speicherpfad
- Kopieren Sie nun die exportierte APK-Datei von Ihrem Smartphone / Tablet auf einen USB-Stick und schließen Sie diesen an Ihr tci-System an.
- Starten Sie den Dateimanager auf Ihrem tci-System und navigieren Sie zu Ihrem USB-Stick ([/storage/usb_host](#)). Klicken Sie auf die gewünschte APK Datei um diese zu installieren.

3. Bezug aus Playstore ohne Smartphone oder Tablet

Wenn Sie über kein weiteres Android Gerät verfügen, besteht noch die Möglichkeit die benötigten APK Dateien über diverse Webseiten zu beziehen, die einen Export aus dem Playstore auf einen PC erlauben. Öffnen Sie den Playstore auf Ihrem PC mit Ihrem Browser und navigieren Sie zu der gewünschten App. Öffnen Sie in einem zweiten Tab z.B. die Webseite <http://apps.evozi.com/apk-downloader/>

Kopieren Sie die URL aus dem Playstore in die Eingabefläche und klicken Sie auf „Generate Download Link“. Sie können die APK Datei nun auf Ihren PC herunterladen. Installation der App erfolgt dann wieder über einen USB-Stick auf dem Zielsystem, wie in Punkt 2 beschrieben.

6.3 App Installation mit ADB (Android Debug Bridge)

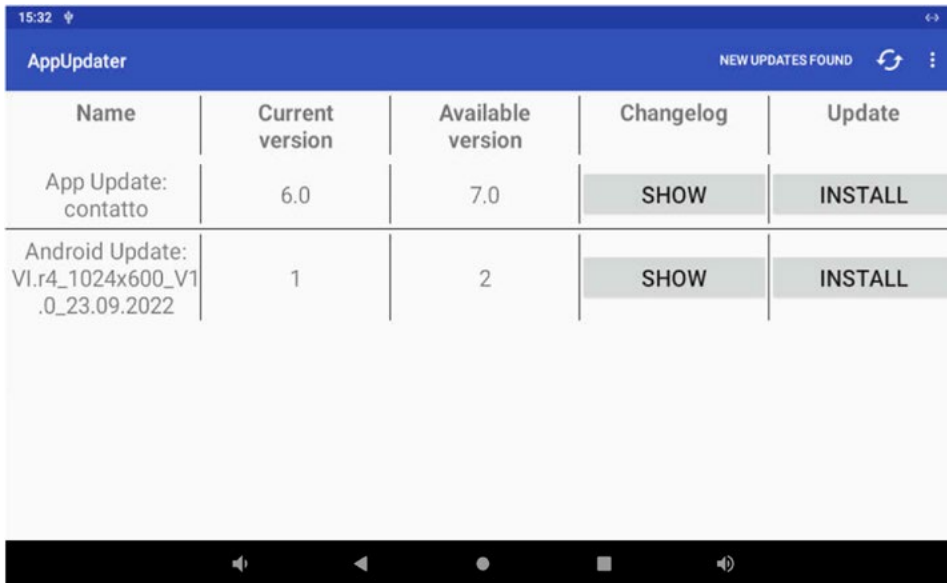
Bei Systemen ohne USB-Anschluss können Sie die Android Debug Bridge zur App-Installation nutzen. Installieren Sie hierfür entweder das Android-SDK (enthält ADB-Tool) oder z.B. das leichtgewichtige „Minimal ADB and Fastboot“-Tool auf Ihrem PC. Öffnen Sie anschließend die ADB-Konsole und verwenden Sie folgenden Befehl: `adb install path_to_apk`

Wie Sie eine Verbindung per ADB-Ethernet zu Ihrem System aufbauen ist im Kapitel 4 beschrieben.

7. App Updater 2.0

Der tci AppUpdater kann dazu verwendet werden, ihr Android System auf einem aktuellen Stand zu halten. Dies umfasst sowohl App- als auch Android- Updates. Der tci AppUpdater arbeitet mit HTTP Links.

Manuell nach Updates suchen



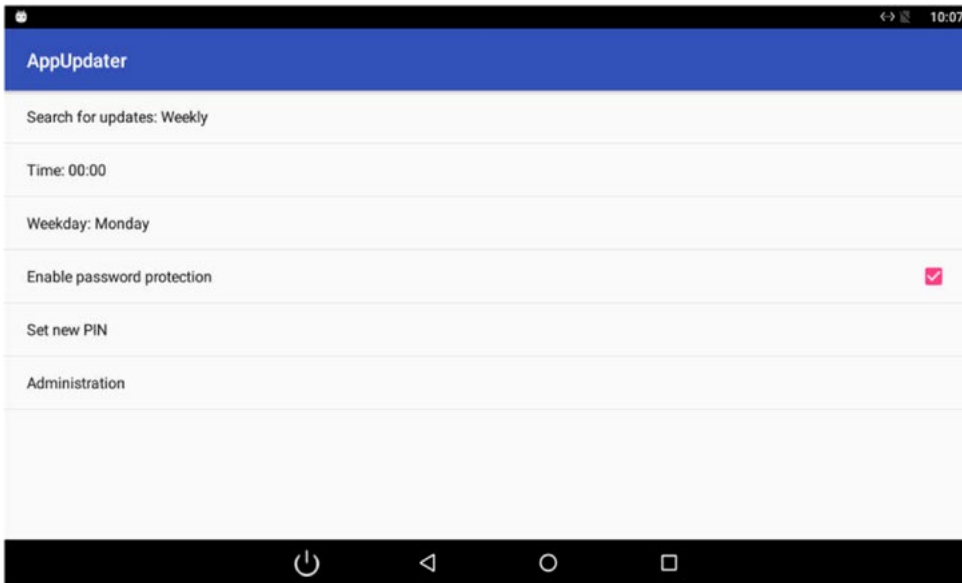
The screenshot shows the tci AppUpdater application interface. At the top, there is a blue header with the time 15:32, a signal strength indicator, and a refresh icon. Below the header, the text 'AppUpdater' is on the left and 'NEW UPDATES FOUND' with a refresh icon and a menu icon is on the right. The main content is a table with the following data:

Name	Current version	Available version	Changelog	Update
App Update: contatto	6.0	7.0	SHOW	INSTALL
Android Update: V1.r4_1024x600_V1.0_23.09.2022	1	2	SHOW	INSTALL

1. Starten Sie den AppUpdater
2. Klicken Sie in der Actionbar das Icon zum Suchen nach Updates
3. Sind Updates verfügbar werden diese tabellarisch aufgelistet
4. Mit einem Click auf „Update“ wird die entsprechende App geladen und installiert

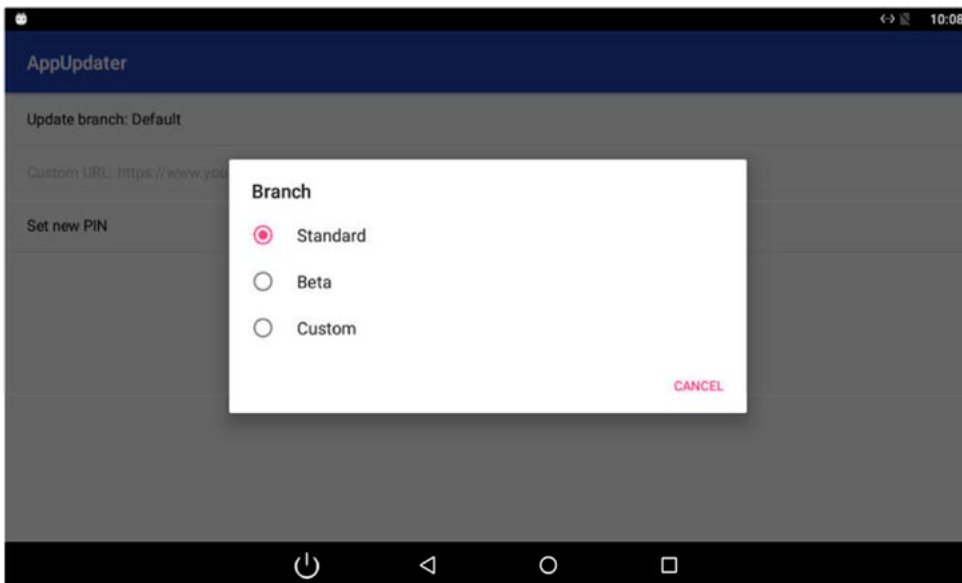


Einstellungen und automatische Suche nach Updates



1. Search for updates: Erlaubt das zeitbasierte suchen nach Updates
 - a. Never: Nie nach Updates suchen
 - b. Daily: Sucht täglich zur angegebenen Uhrzeit nach Updates
 - c. Weekly: Sucht wöchentlich am festgelegten Tag und zur angegebenen Uhrzeit nach Updates
2. Enable password protection: Schaltet den Passwortschutz für das Einstellungsmenü ein/aus
3. PIN: Erlaubt das Festlegen des Pins (default: 0000)
4. Administration: Öffnet die erweiterte Konfiguration zur Auswahl des Aktualisierungszweigs

Auswahl des Aktualisierungszweigs



1. Erlaubt die Auswahl des Aktualisierungszweigs (Standard, Beta und Custom)
2. Wenn Sie den „Custom“-Zweig wählen, müssen Sie den Pfad zur Konfigurationsdatei auf Ihrem Webserver selbst konfigurieren. Folgen Sie dem nächsten Abschnitt „Konfiguration“
3. Erlaubt das setzen eines neuen Pins (default:1234)

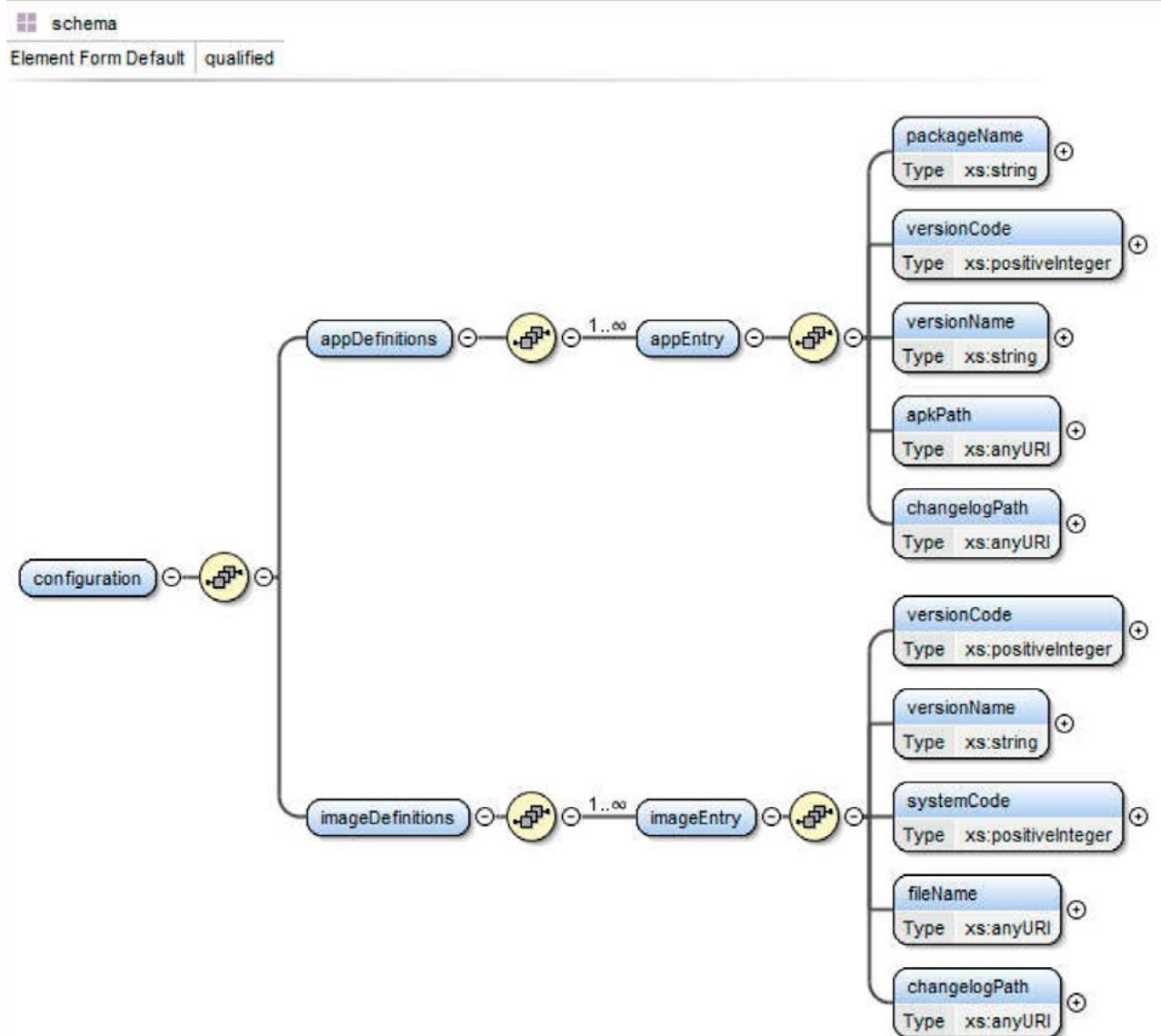
Konfiguration

Erstellen der Konfigurationsdatei „UpdateConfig.xml“.

In der Datei „UpdateConfig.xml“ werden die zur Verfügung stehenden Updates konfiguriert. Anschließend muss die Datei „UpdateConfig.xml“ als HTTP-Link online verfügbar gemacht werden.

Der Link zur „UpdateConfig.xml“ muss in den App Settings unter „URL to config file“ angegeben werden.

XSD Definition





```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <xs:element name="configuration">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="appDefinitions">
          <xs:complexType>
            <xs:sequence>
              <xs:element maxOccurs="unbounded" name="appEntry">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element name="packageName" nillable="false" type="xs:string"/>
                    <xs:element minOccurs="1" name="versionCode"
                      type="xs:positiveInteger"/>
                    <xs:element minOccurs="1" name="versionName" type="xs:string"/>
                    <xs:element maxOccurs="1" name="apkPath" type="xs:anyURI"/>
                    <xs:element minOccurs="1" name="changelogPath" type="xs:anyURI"/>
                  </xs:sequence>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="imageDefinitions">
          <xs:complexType>
            <xs:sequence>
              <xs:element maxOccurs="unbounded" name="imageEntry">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element name="versionCode" type="xs:positiveInteger"/>
                    <xs:element name="versionName" type="xs:string"/>
                    <xs:element name="systemCode" type="xs:positiveInteger"/>
                    <xs:element name="fileName" type="xs:anyURI"/>
                    <xs:element name="changelogPath" type="xs:anyURI"/>
                  </xs:sequence>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Tag Definition

Configuration:	Anfang und Ende der Konfigurationsdefinition
appDefinitions:	Einleitendes Tag für alle App-Updates
appEntry:	Einleitendes Tag für eine App-Update Definition
packageName:	Der Package Name der Anwendung, z.B. de.tci.contatto
versionCode:	Ein ganzzahliger fortlaufender Wert zur Versionsprüfung, wird verwendet um zu entscheiden welche Version neuer ist. Z.B. „123“
versionName:	Ein Alias für den versionCode, z.B. „7.1“
apkPath:	Der http Link zum Download der APK-Datei
changelogPath:	Der http Link zu einer Change log Textdatei
imageDefinitions:	Einleitendes Tag für alle Android Image-Updates
imageEntry:	Einleitendes Tag für eine Android Image Update Definition
versionCode:	Ein ganzzahliger fortlaufender Wert zur Versionsprüfung, wird verwendet um zu entscheiden welche Version neuer ist. Z.B. „123“
versionName:	Ein Alias für den versionCode, z.B. „7.1“
systemCode:	eindeutige Image ID
fileName:	Der http-Link zum Download der Image Datei
changelogPath:	Der http Link zu einer Change log Textdatei



Beispiel für eine vollständige UpdateConfig.xml Datei

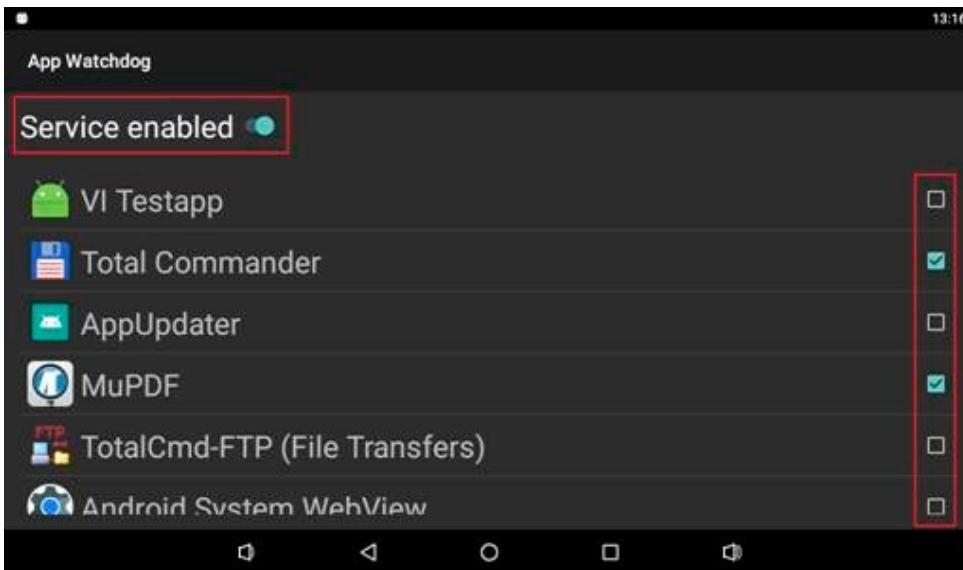
```
<?xml version="1.0" encoding="UTF-8"?>
<configuration xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="file://D:/ConfigScheme.xsd">
  <appDefinitions>
    <appEntry>
      <packageName>de.tci.contatto</packageName>
      <versionCode>8</versionCode>
      <versionName>7.0</versionName>
      <apkPath>http://tci.de/contattoVI_7_0.apk</apkPath>
      <changelogPath>http://tci.de/contatto_changelog.txt</changelogPath>
    </appEntry>
    <appEntry>
      <packageName>net.spidercontrol.knxserver</packageName>
      <versionCode>14</versionCode>
      <versionName>7.4</versionName>
      <apkPath>http://tci.de/knxServer.apk</apkPath>
      <changelogPath>http://tci.de/knxServer_changelog.txt</changelogPath>
    </appEntry>
  </appDefinitions>
  <imageDefinitions>
    <imageEntry>
      <versionCode>3</versionCode>
      <versionName>2.1</versionName>
      <systemCode>1337</systemCode>
      <fileName>http://tci.de/update_1337.zip</fileName>
      <changelogPath>http://tci.de/update_1337_changelog.txt</changelogPath>
    </imageEntry>
    <imageEntry>
      <versionCode>2</versionCode>
      <versionName>2.0</versionName>
      <systemCode>4711</systemCode>
      <fileName>http://tci.de/update_4711.zip</fileName>
      <changelogPath>http://tci.de/update_4711_changelog.txt</changelogPath>
    </imageEntry>
  </imageDefinitions>
</configuration>
```

8. AppWatchdog

Der AppWatchdog befindet sich im „Tools“- Fenster des Launchers (mit 8273 einloggen).



Wählen Sie im „Tools“-Menü den AppWatchdog aus.



Im AppWatchdog finden Sie eine Liste aller installierten Apps. Wählen Sie die gewünschten Apps aus, indem Sie die zugehörige Checkbox aktivieren.

Aktivieren Sie anschließend den AppWatchdog selbst (Erster Listeneintrag: Service enabled).

Der AppWatchdog überwacht nun für alle ausgewählten Apps, ob die App, bzw. ihr Prozess noch läuft. Ist dies nicht der Fall, wird die betreffende App neu gestartet. Zusätzlich werden alle ausgewählten Apps um 2 Uhr nachts beendet und neu gestartet.

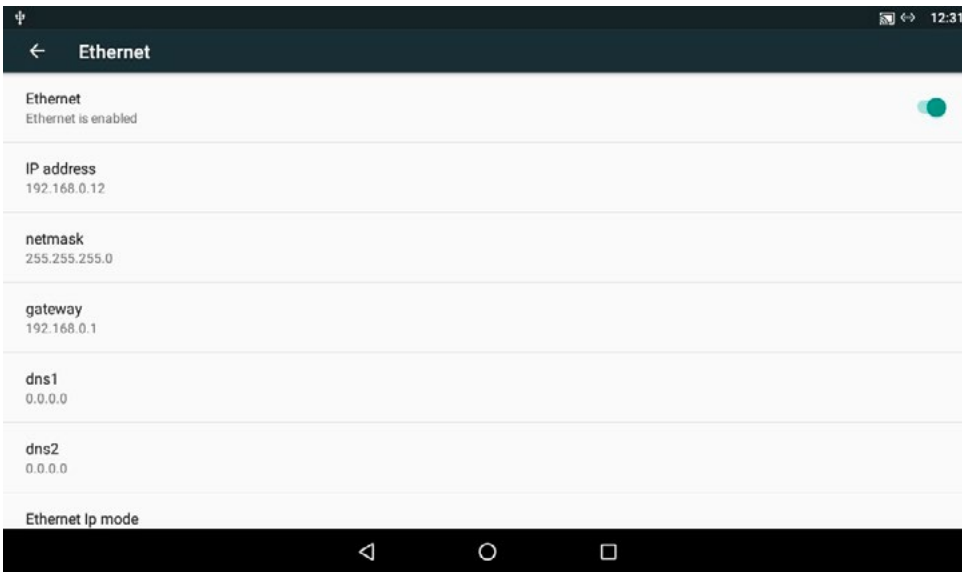


9. Gira URL-Endpoints (nur VI.r3)

Gira URL-Endpoints sind Verknüpfungen mit denen Sie verschiedene Funktionen Ihres Systems mit der LED oder dem Temperatursensor ihres luna Systems verknüpfen können. Sie können dabei die LED dem Zustand des Objektes oder Objekte dem Zustand der LED anpassen.Sie können über den integrierten Temperatursensor des luna Systems die Raumtemperatur auf ihrem Gira Home Server anzeigen.

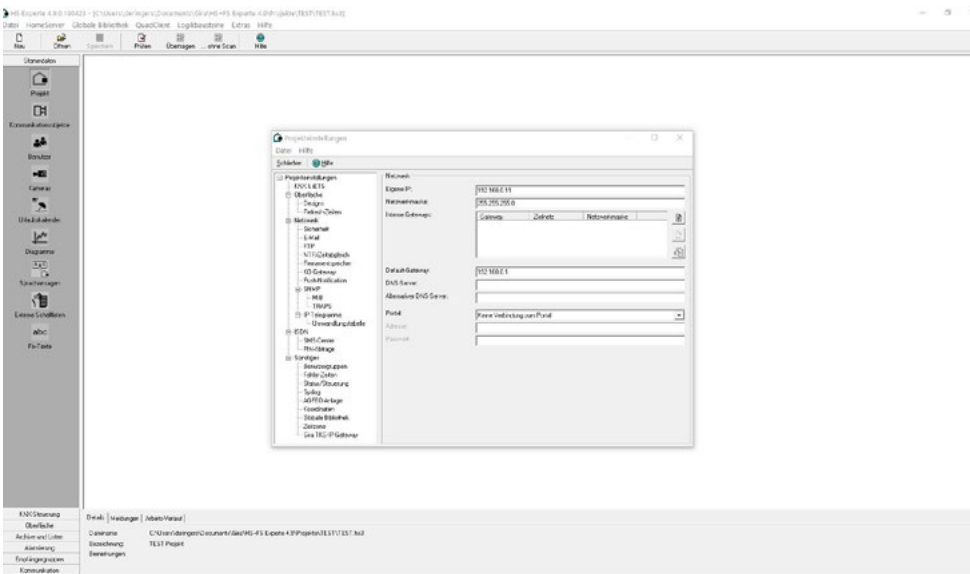
1 Starten Sie den Launcher als Administrator

2 Stellen Sie sicher, dass der Gira HomeServer (GHS) und das tci Touchpanel im selben Netzwerk sich befinden: Gehen Sie dazu auf „Settings > System > Android Settings > Ethernet & VPN > Ethernet“.



Netzwerkeinstellungen

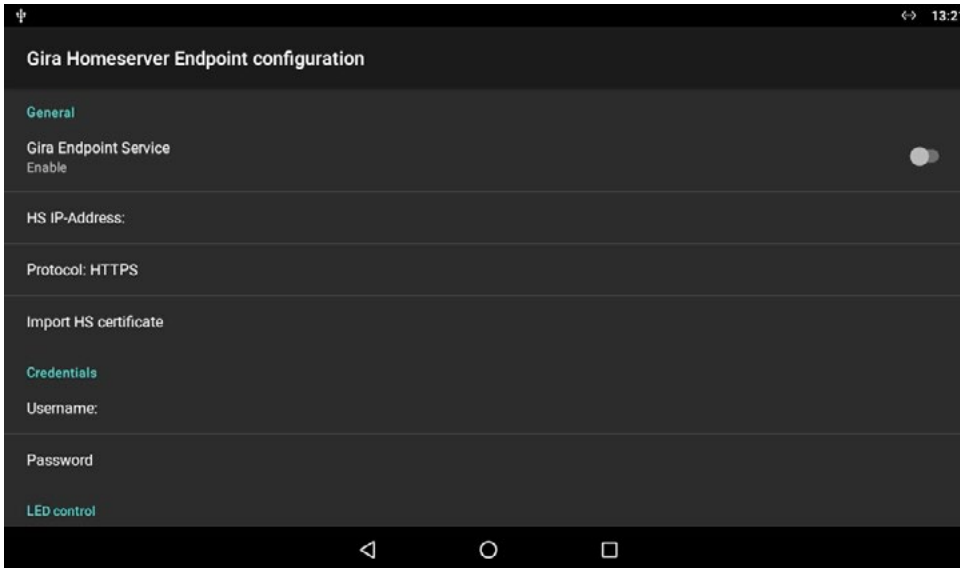
Vergleichen Sie die Netzmaske und das Gateway mit den Einstellungen Ihres GHS und stellen diese ggfls ein.



GHS Endpoints einrichten

Öffnen Sie die Seitenleiste aus und tippen Sie auf „Settings > Tools > Gira Homeserver Endpoints“. Geben Sie hier die IP-Adresse Ihrer GHS ein. Wählen Sie das Protokoll entsprechend Ihrer Einstellungen aus und importieren Sie das Zertifikat für Android. Eine Anleitung dazu finden Sie hier unter Zertifikat. Geben Sie hier den Benutzernamen Ihres GHS ein.

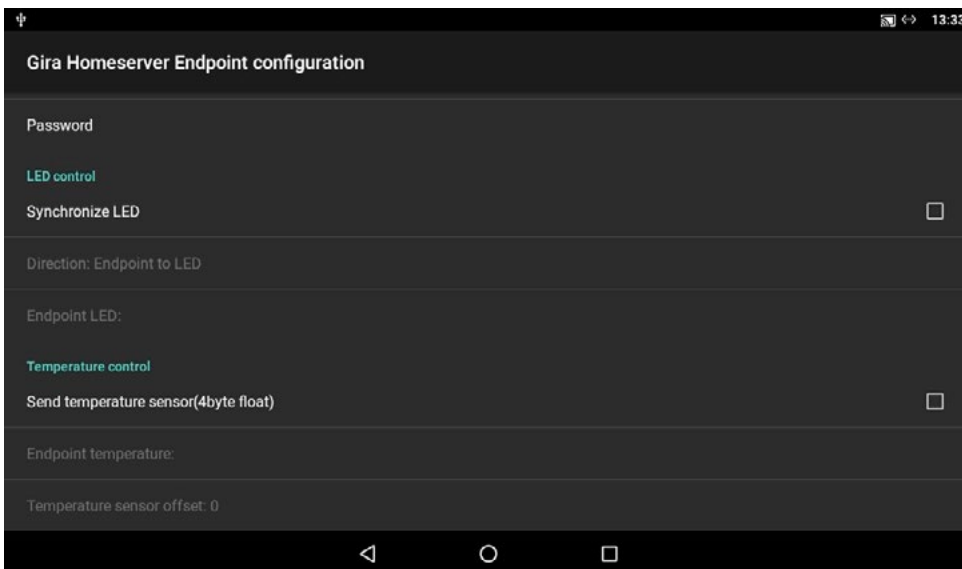
Geben Sie hier das Passwort zu Ihrem Benutzernamen ein. Setzen Sie nun das Häkchen hinter „Synchronize LED“.



Wählen Sie die Richtung der Steuerung:

Endpoint to LED: Die LED wird dem Zustand des Objektes angepasst.

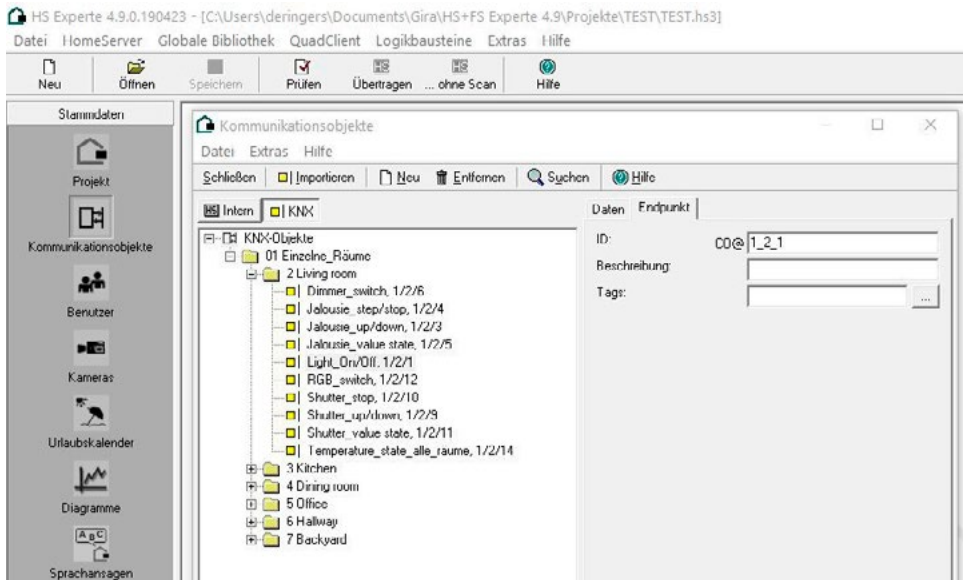
LED to Endpoint: Das Objekt wird dem Zustand der LED angepasst.





Die ID zur Adressierung des jeweiligen Endpoint finden Sie über den HS Experten heraus unter „Stammdaten > Kommunikationsobjekte > (gewünschter Endpoint) > Endpunkt“.

Geben Sie hier die ID des gewünschten Endpoints an mit dem Sie die LED verbinden möchten. Diese ist wie folgt aufgebaut „CO@X_X_X“, wobei die Platzhalter X die Haupt-, Neben-, und Untergruppe der vom Kommunikationsobjekt benutzten Gruppenadresse beinhalten.



Nach dem einrichten tippen Sie auf den „Enable-Schalter“ am Anfang der Seite. Ist alles richtig bertragen worden, lässt sich dieser aktivieren. Falls dies nicht der Fall sein sollte, achten Sie auf die Fehlermeldung auf dem Display und ggf. überprüfen Sie nochmal die Einstellungen.

10. Fehlerbehebung

Bei der Verwendung von exFAT formatierten Speichermedien kann es vorkommen, dass diese nicht erkannt werden: Verwenden Sie FAT32 oder NTFS formatierte Datenträger.

List of Content

1. Liability / Copyright	27
2. Device specific properties	
2.1 Rotation	28
3. Developeroptions	
3.1 Application development system	29
3.2 ADB driver	29
3.3 Description of the API for controlling temperature sensors, GPIO's and LED's	29
3.3.1 Temperature sensor	29
3.3.2 LED and sensor	30
3.3.3 GPIO	31
4. Launcher settings	32
4.1 Settings	34
4.2 Design settings	35
4.3 Complete app overview	36
4.4 widgets	36
4.5 Lock	36
4.6 Brightness	36
5. PanelControl	
5.1 Brightness control	37
6.2 Adaptive Brightness	38
6. App installation	
6.1 Standard installation	39
6.2 App installation without Google Playstore	39
6.3 App Installation with ADB (Android Debug Bridge)	39
7. App Updater 2.0	40
8. App-Watchdog	46
9. GIRA URL-Endpoints (only VI.r3)	47
10. Troubleshooting	49



1. Liability / Copyright

Copyright	<p>tci GmbH, Ludwig-Rinn-Str. 10-14, D-35452 Heuchelheim, Germany</p> <p>This manual, as well as the hard and software, which it describes is protected by copyright and may not be duplicated, translated or presented in any form without the written consent of tci GmbH.</p>
Note	<p>Publisher, translators and authors of this publication have carefully developed the texts, illustrations and programs. However, errors can not be completely ruled out. tci GmbH, shall neither warrant nor be held legally responsible for consequences which occur due to incorrect data.</p> <p>Information concerning errors are welcome at any time. The information in this operating manual does not guarantee definite product properties. Modifications concerning the support of technical progress will be reserved.</p>
Liability	<p>tci GmbH shall not be held liable for immediate damage occurring in connection with the supply or utilization of the documents. In addition, we shall not be held liable for any errors found in this publication. We are not obliged in any way to update the information contained in this document to latest standards.</p> <p>All rights reserved</p>

Last Update May 2023

2. Device specific properties

2.1 Rotation

The device does not have an inclination sensor. Due to this the display content will not rotate automatically by changing the device's position or orientation. The rotation of the screen content can be configured by means of the android settings. Applications which change the current content orientation may not be suitable for this device.



3. Developer options

3.1 Application development system

For developing own applications every development environment that is suitable for android can be used. Therefore no special requirements are necessary.

3.2 ADB Driver

A suitable driver for the device can be requested by the manufacturer. This driver will be completely compatible with the ADB tools and applications.

3.3 Description of the API for controlling temperature sensors, GPIO and LED

General: The temperature sensor, the GPIO's and the LED's are controlled via the Linux based file system of Android. All I/O operations can be carried out by simple read/write actions in the associated file.

The following options depends on equipment.

3.3.1 Temperature sensor

The built-in thermistor can be read out via the file system as outlined above. The outcome of the reading operation is a screened voltage between 0 and 1024. The result obtained by this method must be subsequently converted into the present resistance of the thermistor. The current resistance of the thermistor determined now can be used to calculate the temperature by means of the simplified Steinhart-Hart equation (B or β parameter equation).

Example:

Conversion of the voltage rating into the present resistance of the thermistor

Reading the screened voltage (rawValue) from the file: /look at cap. 3.3.2..)

VOLTAGE_INPUT = 1.8

VOLTAGE_MULTIPLIER = VOLTAGE_INPUT / 1024

R1 = 10000

voltageOut = rawValue * VOLTAGE_MULTIPLIER

$R = (\text{voltageOut} * R1) / (\text{VOLTAGE_INPUT} - \text{voltageOut})$

The resistance R can now be inserted in the Steinhart-Hart equation to calculate the current temperature in degrees Celsius.

Steinhart-Hart equation (B or β parameter equation)

The temperature in degrees Kelvin, R0 is the resistance at temperature T0 (25°C = 298.15K)

R0= 10 k Ω = 10000

B = 3435

T0 = 298.15

R = current resistance of the thermistor

T = temperature in degrees Kelvin

$$r_{\infty} = R_0 e^{-B/T_0}$$

$$T = \frac{B}{\ln(R/r_{\infty})}$$

Degrees Kelvin into degrees Celsius conversion

degrees Celsius = T - 273,15

3.3.2 LEDs and sensores

A red and a green LED are available. Both LEDs can be set independantly from each other. The access is provided through the file system again:

Rk3288-VI (depending model, see Type plate e.g. luna-X-Vi.r4)

LED red: ksp0630:red

LED green: ksp0630:green

LED switch on: echo 1 > /sys/class/leds/ksp0630:green/brightness

LED switch off: echo 0 > /sys/class/leds/ksp0630:green/brightness

brightness sensor: /sys/bus/iio/devices/iio:device0/in_voltage0_raw

Temperature sensor: /sys/bus/iio/devices/iio:device0/in_voltage1_raw

Rk3399-VI (depending model, see Type plate e.g. luna-x-VI.r5)

LED green on = echo 255 > /sys/class/leds/work/brightness

LED green off = echo 0 > /sys/class/leds/work/brightness

LED red on = echo 255 > /sys/class/leds/standby/brightness

LED red off = echo 0 > /sys/class/leds/standby/brightness

brightness sensor: /sys/bus/iio/devices/iio:device0/in_voltage0_raw

Temperature sensor: /sys/bus/iio/devices/iio:device0/in_voltage1_raw

Rk3288-XI (epending model, see Type plate e.g. F16-XI)

LED green on = echo 1 > /sys/class/leds/egpio_o1/brightness

LED green off = echo 0 > /sys/class/leds/egpio_o1/brightness

LED red on = echo 1 > /sys/class/leds/egpio_o2/brightness

LED red off = echo 0 > /sys/class/leds/egpio_o2/brightness

Rk3188-XI (epending model, see Type plate e.g. E07A-XI)

LED green on = echo 1 > /sys/class/gpio/gpio252/value

LED green off = echo 0 > /sys/class/gpio/gpio252/value

LED red on = echo 1 > /sys/class/gpio/gpio253/value

LED red off = echo 0 > /sys/class/gpio/gpio253/value



Rk3288-VI-solo (depending model, see Type plate e.g. solo or E04)

Brightness sensor: `/sys/devices/virtual/optical_sensor/lightsensor/ls_level`

Temperature sensor internal: `/sys/bus/iio/devices/iio:device0/in_voltage0_raw`

Temperature sensor external (optionally): `/sys/bus/iio/devices/iio:device0/in_voltage2_raw`

Proximity: `/sys/devices/virtual/optical_sensor/proximity/ps_level`

3.3.3 GPIO

Rk3288-VI (depending model, see Type plate e.g. luna-X-Vi.r4)

Floor call = `cat /sys/class/gpio/gpio251/value` (default value = 1)

Relays on= `echo 1 > /sys/class/gpio/gpio48/value`

Relays off= `echo 0 > /sys/class/gpio/gpio48/value`

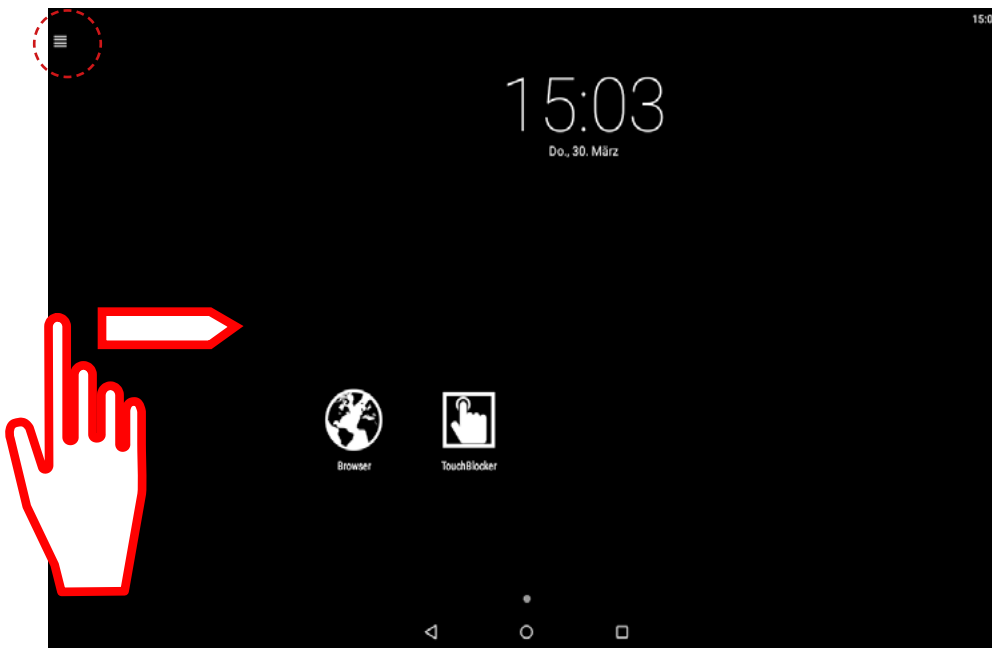
Rk3399-VI (depending model, see Type plate e.g. luna-X-Vi.r5)

Floor call = = `cat /sys/class/gpio/gpio102/value` (default value = 1)

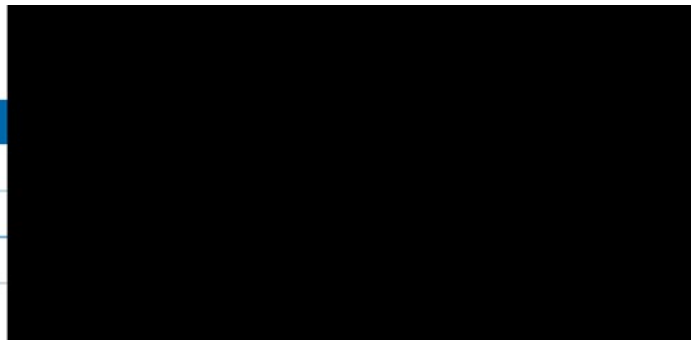
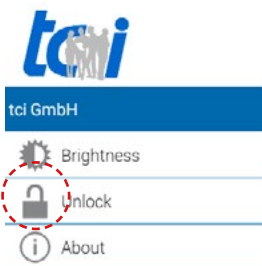
Relays on= `echo 1 > /sys/class/leds/gpio1/brightness`

Relays off= `echo 0 > /sys/class/leds/gpio1/brightness`

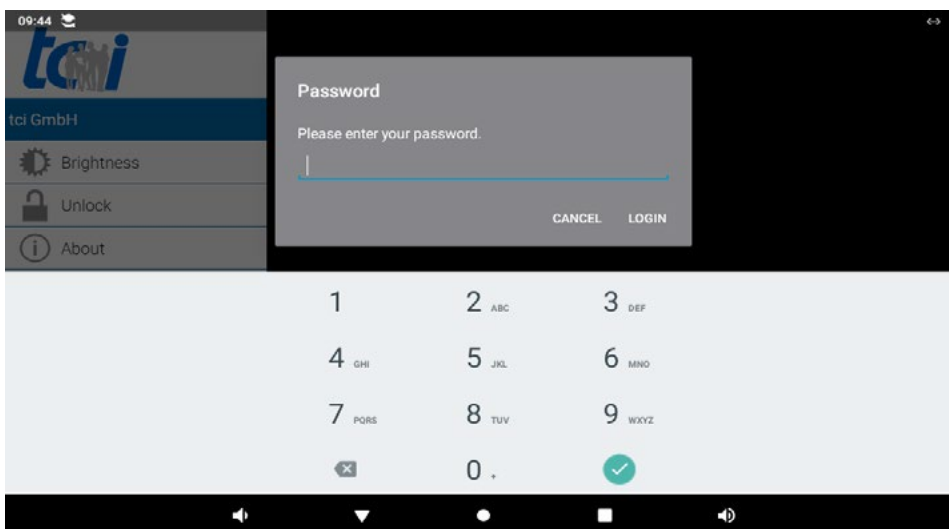
4. Launcher settings



Open the side menu bar by wiping from the left edge to the centre of the screen or by touching the menu icon in the upper left corner.



The Unlock item leads you to the password protected area. There you get access to the apps and the system settings.



After selecting unlock a dialog box for entering the password appears. There a two predefined users:

User: End user

Password: 0000

User: Integrator

Password: 8273

The „end user“ only has restricted possibilities to alter the settings, whereas the integrator is the almighty overlord at the end of the food chain, the sole ruler of the system and the end users' wallet.



End user restrictions


- Only access to display and audio settings
- Preinstalled apps as Dev Tools but also system apps are hidden in the „all apps“ overview, they neither can be fetched otherwise
- Applications cannot be installed or deinstalled

Login as integrator

After a log in the side bar appears with new items. The following items are available if you logged in as integrator:



tci GmbH

 Brightness

 Settings

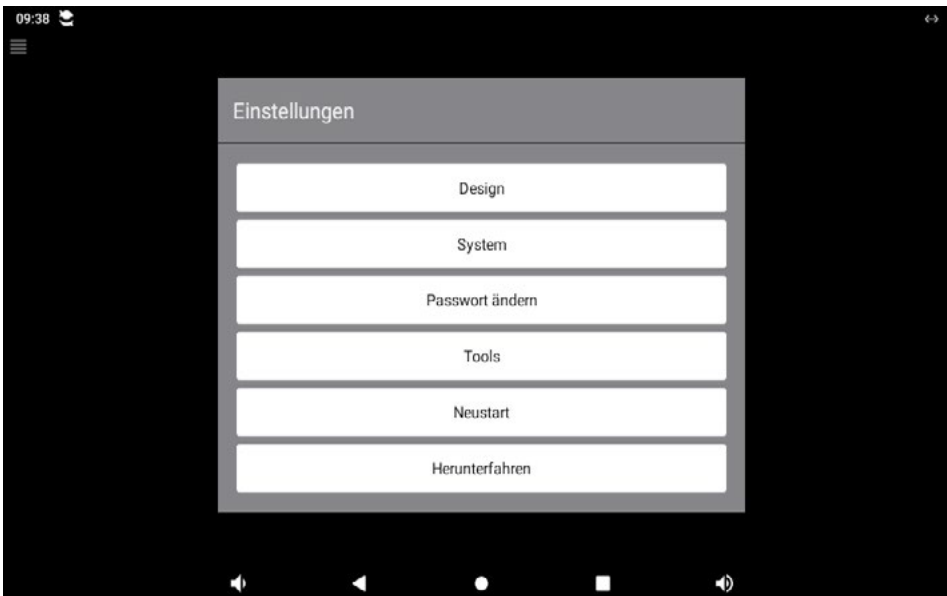
 All Apps

 Widgets

 Lock

 About

4.1 Settings



Design

Modifications that affects the background image and the look and feel of the launcher. Please see point 2 „Design“ for further information.

System

Leads you to the standard android settings, as display, sound & notifications, ethernet, memory, apps, language, accounts, etc. The standard android options will not be discussed in further detail here.

Change Password

Changes the password of the user who is currently registered.

Tools

- Backup: Saves a backup of the present launcher settings.
- Restore: With this item you can restore the launcher from a backup if malicious settings were made (except for widgets and background images). The Restore function equally can be used to set up multiple systems.
- LLauncher settings reset: Removes all widgets and apps from the start screen.

Running the kiosk mode

Set the system to kiosk mode and removes the option menus for design and system. The task bar will be deactivated too. Applications that don't have an explicit exit function cannot be quitted any more.

Turn on ADB ethernet

Activates the ADB interface via ethernet. With the command: „adb connect 192.168.0.25:5555“ in the ADB shell establishes a connection. The shown IP address is just an example. Please insert the IP address of your system in this field.

Reboot

Starts the system again.

Shutdown

Turns the system off.



4.2 Design settings



Logo

Changes the logo that is shown in the upper left corner of the launcher

tci GmbH (Name)

Alters the name that appears beneath the logo (1).

Background Color (title)

Changes the background color of the text (2) in the launcher

Text Color (Title)

Changes the text color of the freely selectable text

Solid Wallpaper

Changes the screen background to a single color. It can be activated or deactivated by checking or unchecking and must not be active if a background image shall be shown. A wallpaper (background image) can be set in the standard android settings.

Save

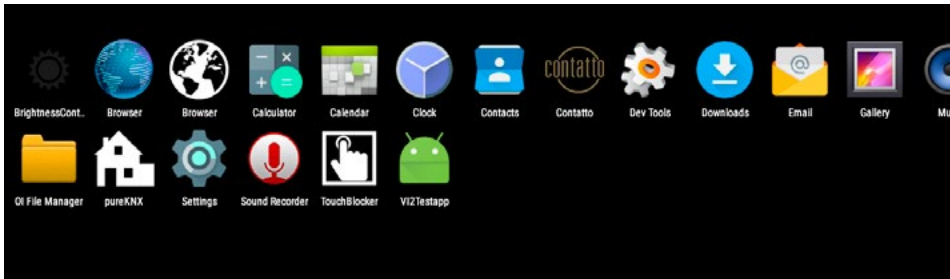
Saves the set text, logo and settings.

Cancel

Discards the made settings

4.3 Complete app overview

Shows an overview of all installed applications. You can drag and drop any desired app to the home screen. To remove an app from the home screen, just drag and drop it into the trashcan. You must be logged in with a password to move apps and widgets.



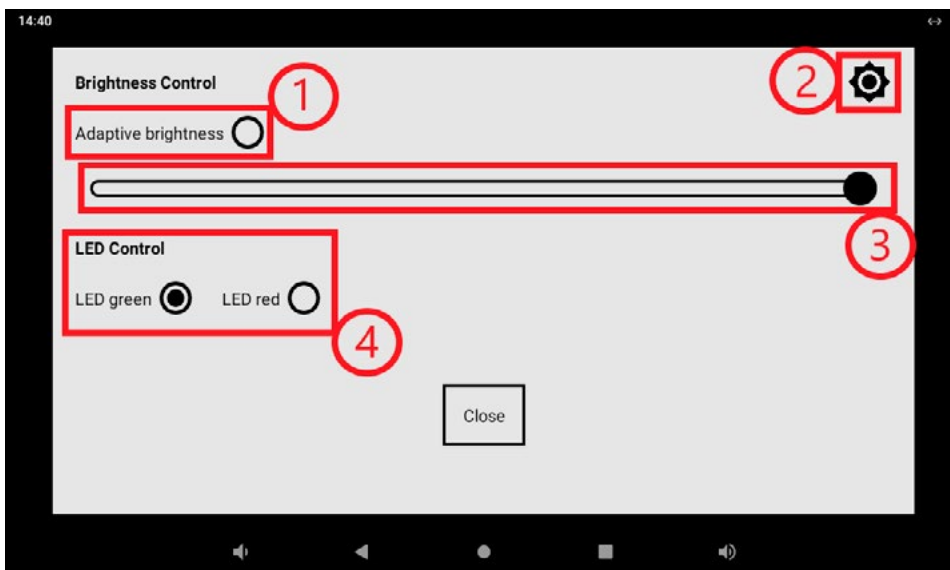
4.4 Widgets

Shows you all the little apps (widgets) for the home screen, like time, calendar and other useful little gadgets. Hold and drag a widget to put it on the home screen or remove widgets from the home screen by dragging and dropping it into the trashcan. Sperrt das System gegen unbefugten Zugriff auf die Systemeinstellungen. Wenn Einstellungen mit dem Integrator Benutzer getätigt wurden, werden dazu alle laufenden Applikationen beendet.

4.5 Lock

Locks the system to prevent unauthorised access to the system settings. If changes are initiated by the user „integrator“, all running applications will be closed for it.

5. PanelControl



following options depends on equipment

5.1 Brightness control

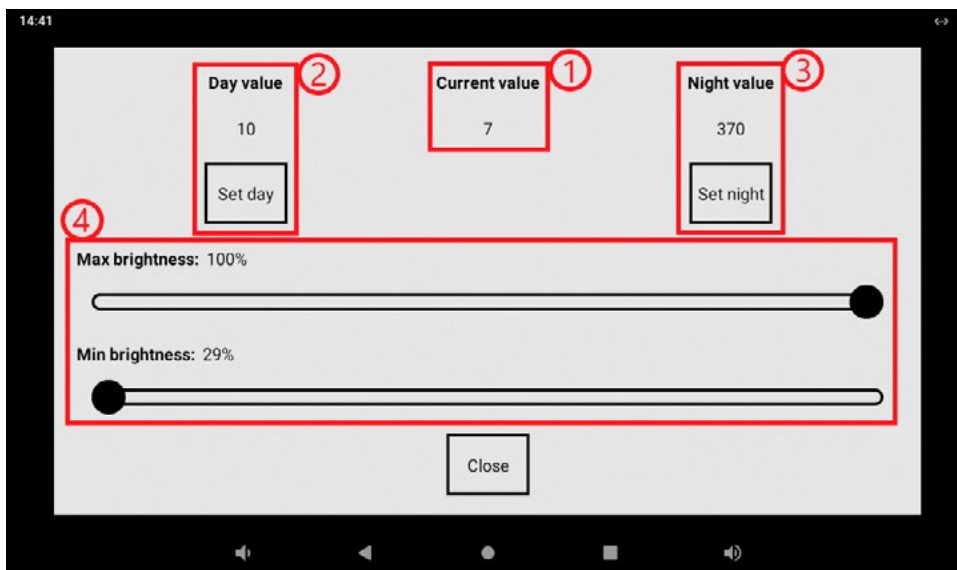
PanelControl is used as a tool für control the brightness.

1. Adaptive brightness. When activated for the first time, the screen brightness automatically adjusted to the ambient light. A message appears that you need to calibrate the sensor.
2. This icon opens the menu for calibrating and setting the sensor brightness mode automatically adjust the brightness.
3. Use the slider to set the brightness manually and permanently. The automatic brightness control is thus deactivated.
4. With LED Control you can set the status LED at the top of the touch

5.2 Adaptive brightness

Please note that the settings should be made when the device is installed.

1. Current value shows the current, constantly changing sensor value.
2. Day value; here you set the day value [Set day], from this value the display is set to the maximum brightness on
3. Night value; Here you set the night value [Set night], if the sensor value falls below this specification, it turns on set the display to the lowest brightness level. Ideally, you should also make these entries at night.
4. Use the Max brightness and Min brightness sliders to limit the overall screen brightness. The screen brightness changes linearly to the prevailing lighting conditions.





6. App installation

6.1 Standard installation

To install an APK (APK file), please log in as Integrator as described in chapter 5.

Copy the desired app to an USB stick and connect it to one of the USB ports of your device. Now navigate to „all apps“ in the settings menu. Open the „Explorer“ app afterwards. The content of your USB stick can be found in the „USB“ menu item. Clicking on the desired APK file will start the installation.

If an APK is already installed it will be updated then.

6.2 App installation without Google Playstore

You can get the needed APK files (APK = Android Package, format of installable software for android systems) without a direct access to the Playstore:

1. Receiving the APK file from the manufacturer directly

A lot of manufacturers of home automation possess their own apps to control IP based intercoms for example. Often the simplest way to get the needed files is to contact the manufacturer directly.

2. Obtaining the APK from the Playstore with the aid of an android smartphone or tablet

If you have another android device, such as a smartphone or a tablet with a Playstore account, you can get your app in this way

- First install the desired app on your smartphone or tablet. You will install it on your tci unit later on
- Now look for an app for exporting APK files like „SaveAPK“ and install it in addition
- Start the app „SaveAPK“ and choose the app you want to export and the storage path
- Copy the exported APK file from the smartphone / tablet to an USB stick and connect it to your tci device afterwards
- Finally start the file manager of the tci device and navigate to the USB stick (/storage/usb_host). Click on the desired APK file to install the app.

3. Receipt of apps from the Playstore without a smartphone or tablet

If you do not have another android device available, it is possible to obtain the needed APKs from various websites that allow to export the files from the Playstore to a PC. Open the Playstore by means of a browser on your PC and navigate to the desired app. Open the site: <http://apps.evozi.com/apk-downloader/> for example in a second tab.

Copy the URL from the Playstore and paste it into the input box. Click on „Generate download link“. Now you can download the APK to your PC. Use an USB stick to install the app on your tci device as described in item 2.

6.3 App installation with ADB (Android Debug Bridge)

You can use the Android Debug Bridge for installing apps if the system does not have a USB port.

Therefore either install the Android SDK (containing the ADB tool) or for example the leaner „Minimal ADB and fastboot“ tool on your PC.

Subsequently open the ADB console and type in the following command:

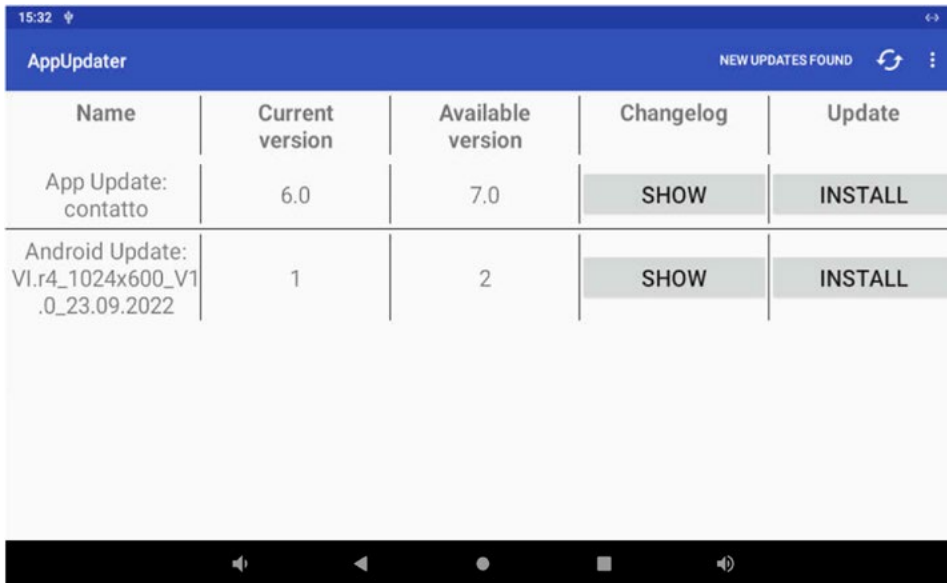
```
adb install path_to_apk
```

To establish a connection via ADB ethernet please see chapter 4 „Launcher settings“.

7. App Updater 2.0

The tci AppUpdater can be used to keep your Android system up to date, including both app and Android updates.
The tci AppUpdater works with HTTP links..

Manuell nach Updates suchen



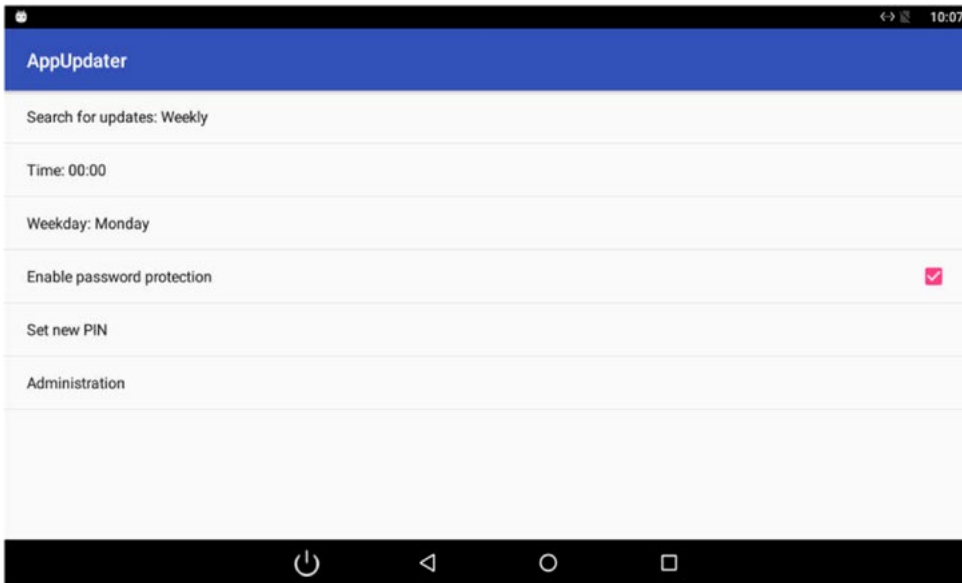
The screenshot shows the tci AppUpdater interface on an Android device. The top bar is blue with the text 'AppUpdater' and 'NEW UPDATES FOUND' with a refresh icon. Below the bar is a table with the following data:

Name	Current version	Available version	Changelog	Update
App Update: contatto	6.0	7.0	SHOW	INSTALL
Android Update: Vi.r4_1024x600_V1.0_23.09.2022	1	2	SHOW	INSTALL

1. Start the AppUpdater
2. Click on the Icon in the action bar to search vor updates
3. If updates are available, they will be listed in a table
4. Clicking "Update" will download and install the corresponding app

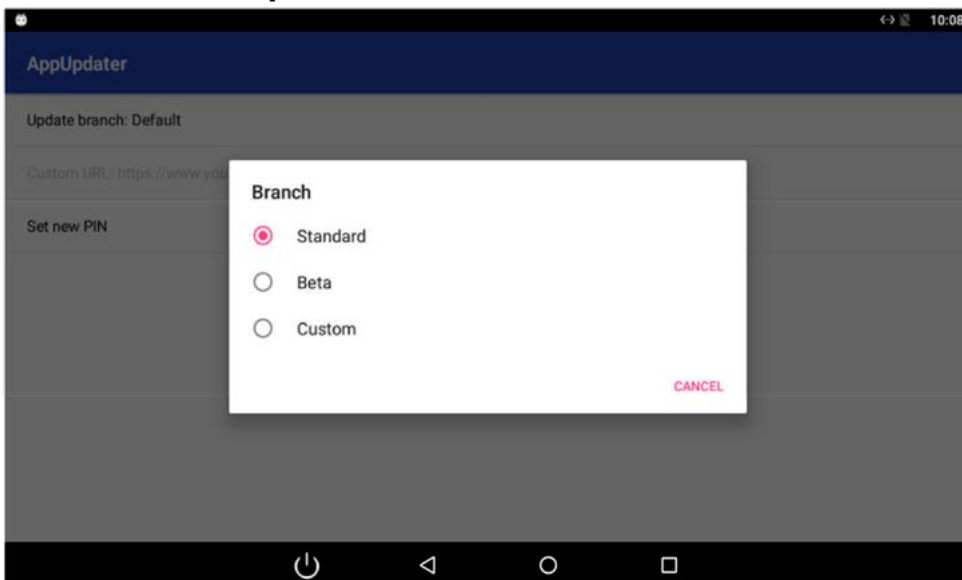


Settings and automatic search for updates



1. Search for updates: Allows a time-based search for updates
 - a. Never: Never search for updates
 - b. Daily: Checks for updates daily at the specified time
 - c. Weekly: Searches for updates weekly on the designated day and at the specified time
2. Enable password protection: Turns on/off the password protection for the settings menu
3. PIN: Let you set the PIN (default is: 0000)
4. Administration: Opens the advanced configuration for selecting the update branch

Selection of the update branch



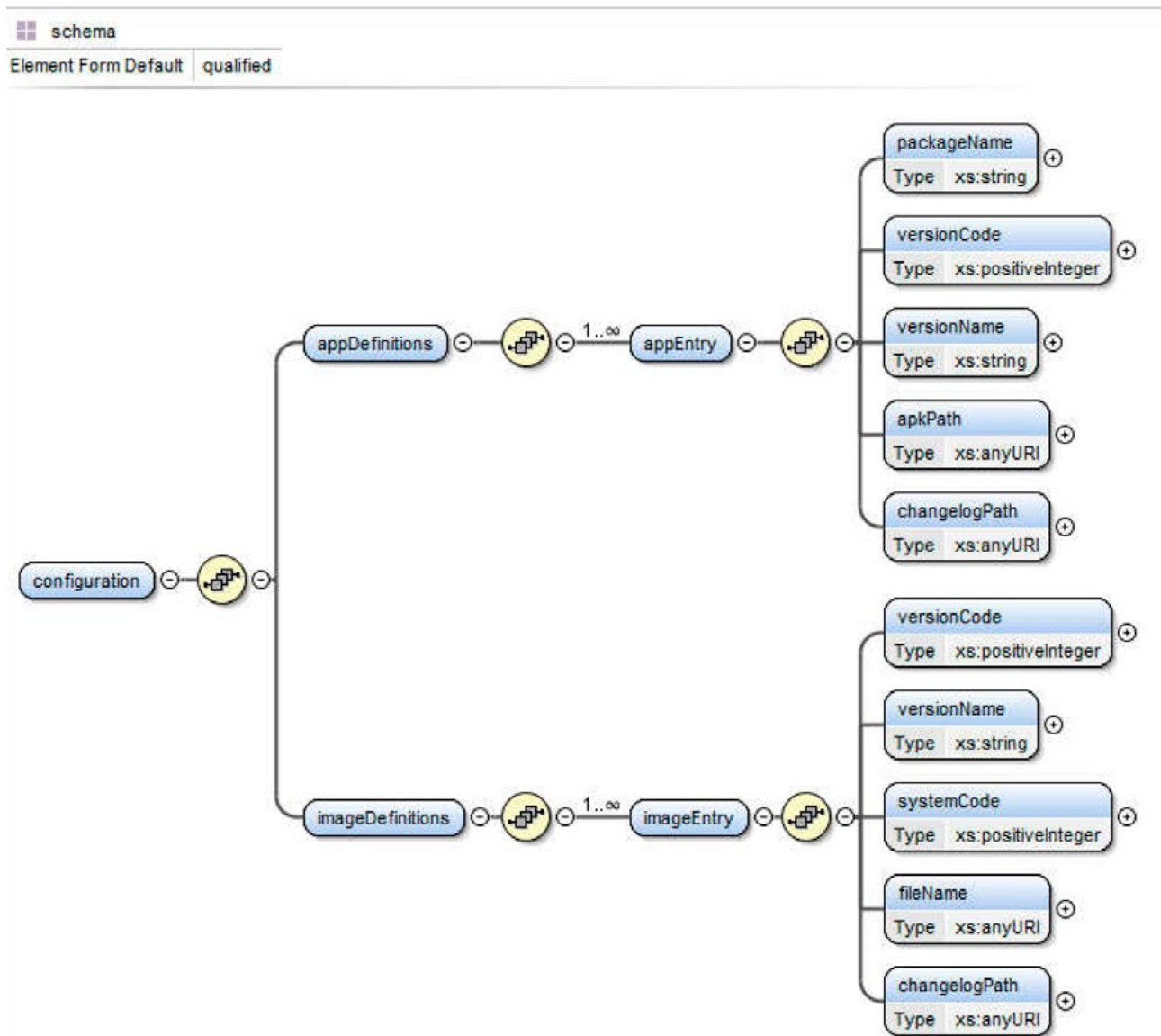
1. Allows you to select the update branch (Standard, Beta and Custom)
2. If you choose the “Custom” branch, you must configure the path to the configuration file on your web server yourself. Please refer to the “Configuration” section on the following page for details.
3. Let you define a new PIN (default: 1234)

Configuration

Setting up the configuration file "UpdateConfig.xml".

The "UpdateConfig.xml" file configures the available updates. Afterwards, the "UpdateConfig.xml" file must be made available online as an HTTP link

XSD Definition





```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified">
  <xs:element name="configuration">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="appDefinitions">
          <xs:complexType>
            <xs:sequence>
              <xs:element maxOccurs="unbounded" name="appEntry">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element name="packageName" nillable="false" type="xs:string"/>
                    <xs:element minOccurs="1" name="versionCode"
                      type="xs:positiveInteger"/>
                    <xs:element minOccurs="1" name="versionName" type="xs:string"/>
                    <xs:element maxOccurs="1" name="apkPath" type="xs:anyURI"/>
                    <xs:element minOccurs="1" name="changelogPath" type="xs:anyURI"/>
                  </xs:sequence>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element name="imageDefinitions">
          <xs:complexType>
            <xs:sequence>
              <xs:element maxOccurs="unbounded" name="imageEntry">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element name="versionCode" type="xs:positiveInteger"/>
                    <xs:element name="versionName" type="xs:string"/>
                    <xs:element name="systemCode" type="xs:positiveInteger"/>
                    <xs:element name="fileName" type="xs:anyURI"/>
                    <xs:element name="changelogPath" type="xs:anyURI"/>
                  </xs:sequence>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Tag Definition

Configuration:	Beginning and end of configuration definition
appDefinitions:	opening tag for all app updates
appEntry:	Initial tag for an app update definition
packageName:	the package name of the application, for example, de.tci.contatto
versionCode:	An integer sequential value for version checking, used to determine which version is newer. For example, „123“
versionName:	An alias for the versionCode, for example, „7.1“
apkPath:	The HTTP link to download the APK file
changelogPath:	Der http Link zu einer Change log Textdatei
imageDefinitions:	Einleitendes Tag für alle Android Image-Updates
imageEntry:	Einleitendes Tag für eine Android Image Update Definition
versionCode:	Ein ganzzahliger fortlaufender Wert zur Versionsprüfung, wird verwendet um zu entscheiden welche Version neuer ist. Z.B. „123“
versionName:	n alias for the versionCode, for example, „7.1“
systemCode:	unique image ID
fileName:	The HTTP link to download the image file
changelogPath:	he HTTP link to a changelog text file

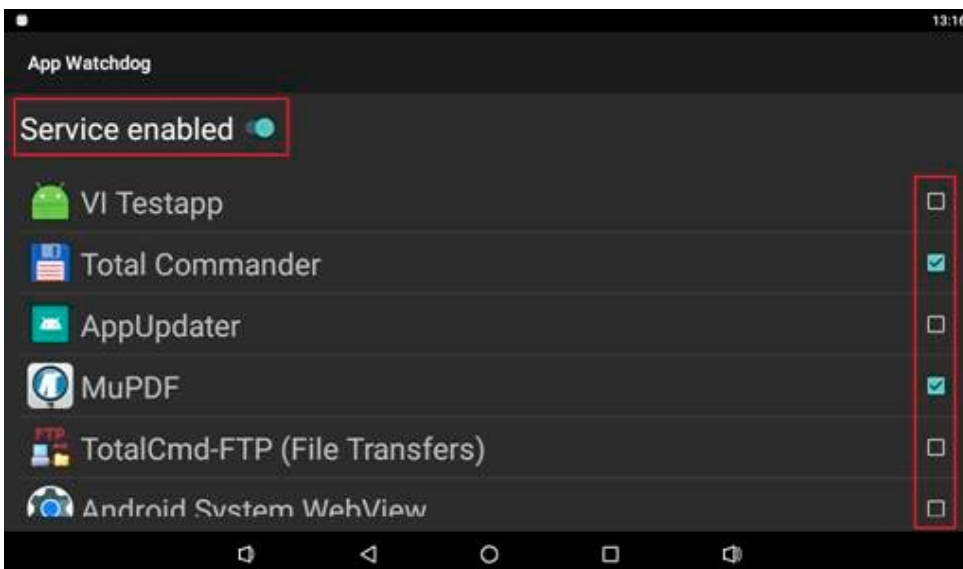


Example of a complete UpdateConfig.xml file

```
<?xml version="1.0" encoding="UTF-8"?>
<configuration xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="file://D:/ConfigScheme.xsd">
  <appDefinitions>
    <appEntry>
      <packageName>de.tci.contatto</packageName>
      <versionCode>8</versionCode>
      <versionName>7.0</versionName>
      <apkPath>http://tci.de/contattoVI_7_0.apk</apkPath>
      <changelogPath>http://tci.de/contatto_changelog.txt</changelogPath>
    </appEntry>
    <appEntry>
      <packageName>net.spidercontrol.knxserver</packageName>
      <versionCode>14</versionCode>
      <versionName>7.4</versionName>
      <apkPath>http://tci.de/knxServer.apk</apkPath>
      <changelogPath>http://tci.de/knxServer_changelog.txt</changelogPath>
    </appEntry>
  </appDefinitions>
  <imageDefinitions>
    <imageEntry>
      <versionCode>3</versionCode>
      <versionName>2.1</versionName>
      <systemCode>1337</systemCode>
      <fileName>http://tci.de/update_1337.zip</fileName>
      <changelogPath>http://tci.de/update_1337_changelog.txt</changelogPath>
    </imageEntry>
    <imageEntry>
      <versionCode>2</versionCode>
      <versionName>2.0</versionName>
      <systemCode>4711</systemCode>
      <fileName>http://tci.de/update_4711.zip</fileName>
      <changelogPath>http://tci.de/update_4711_changelog.txt</changelogPath>
    </imageEntry>
  </imageDefinitions>
</configuration>
```

9. AppWatchdog

The AppWatchdog is located in the „Tools“ window of the launcher (log in with 8273).



In the AppWatchdog you will find a list of all installed apps. Select the desired apps by activating the corresponding checkbox.

After that activate the AppWatchdog itself (first item: Service enabled).

The AppWatchdog now monitors for all selected apps whether the app or its process is still running. If this is not the case, the respective app is restarted. In addition, all selected apps are terminated and restarted at 2 o'clock in the morning.

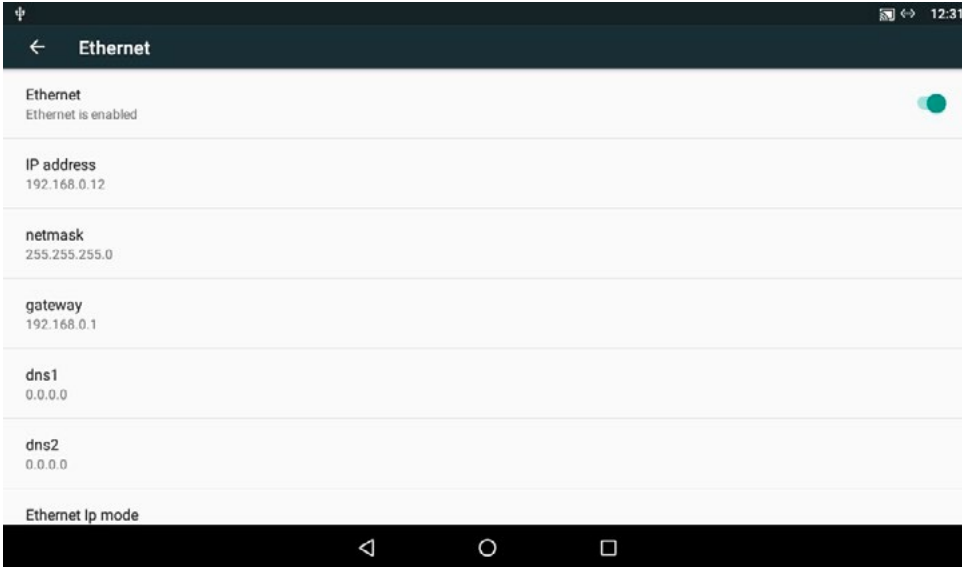


10. Gira URL-Endpoints (only VI.r3)

Gira URL endpoints are links with which you can link various functions of your system with the LED or the temperature sensor of your luna system. You can adjust the LED to the state of the object or objects to the state of the LED. You can display the room temperature on your Gira Home Server via the integrated temperature sensor of the luna system.

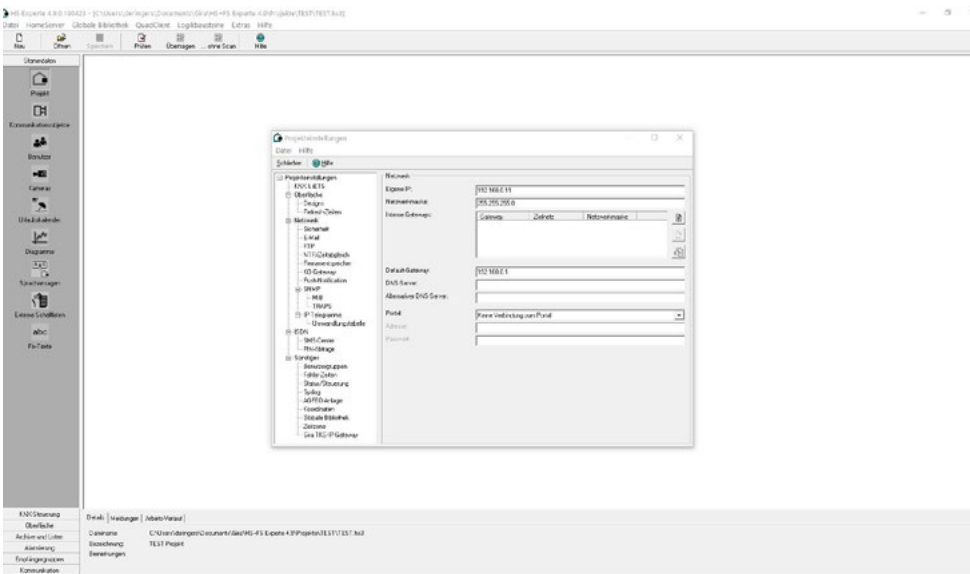
1 Start the Launcher as Administrator

2 Make sure that your devices (GHS and luna) are on the same network. To do this, go to „Settings > System > Android Settings > Ethernet & VPN > Ethernet“.



Network settings

Check the netmask and the gateway of the GHS and adjust them.

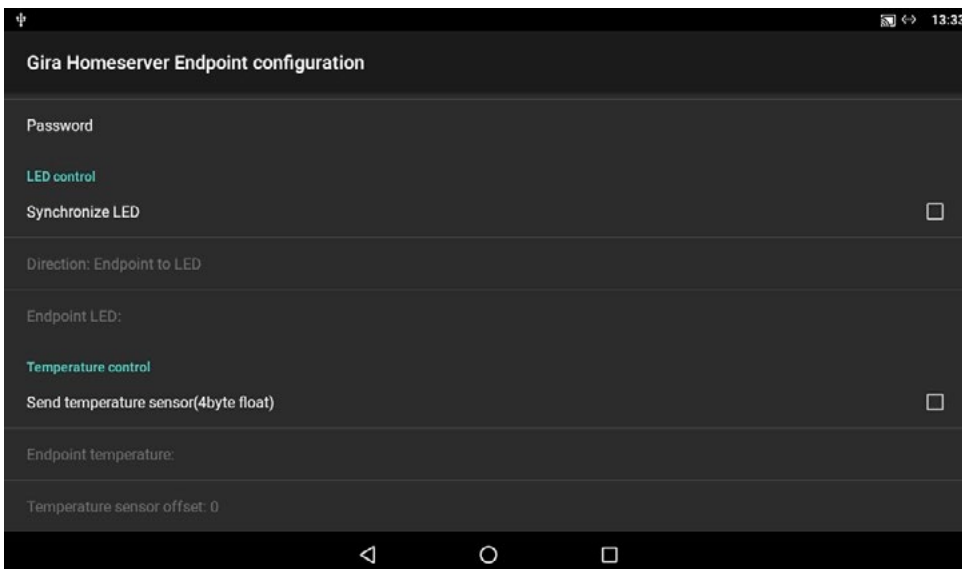


GHS Endpoints set up

Open the sidebar and tap on „Settings > Tools > Gira Homeserver Endpoints“ 10.3.2. Enter the IP address of your GHS here. Select the protocol according to your settings and import the certificate for Android. Instructions can be found here under Certificate. Enter the user name of your GHS and the password. Now place a tick with „Synchronize LED“.



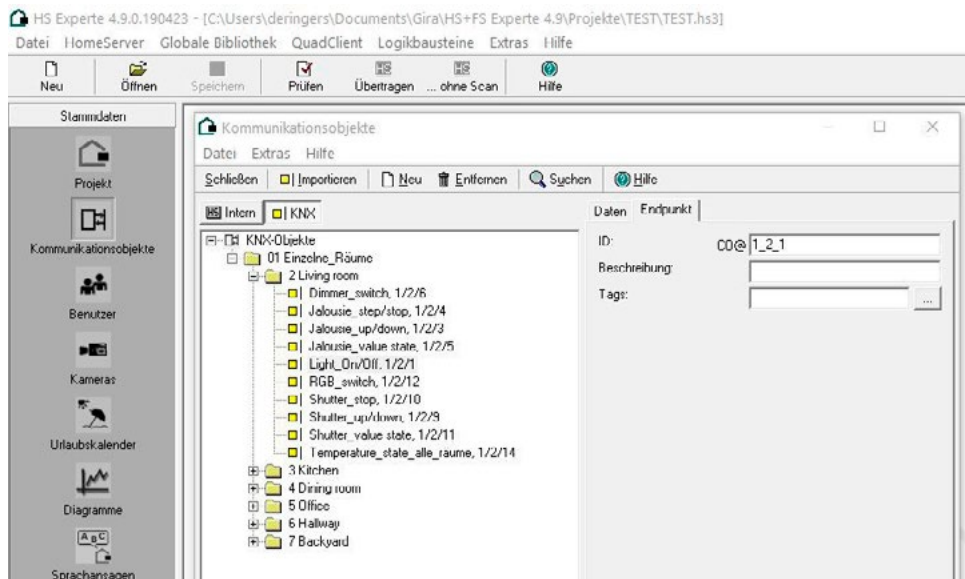
Select the direction of the control. Endpoint to LED: The LED is adapted to the state of the object.
LED to Endpoint: The object is adapted to the state of the LED.





The ID for addressing the respective endpoint can be found via the HS expert under „Master data > Communication objects > (desired endpoint) > Endpoint“.

Enter the ID of the desired endpoint to which you want to connect the LED. This is structured as follows „CO@X_X_X“, where the placeholders X contain the main group, secondary group and subgroup of the group address used by the communication object.



After setup, tap the „Enable“ button at the top of the page. If everything has been transferred correctly, this can be activated. If this is not the case, please pay attention to the error message on the display and check the settings again.

11. Troubleshooting

It may occur that exFAT formatted volumes cannot be detected. Please use storage devices that are formatted as FAT32 or NTFS.



verbindet Menschen und Technik