

SmartHomeBoard

Ausführliches Benutzerhandbuch

Codebasierte Dokumentation · Stand des bereitgestellten Projekts

Inhalt

- 1 Überblick
- 2 Erste Einrichtung und Verbindung
- 3 Navigation und Verbindungsstatus
- 4 Geräteliste und Gerätedetails
- 5 Dashboard, Seiten und Kacheln
- 6 SmartWidgets
- 7 Spezialkacheln
- 8 Darstellungsregeln und Vorlagen
- 9 Lokale Automationen
- 10 Zusätzliche Integrationen
- 11 Wanddisplay-Betrieb
- 12 Datensicherung und Übertragung
- 13 Diagnose und Fehlerbehebung
- 14 Bekannte Grenzen und Hinweise

1. Überblick

SmartHomeBoard ist eine native Dashboard-App für iPhone, iPad und Mac. Sie verbindet homee-kompatible Geräte mit einem frei gestaltbaren Dashboard und kann zusätzlich Geräte aus Apple Home, MotionBlinds und Velux einbinden. Die App zeigt Live-Werte, bietet Gerätesteuerung, Verlaufskurven, SmartWidgets und lokale WENN/UND/DANN-Automationen.

Wichtig: Diese Anleitung wurde aus dem bereitgestellten Swift-Projekt erstellt. Sie beschreibt nur Funktionen, die im Code vorhanden sind. Bezeichnungen können sich in späteren TestFlight-Versionen ändern.

Unterstützte Hauptbereiche

- Dashboard mit mehreren Seiten und frei platzierbaren Kacheln
- Geräteliste mit Suche, Quellenfilter und Sortierung
- Gerätedetails mit Steuerung und History-Diagrammen
- SmartWidgets für Gerätegruppen
- Kalender-, Kamera-, Energiefluss-, Alarm- und homeegram-Kacheln
- Lokale Automationen mit Zeit-, Geräte- und SmartWidget-Auslösern
- HomeKit-, MotionBlinds- und Velux-Integration
- Wanddisplay-Funktionen wie Abdunkeln und Aufwecken
- Export und Import von Dashboard und Automationen

2. Erste Einrichtung und Verbindung

2.1 Gateway einrichten

1. Öffne über das Zahnrad die Einstellungen.
2. Trage die Gateway-ID oder die lokale IP-Adresse ein.
3. Trage Nutzernamen und Passwort des Gateways ein.
4. Aktiviere „Lokale Verbindung“, wenn die App direkt im Heimnetz auf eine IP-Adresse zugreifen soll.
5. Aktiviere bei Bedarf „Automatisch verbinden“.
6. Speichere die Einstellungen beziehungsweise tippe auf Verbinden.

Die App speichert einen Access Token im Schlüsselbund. Ist dieser weiterhin gültig, wird er erneut verwendet. Wird er abgelehnt, fordert die App automatisch einen neuen Token an.

2.2 Demo-Modus

Der Demo-Modus lädt lokale Beispieldaten und benötigt kein Gateway. Er ist für App-Review, TestFlight und die Erprobung der Oberfläche vorgesehen. Beim Wechsel zurück in den normalen Betrieb werden wieder echte Verbindungen und Geräte verwendet.

2.3 Lokaler Gerätestand

Der zuletzt bekannte Gerätestand wird lokal gespeichert. Dadurch sind Geräte und Werte nach einem Neustart sofort sichtbar, bis die App aktuelle Daten mit GET:nodes vom Gateway erhält.

3. Navigation und Verbindungsstatus

Die Hauptnavigation enthält Dashboard, Geräte und Automationen. Auf kompakten Geräten wird „Automationen“ in der unteren Umschaltung verkürzt als „Regeln“ angezeigt.

Anzeige	Bedeutung
Grün	WebSocket verbunden und aktuelle Gerätedaten geladen.
Gelb, blinkend	Die App synchronisiert Geräte und wartet auf GET:nodes.
Orange	Verbindung besteht, der Gerätestand ist jedoch noch nicht als frisch markiert.
Grau	Getrennt.
Rot	Verbindungsfehler.

Nach dem Wechsel in den Hintergrund kann iOS die WebSocket-Verbindung beenden. Beim erneuten Aktivieren der App wird die Verbindung wieder aufgebaut. Auch bei einem unerwarteten Abbruch versucht die App einen Reconnect.

4. Geräteliste und Gerätedetails

4.1 Geräteliste

Die Geräteliste zeigt alle eingebundenen Nodes unabhängig von ihrer Quelle. Sie kann durchsucht, nach Quelle gefiltert und sortiert werden. Die Zeilen zeigen Gerätenamen, Integrationsquelle und wichtige aktuelle Attribute.

4.2 Hauptbedienung

Die Detailansicht wählt abhängig von Profil und Attributen passende Bedienelemente. Dazu gehören Ein/Aus-Schalter, Slider, Farbauswahl, Farbtemperatur und Rollladenbefehle. Geräte mit mehreren Instanzen, etwa DoubleSwitch oder DoubleLamp, werden getrennt behandelt.

Gerätetyp	Typische Bedienung
Lampen	Ein/Aus, Dimmen, Farbe und Farbtemperatur
Steckdosen/Schalter	Ein/Aus; mehrere Instanzen separat
Rollläden/Jalousien	Auf, Stopp, Zu und Positionswert
Thermostate	Solltemperatur und Messwerte

Sensoren	Temperatur, Feuchte, Bewegung, Kontakt, Rauch, Wasser, Helligkeit, Batterie
Sirenen	Alarmzustand und schaltbare Attribute

4.3 Zusatzattribute

Nicht zur Hauptbedienung gehörende, aber schreibbare Attribute werden als zusätzliche Werte angezeigt. Vor einer Änderung erscheint eine Warnung, weil solche Attribute Geräteeinstellungen verändern können.

4.4 Verlauf

Für geeignete numerische Attribute kann die App History-Daten laden und als Diagramm anzeigen. Verfügbar sind Tag, Woche und Monat. Das eignet sich unter anderem für Temperatur, Luftfeuchtigkeit, Leistung, Energie, Batterie, Dimmlevel und Position.

5. Dashboard, Seiten und Kacheln

5.1 Seiten

Das Dashboard kann mehrere Seiten enthalten. Jede Seite hat einen Titel und ein Symbol. Seiten können thematisch genutzt werden, etwa Start, Energie, Garten oder Rollläden. Auf dem iPhone erscheinen sie kompakt; der Seitenwechsel ist auch per Wischen möglich.

5.2 Kacheln hinzufügen

7. Tippe im Dashboard auf das Pluszeichen.
8. Wähle ein einzelnes Gerät, ein SmartWidget oder eine Spezialkachel.
9. Passe Titel und Symbol an.
10. Wähle bei Gerätekacheln die sichtbaren Attribute und deren Reihenfolge.
11. Lege bei Bedarf Icon- und Farbgeln fest.
12. Speichere die Kachel.

5.3 Gerätekachel-Vorlagen

Für einzelne Geräte stehen die Vorlagen Automatisch, Lampe, Wandschalter/Steckdose, Rollläden, Thermostat, Bewegungsmelder und Allgemeines Gerät zur Verfügung. „Automatisch“ wählt die Darstellung anhand der vorhandenen Attribute.

5.4 Größe und Raster

Die globale Widgetgröße kann auf Klein, Mittel oder Groß eingestellt werden. Einzelne Kacheln können zusätzlich eine Breite und Höhe von jeweils einer bis drei Rastereinheiten besitzen. Positionen werden im Dashboard-Raster gespeichert.

5.5 Bearbeitungsmodus

Ein langer Druck aktiviert den Bearbeitungsmodus. Kacheln können verschoben, auf andere Seiten gezogen, bearbeitet, umbenannt oder gelöscht werden. Mit „Fertig“ wird der Modus beendet.

6. SmartWidgets

SmartWidgets fassen mehrere passende Geräte zusammen. Beim Anlegen können Geräte ausdrücklich ausgewählt oder ausgeschlossen werden. Die Detailansicht des Widgets zeigt die enthaltenen Geräte und unterstützt je nach Typ eine Gruppensteuerung.

SmartWidget	Inhalt und Funktion
Lampen	Zählt aktive Lampen/Schalter; Gruppenbefehle Aus und An.
Fenster	Zeigt geschlossen, offen und gekippt.
Rollläden	Zeigt geöffnet, teilweise geöffnet und geschlossen; Gruppenbefehle Auf, Stopp, Zu.
Bewegung	Zeigt aktive und ruhige Bewegungsmelder.
Klima	Zeigt Bereiche von Temperatur und Luftfeuchtigkeit.
Energie	Summiert aktuelle Leistung in W/kW; keine kumulierten Wh/kWh-Werte.
Batterien	Fasst Batteriestände zusammen.
Rauchmelder	Zeigt Alarmstatus.
Sirenen	Zeigt aktive Sirenen; Gruppenbefehle Aus und An.
Wetter	Fasst passende Wetterattribute zusammen.
Universal	Frei zusammengestellte Gerätegruppe.

Bei Lampen mit mehreren Schaltinstanzen kann festgelegt werden, welche Instanz im SmartWidget zählt. Gruppenbefehle werden zeitversetzt versendet, um das Gateway nicht mit vielen gleichzeitigen WebSocket-Befehlen zu belasten.

7. Spezialkacheln

7.1 Kalender

Die Kalender-Kachel liest Termine aus den Systemkalendern von iOS/iPadOS. Die App benötigt dafür eine Kalenderfreigabe. Es können alle Kalender oder bestimmte Kalender ausgewählt werden. Breite und Höhe sind jeweils von eins bis drei einstellbar.

7.2 Kamera

Eine Kamera-Kachel kann einen Stream oder ein regelmäßig aktualisiertes Bild anzeigen. Unterstützte Typen sind HLS, WebRTC/go2rtc, MSE/go2rtc, MJPEG und Snapshot. Je nach Typ werden Stream-URL, Snapshot-URL und Aktualisierungsintervall verwendet.

Hinweis: Ob ein Stream auf allen Apple-Geräten funktioniert, hängt vom verwendeten Codec, Transport, Zertifikat und vom Server ab. Eine im Browser funktionierende URL ist nicht automatisch in jeder nativen Webansicht kompatibel.

7.3 Energiefluss

Die Energiefluss-Kachel ordnet einzelne Attribute Rollen zu: Netz, AC-PV, DC-PV, Batterieleistung, Batteriestand, Batteriemodus und Wechselrichter. Für jede Zuordnung kann das Vorzeichen umgekehrt werden. So lassen sich unterschiedliche Messrichtungen vereinheitlichen.

7.4 Alarm

Eine Alarm-Kachel nutzt ein ausgewähltes Attribut mit frei festlegbarem Scharf- und Unscharf-Wert. Die Logik kann invertiert und ein Entschärfcode hinterlegt werden. Sind Alarm und Sirenen-SmartWidget aktiv, kann ein hervorgehobener Alarmdialog erscheinen.

7.5 homeegram

Eine homeegram-Kachel verweist auf ein vorhandenes homeegram und kann es vom Dashboard aus ausführen. Die App lädt homeegrams zusätzlich zu den Geräten vom Gateway.

8. Darstellungsregeln und Vorlagen

Kacheln können abhängig von Attributwerten ihr Symbol, ihre Farbe und ihre Animation ändern. Eine Regel kann einen exakten Wert oder einen Von-bis-Bereich prüfen. Als Effekte stehen Keine, Puls, Glow, Ring und Glow + Ring zur Verfügung. Das Ring-Attribut kann getrennt vom Auslöseattribut gewählt werden.

- Beispiel Fenster: geschlossen grau, offen farbig, gekippt mit anderem Symbol.
- Beispiel Batterie: 0-20 rot, 21-60 gelb, 61-100 grün.
- Beispiel Bewässerung: bei Aktivität blau und blinkend.

Darstellungsregeln können als lokale Vorlage gespeichert und auf andere Kacheln angewendet werden. Titel, eigene Icons und Regeln bleiben im Dashboard-Export erhalten.

9. Lokale Automationen

9.1 Grundprinzip

Eine Automation besteht aus mindestens einem AUSLÖSER, optionalen UND-BEDINGUNGEN und mindestens einer DANN-AKTION. Mehrere Auslöser wirken alternativ: Einer genügt. Alle Bedingungen müssen gelten.

9.2 Auslöser und Bedingungen

Typ	Beschreibung
Gerätewert	Vergleicht ein Attribut mit gleich, ungleich, größer oder kleiner.
Gerätewert ändert sich um	Reagiert auf eine absolute oder prozentuale Änderung.
SmartWidget-Zustand	Prüft, ob mindestens eines, keines oder alle Einträge einen Zustand erfüllen.
SmartWidget-Wert ändert sich	Reagiert auf die Änderung eines zusammengefassten Widget-Wertes.
Täglich um	Startet täglich zu einer Uhrzeit.
Einmalig am	Startet einmalig zu einem Datum und einer Uhrzeit.
Uhrzeit nach/vor	Dient als zeitliche Bedingung.

Bedingungen können beim Auslösen, erst beim Ausführen oder zu beiden Zeitpunkten geprüft werden. Das ist besonders wichtig, wenn Aktionen eine Verzögerung besitzen.

9.3 Aktionen

Aktion	Funktion
Gerät steuern	Schreibt einen Wert in ein Geräteattribut.
Dashboard-Seite öffnen	Wechselt automatisch auf eine ausgewählte Seite.
Pop-up anzeigen	Zeigt Titel und Nachricht in der App.
Ton abspielen	Spielt Hinweis, Türklingel oder Alarm ab.
Displayhelligkeit ändern	Setzt die Helligkeit des Displays.
Widget ein-/ausblenden	Ändert die Sichtbarkeit einer Dashboard-Kachel.

Für jede Aktion kann eine Verzögerung festgelegt werden. Während eines laufenden Zeitablaufs zeigt die Regel einen Countdown und kann über den Stopp-Button abgebrochen werden. Regeln

können mit dem Play-Button manuell getestet werden. Eine Abkühlzeit verhindert zu häufiges erneutes Auslösen.

Wesentliche Grenze: Die Automation-Engine läuft lokal und nur solange SmartHomeBoard aktiv ist. Wird die App von iPadOS beendet oder vollständig inaktiv, verarbeitet sie keine Ereignisse. Diese Regeln ersetzen deshalb keine dauerhaft laufende Serverautomation.

10. Zusätzliche Integrationen

10.1 Apple Home / HomeKit

Nach Aktivierung lädt die App Häuser, Zubehör, Dienste und unterstützte Eigenschaften über HomeKit. Beim ersten Zugriff verlangt iOS die Freigabe für Apple Home. Unterstützte Eigenschaften werden in homeee-kompatible Attribute übersetzt, unter anderem Ein/Aus, Helligkeit, Temperatur, Solltemperatur, Luftfeuchtigkeit, Batterie, Helligkeitssensor, Kontakt, Rauch, Wasser und Bewegung. Fensterabdeckungen werden als Rollladenprofil abgebildet. Nicht jeder HomeKit-Dienst und nicht jede herstellerspezifische Eigenschaft wird unterstützt.

10.2 MotionBlinds

MotionBlinds wird lokal über die Bridge eingebunden. Benötigt werden Bridge-IP, Secret Key und ein Antwort-Port. Die App sucht Geräte, bildet sie als normale Rollladen-Nodes ab und unterstützt Position, Fahrbefehle und Batterie. Erkannte Geräte können in der App eigene Namen erhalten.

10.3 Velux

Velux wird über die Velux-Cloud eingebunden. Benötigt werden E-Mail und Passwort. Nach der Anmeldung sucht die App Zuhause, Bridge und Geräte. Positionen werden regelmäßig in einem einstellbaren Intervall aktualisiert; Fahrbefehle werden an die Cloud gesendet. Für erkannte Geräte können eigene Namen gespeichert werden.

Datenschutz: Zugangsdaten für externe Dienste sollten nur auf vertrauenswürdigen Geräten verwendet werden. Velux benötigt eine Cloud-Verbindung; MotionBlinds arbeitet im lokalen Netzwerk; HomeKit folgt den Apple-Home-Berechtigungen.

11. Wanddisplay-Betrieb

Für ein dauerhaft montiertes iPad kann die App das Display nach einer einstellbaren Zeit abdunkeln. Die reduzierte Helligkeit ist prozentual wählbar. Ein Antippen stellt die vorherige Helligkeit wieder her. Zusätzlich kann ein Geräteattribut als Wake-Attribut festgelegt werden; meldet es den konfigurierten Wert, wird das Display aufgeweckt.

Für einen zuverlässigen Dauerbetrieb sollten Auto-Lock, Stromversorgung, geführter Zugriff und die iPadOS-Energieeinstellungen passend konfiguriert werden. Die App kann das Betriebssystem jedoch nicht daran hindern, sie bei Speicherdruck oder längerer Inaktivität zu beenden.

12. Datensicherung und Übertragung

Unter „Dashboard übertragen“ kann eine JSON-Datei exportiert und auf einem anderen Gerät importiert werden. Der Export enthält Seiten, Dashboard-Kacheln, SmartWidgets, Titel, Symbole, Darstellungsregeln und Automationen. Zugangsdaten und Tokens sind nicht Bestandteil des Exports. Der Import ersetzt das aktuelle Dashboard und übernimmt enthaltene Automationen.

13. Diagnose und Fehlerbehebung

13.1 Diagnoseprotokoll

Das Diagnoseprotokoll enthält App-Start, Token-Verwendung, Verbindungsaufbau, WebSocket-Status, GET:nodes, geladene Geräte und Fehler. Es kann kopiert oder geleert werden. Reine WebSocket-Nutzdaten werden nicht dauerhaft protokolliert.

13.2 Häufige Probleme

Problem	Prüfung
Keine Geräte	Verbindungsstatus, lokale IP/Gateway-ID, Nutzernamen, Passwort und Diagnose prüfen.
Gelber Punkt bleibt	Die App wartet auf GET:nodes; Gateway-Verbindung und Netzwerk prüfen.
HomeKit leer	Apple-Home-Freigabe und vorhandenes Zuhause prüfen.
MotionBlinds nicht gefunden	Bridge-IP, Secret Key, Antwort-Port und lokales Netzwerk prüfen.
Velux-Anmeldung scheitert	E-Mail, Passwort, Internet und Cloud-Status prüfen.
Automation reagiert nicht	Regel aktivieren, App im Vordergrund/aktiv halten, Auslöser und Bedingungen testen.
Kalender leer	Kalenderfreigabe und ausgewählte Kalender prüfen.
Kamerastream leer	URL, Codec, HTTPS/Zertifikat und Streamtyp prüfen.

14. Bekannte Grenzen und Hinweise

- Lokale Automationen laufen nicht zuverlässig weiter, wenn iPadOS die App beendet.
- HomeKit-Unterstützung umfasst nur im Adapter abgebildete Dienste und Eigenschaften.
- Velux benötigt die Hersteller-Cloud und Internetzugang.
- Kamera-Kompatibilität hängt vom jeweiligen Stream und Apple-Framework ab.
- Dashboard-Konfiguration und benutzerdefinierte Namen werden lokal gespeichert; sie sind ohne Export nicht automatisch auf anderen Geräten vorhanden.
- Beim Import wird das vorhandene Dashboard ersetzt. Vorher sollte ein Export erstellt werden.