HomeKit Plugin

The **HomeKit** plugin for Wiser for KNX/spaceLYnk controller is designed to control KNX devices in the installation through Apple´s **Home** application, available in Apple devices like iPhone, iPad, iWatch, Mac, Apple TV, etc. (hereafter referred to as the app/mobile application).

System Requirements

To ensure optimal performance and compatibility with HomeKit plugin, please make sure you meet the following system requirements:

Hardware	Wiser for KNX or spaceLYnk controller	Ensure you have a hardware version greater than 2.
Firmware	Latest firmware	Make sure you have the latest firmware installed. The latest firmware is available for download at se.com.
Software	Plugin	Ensure the KNX IoT 3rd Party API plugin is installed and enabled (for Local). Enabling automatic updates is recommended to ensure you always have the latest version of the plugin installed.

Limitations of the HomeKit Plugin:

- Control of slats in blinds (not supported).
- Stop commands for shutters/blinds.
- Heating works only in the room temperature controller.
- KNX scenes cannot be called, but new scenes can be created in the **Home** app.
- Separate settings from the **Touch** plugin.
- Terminology/naming of group objects is different from KNX conventions.
- In larger or more complex KNX installations, the app may behave unreliably.

Home App

The **Home** app can be operated in two ways:

1. **Without a home hub**: You can control the KNX installation locally from your mobile device using the **HomeKit** plugin.
2. **With a home hub (e.g., Apple TV)**: With a home hub, you can access the installation remotely via the internet through Apple Cloud. Additionally, you can set up automations/scenes and share your home with other users.

Main features of the Apple Home app

The **Home** app is a centralized hub to control and manage your smart home devices. Here are the key features:

1. **Device control**: You can manage a variety of smart devices such as lights, thermostats, cameras, and locks directly from the app. Toggle devices on or off and adjust settings to suit your preferences.
2. **Scenes**: Create customized scenes that allow you to control multiple devices simultaneously.
3. **Automations**: You can set up automations to trigger actions based on specific conditions such as time of day, location, or device activity.
4. **Secure remote access**: Access and control your devices remotely through an Apple TV or HomePod acting as a hub. This ensures you can monitor and manage your home even when you're away.
5. **User sharing**: Share control of your devices with family members and friends, allowing them to access and manage devices from their own Apple devices.
6. **Room and zone management**: Organize your smart home devices by assigning them to specific rooms and zones within the app.

For more information about the Apple **Home** app, visit https://www.apple.com/uk/home-app/.

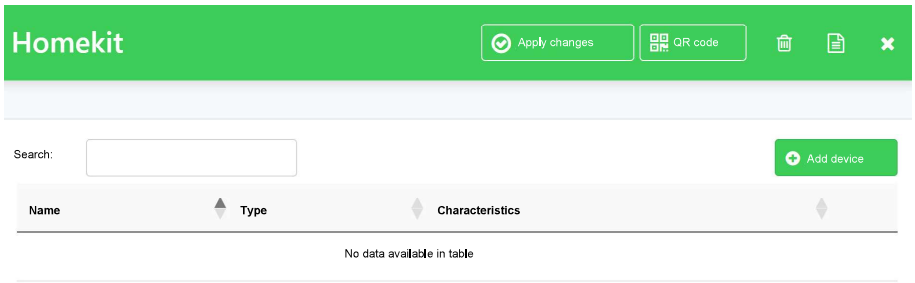
You can find the user guide for the Apple **Home** app at https://support.apple.com/.

HomeKit Plugin Installation

1. Log in to the controller through your web browser.
2. Click on  in the top right corner to open the **Marketplace**.
3. Click on the green bar in the middle to display the applications available for installation under **Install new apps and widgets**.
4. From the list of available applications, select the **HomeKit** plugin and click  on the right to start the installation.
5. Return to the **Start page** of your controller in the web browser. The installed **HomeKit** plugin will appear among other applications:

Homekit

6. Click on the **HomeKit** icon to open the plugin:



NOTE: We recommend enabling the plugin's automatic update to ensure you always have the latest version.

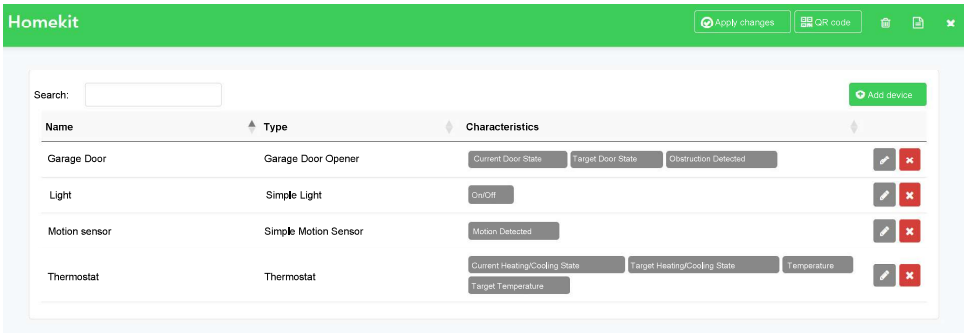
Adding Devices into HomeKit Plugin

To control your devices through the **Home** app, you need to create new widgets within the **HomeKit** plugin. This allows you to control your devices directly from the app on your Apple device.

Follow these steps:

1. Click on the **Add device** button in the top right corner.
2. Name your device: Give it a unique identifier.
3. Select the type of device: Available objects for the device will appear. Objects are divided into three groups (see more here: Types of Devices Available in the HomeKit Plugin, page 111):
   - **Mandatory** (marked with a star): These need to be mapped for the device to be added.
   - **Required**: Necessary for the proper functioning of the device.
   - **Optional**: These can be mapped at your discretion.

After mapping the objects, click on **Save**, and your device will appear in the **HomeKit** plugin device list.



For each device, all mandatory objects are displayed in the **Characteristics** column.

Finally, click on the **Apply changes** button to save the updates.

NOTE:

- The **Apply changes** button will turn red when there are new changes that need confirmation, signaling that you have changes pending to be saved.
- If you want your changes to be saved, click on **Apply changes**.
- This button ensures that the changes you make will be reflected in the mobile application.

After adding devices to the **Home** app, you can assign them to your preferred room structure. This structure can match or differ from your controller's visualization.

Adding Accessory to the Home App via QR Code

After mapping the devices in the **HomeKit** plugin, transfer them to the **Home** app using a bridge created by the mobile app. This bridge represents the controller. To complete the transfer, use the QR code generated by the **HomeKit** plugin.

Follow these steps:

1. Make sure that your mobile device and **HomeKit** plugin are connected to the internet on the same network.

2. In the **HomeKit** plugin, on the main screen, click on the **QR code** button at the top right. A **Setup code** dialog will open with a generated QR code for pairing.
3. Open the mobile application on your mobile device, and tap **Add Accessory**.
4. Point your mobile device at the QR code in the **HomeKit** plugin.
5. The mobile app will load a bridge from your **HomeKit** plugin. You can then name your devices using the wizard and assign them to individual rooms in the mobile app.

NOTE: The Apple **Home** app supports adding multiple bridges, allowing you to integrate other Apple **HomeKit**-compatible systems and devices, such as cameras and sensors.
Additionally, the Apple **Home** app enables you to create automations and scenes that combine all the devices integrated into the **Home** app.
For more information, please refer to the Apple **HomeKit** user guide.

Editing and Deleting Devices in HomeKit Plugin

You can later edit/delete the devices mapped in the **HomeKit** plugin:

1. Editing devices:
   - Click  on the right of your device's row.
   - Make the intended changes.
   - Click on **Save** and then **Apply changes** to save the updates.

2. Deleting devices:
   - Click  on the right of your device's row.
   - In the **Delete device** dialog, click on **Delete**.
   - Remember to click on **Apply changes**.

Types of Devices Available in the HomeKit Plugin

This list includes all types of devices available in the **HomeKit** plugin, along with their available characteristics (objects) for mapping and DPT. It details the naming and structure of the **HomeKit** plugin based on Apple's **Home** app specification. To simplify the creation of widgets, we have added information on the equivalent naming for the group objects currently used in the **Touch** widget configuration, where available.

Simple Controls

Device type	Required characteristics (* Mandatory)	Parameters	Equivalent widget (Touch)	Group objects (Touch)
SIMPLE LIGHT	On/Off Control * (bool) On/Off Status (bool)		Light switch	Switch object Status feedback object
SIMPLE TEMPERATURE SENSOR	Temperature Status * (float16)		Co_2/humidity/temp sensor	Temperature value

Device type	Required characteristics (* Mandatory)	Parameters	Equivalent widget (Touch)	Group objects (Touch)
SIMPLE HUMIDITY SENSOR	Humidity Status * (scale)		Co_2/humidity/temp sensor	Humidity value
SIMPLE LIGHT SENSOR	Light Status * (float16)		Weather station Motion sensor	Brightness value
SIMPLE LEAK SENSOR	Leak Detected Status * (bool)		Water leak alarm	Alarm object, alert
SIMPLE MOTION SENSOR	Motion Detected Status * (bool)		Motion sensor	Detection
SIMPLE OCCUPANCY SENSOR	Occupancy Detected Status * (bool)		Motion sensor	Detection
SIMPLE SMOKE SENSOR	Smoke Detected Status * (bool)		Fire alarm	Alarm object, alert
SIMPLE OUTLET	On/Off Control * (bool) On/Off Status (bool)		Socket switch	Switch object Status feedback object
SIMPLE FAN	Active Control * (bool) Active Status (bool)		Fan switch	Switch object Status feedback object
SIMPLE SWITCH	On/Off Control * (bool) On/Off Status (bool)		General switch	Switch object Status feedback object
SIMPLE AIR QUALITY SENSOR	Air Quality Status * (uint8)	0 = Unknown, 1 = Excellent, 2 = Good, 3 = Fair, 4 = Inferior, 5 = Poor	N/A	
SIMPLE CARBON MONOXIDE SENSOR	Carbon Monoxide Detected Status * (bool)		Air quality sensor	Carbon monoxide
SIMPLE CONTACT SENSOR	Contact Sensor State Status * (bool)		Window/Door sensor, Any switch	Switch object

Other Controls

Device type	Required characteristics (* Mandatory)	Parameters	Equivalent widget (Touch)	Group objects (Touch)
LIGHT ON/OFF BRIGHTNESS	On/Off Control * (bool) On/Off Status (bool) Brightness Control * (scale) Brightness Status (scale)		Dimmer rotary Dimmer horizontal Dimmer vertical	Switch Object Status feedback object Value object Status feedback value object
LIGHT BRIGHTNESS	Brightness Control * (scale) Brightness Status (scale)	DPT of the general object must be a scale. If the object's DPT is an unsigned integer, it is not recognizable.	Dimmer rotary Dimmer horizontal Dimmer vertical	Value object Status feedback value object
LIGHT RGB	RGB Control * (rgb) RGB Status (rgb)		RGB	RGB object RGB status object
LIGHT COLOR TEMPERATURE	On/Off Control * (bool) On/Off Status (bool) Brightness Control * (scale) Brightness Status (scale) Color Temperature Control * (uint16)		Dimmer rotary Dimmer horizontal Dimmer vertical	Switch Object Status feedback object Value object Status feedback value object

Device type	Required characteristics (* Mandatory)	Parameters	Equivalent widget (Touch)	Group objects (Touch)
	Color Temperature Status (uint16)			
THERMOSTAT	Current Heating / Cooling State Status * (uint8) Target Heating / Cooling State Control * (uint8) Target Heating / Cooling State Status (uint8) Temperature Status * (float16) Target Temperature Control * (float16) Target Temperature Status (float16)	0 = Off, 1 = Heat, 2 = Cool 0 = Off, 1 = Heat, 2 = Cool, 3 = Auto 0 = Off, 1 = Heat, 2 = Cool, 3 = Auto	Thermostat	Current Temperature Current setpoint temperature input Current setpoint temperature output
Optional	Humidity Status (scale) Target Relative Humidity Control (scale) Target Relative Humidity Status (scale) Cooling Threshold Temperature Control (float16) Cooling Threshold Temperature Status (float16) Heating Threshold Temperature Control (float16) Heating Threshold Temperature Status (float16) Temperature Display Units Control (bool) Temperature Display Units Status (bool)	0 = Celsius, 1 = Fahrenheit 0 = Celsius, 1 = Fahrenheit		
LOCK MECHANISM	Lock Target State Control * (bool) Lock Target State Status (bool) Lock Current State Status * (uint8)	0 = Unsecured, 1 = Secured 0 = Unsecured, 1 = Secured 0 = Unsecured, 1 = Secured, 2 = Jammed, 3 = Unknown	N/A	
DOOR	Current Position Status * (scale) Target Position Control * (scale) Target Position Status (scale)		Shutter rotary Shutter horizontal Shutter vertical	Status feedback for height Height position (Status feedback for height)
Optional	Position State Status (uint8) Invert value checkbox	0 = going to minimum, 1 = going to maximum, 2 = stopped Option to invert the value		
WINDOW	Current Position Status * (scale) Target Position Control * (scale)		Shutter rotary Shutter horizontal Shutter vertical	Status feedback for height Height position

Device type	Required characteristics (* Mandatory)	Parameters	Equivalent widget (Touch)	Group objects (Touch)
	Target Position Status (scale)			(Status feedback for height)
Optional	Position State Status (uint8) Invert value checkbox	0 = going to minimum, 1 = going to maximum, 2 = stopped Option to invert the value		
WINDOW COVERING	Current Position Status * (scale) Target Position Control * (scale) Target Position Status (scale)		Shutter rotary Shutter horizontal Shutter vertical	Status feedback for height Height position (Status feedback for height)
Optional	Position state status Invert value checkbox	0 = going to minimum, 1 = going to maximum, 2 = stopped Option to invert the value		
AIR PURIFIER	Active Control * (bool) Active Status (bool) Current Air Purifier State Status * (uint8) Target Air Purifier State Control * (bool) Target Air Purifier State Status (bool)	0 = Inactive, 1 = Idle, 2 = Purifying Air 0 = Manual, 1 = Auto 0 = Manual, 1 = Auto	N/A	
Optional	Lock Physical Controls Control (bool) Lock Physical Controls Status (bool) Rotation Speed Control (scale) Rotation Speed Status (scale) Swing Mode Control (bool) Swing Mode Status (bool)			
HEATER COOLER	Active Control * (bool) Active Status (bool) Current Heater / Cooler State Status * (uint8) Target Heater / Cooler State Control * (uint8) Target Heater / Cooler State Status (uint8) Temperature Status * (float16)	0 = Inactive, 1 = Idle, 2 = Heating, 3 = Cooling 0 = Auto, 1 = Heat, 2 = Cool 0 = Auto, 1 = Heat, 2 = Cool	Thermostat	Switch on/off thermostat Switch on/off thermostat status Current Temperature
Optional	Lock Physical Controls Control (bool) Lock Physical Controls Status (bool) Rotation Speed Control (scale) Rotation Speed Status (scale)	0 = Celsius, 1 = Fahrenheit 0 = Celsius, 1 = Fahrenheit		

Device type	Required characteristics (* Mandatory)	Parameters	Equivalent widget (Touch)	Group objects (Touch)
	Swing Mode Control (bool) Swing Mode Status (bool) Cooling Threshold Temperature Control (float16) Cooling Threshold Temperature Status (float16) Heating Threshold Temperature Control (float16) Heating Threshold Temperature Status (float16) Temperature Display Units Control (bool) Temperature Display Units Status (bool)			
HUMIDIFIER/ DEHUMIDIFIER	Active Control * (bool) Active Status (bool) Current Humidifier / Dehumidifier State Status * (uint8) Target Humidifier / Dehumidifier State Control * (uint8) Target Humidifier / Dehumidifier State Status (uint8) Humidity Status * (scale)	0 = Inactive, 1 = Idle, 2 = Heating, 3 = Humidifying, 4 = Dehumidifying 0 = Auto, 1 = Humidifier, 2 = Dehumidifier 0 = Auto, 1 = Humidifier, 2 = Dehumidifier	N/A	
Optional	Lock Physical Controls Control (bool) Lock Physical Controls Status (bool) Rotation Speed Control (scale) Rotation Speed Status (scale) Swing Mode Control (bool) Swing Mode Status (bool) Relative Humidity Dehumidifier Threshold Control (scale) Relative Humidity Dehumidifier Threshold Status (scale) Relative Humidity Humidifier Threshold Control (scale) Relative Humidity Humidifier Threshold Status (scale) Water Level Status (scale)			

Device type	Required characteristics (* Mandatory)	Parameters	Equivalent widget (Touch)	Group objects (Touch)
VALVE	Active Control * (bool) Active Status (bool) In Use Status * (bool)		Thermostat	Switch on/off thermostat Switch on/off thermostat status
Optional	Valve Type Status (uint8)	0 = Water faucet, 1 = Irrigation, 2 = Shower head		
AIR QUALITY SENSOR	Air Quality Status * (uint8)	0 = Unknown, 1 = Excellent, 2 = Good, 3 = Fair, 4 = Inferior, 5 = Poor	Air quality Particle sensor	
Optional	Nitrogen Dioxide Density Status (float16) Ozone Density Status (float16) PM10 Density Status (float16) PM2.5 Density Status (float16) Sulphur Dioxide Density Status (float16) VOC Density Status (float16) Status Active Status (bool) Status Fault Status (bool) Status Low Battery Status (bool) Status Tampered Status (bool)			NO_2 (Air quality) Ozone (Air quality) Part.10 (Particle sensor) Part.2.5 (Particle sensor) TVOC (Air quality)
SECURITY SYSTEM	Security System Current State Status * (uint8) Security System Target State Control * (uint8) Security System Target State Status (uint8)	0 = Stay Arm, 1 = Away Arm, 2 = Night Arm, 3 = Disarmed, 4 = Alarm Triggered 0 = Stay Arm, 1 = Away Arm, 2 = Night Arm, 3 = Disarm 0 = Stay Arm, 1 = Away Arm, 2 = Night Arm, 3 = Disarm	N/A	
Optional	Security System Alarm Type Status (bool) Status Fault Status (bool) Status Tampered Status (bool)			
GARAGE DOOR OPENER	Current Door State Status * (uint8) Target Door State Control * (bool) Target Door State Status (bool) Obstruction Detected Status * (bool)	0 = Open, 1 = Closed, 2 = Opening, 3 = Closing, 4 = Stopped 0 = Open, 1 = Closed 0 = Open, 1 = Closed	N/A	

Device type	Required characteristics (* Mandatory)	Parameters	Equivalent widget (Touch)	Group objects (Touch)
Optional	Lock Current State Status (uint8) Lock Target State Control (bool) Lock Target State Status (bool)	0 = Unsecured, 1 = Secured, 2 = Jammed, 3 = Unknown 0 = Unsecured, 1 = Secured 0 = Unsecured, 1 = Secured		