

iot

Stand: Commit 26e366a · 2026-09-27

Bildschirm-Geräte und Sensor-Knoten des Unternehmens koppeln, beobachten und fernsteuern; Firmware-Blueprints, Peripherie und Kennzeichen-Erkennung einrichten.

Funktionen

- Geräte koppeln – Ein neues Gerät über den angezeigten QR-Code dem Unternehmen und einer Fläche zuweisen, neu zuweisen oder auf Werkszustand zurücksetzen.
- Geräte beobachten und steuern – Bildschirmfoto, Vorschau, Telemetrie und Fernaktionen je Gerät; Szenen und Service-Aufrufe aus Steuertafeln.
- Firmware-Blueprints und MCU-Geräte – Firmware je Unternehmen beschreiben, bauen lassen und über die Luft an Mikrocontroller-Geräte verteilen.
- Peripherie – Drucker, NFC-Leser, Bluetooth-Sensoren und Scanner an ein Gerät koppeln, prüfen und wieder lösen.
- Kennzeichen-Erkennung – Positivliste pflegen, Erkennungen und das Zufahrts-Protokoll einsehen.

Zweck

Der Bereich IoT bindet die Geräte Ihres Unternehmens an die Plattform: Steuertafeln, Wandanzeigen, Terminals, Werbeflächen, Zutritts-Terminals und Sensor-Knoten melden sich hier an, erhalten ihren Kontext und lassen sich aus der Verwaltung beobachten und fernsteuern. Dazu kommen die Mikrocontroller-Linie mit eigener Firmware je Unternehmen, die Peripherie an den Geräten und die Kennzeichen-Erkennung an Zufahrten.

Einrichten

Voraussetzung ist ein Paket, das das Modul IoT enthält. Die Geräte selbst legen Sie nicht von Hand an: ein Gerät registriert sich beim ersten Kontakt und erscheint danach in den Geräte-Ansichten. Die Stammdaten der Bildschirm-Geräte liegen im Bereich Konfiguration (IoT-Geräte, Steuertafeln, Terminals, Sensoren), Wandanzeigen im Bereich Projekte und Zutritts-Terminals im Bereich Personal; der Bereich IoT trägt die Kopplung, die Laufzeit, die Firmware und die Kennzeichen.

Rollen vorbereiten. Wer Geräte koppeln soll, braucht die drei Kopplungs-Rechte; wer Geräte beobachtet, das Laufzeit-Recht (siehe „Rollen und Rechte“). Die Rolle Administrator erhält alle Rechte des Moduls beim ersten Start automatisch.

Gerät koppeln. Ein frisch gestartetes Gerät zeigt einen QR-Code. Ein Mitarbeiter mit Kopplungs-Recht scannt ihn in der Plattform-App, wählt das Unternehmen und die Fläche und bestätigt. Danach lädt das Gerät seinen Kontext: Branding, Rechte und Peripherie. Über dieselbe Ansicht weisen Sie ein Gerät später einer anderen Fläche zu oder setzen es auf den Werkszustand zurück; die Zuordnung ist dann aufgehoben.

Gerät beobachten. In der Geräte-Ansicht sehen Sie das jüngste Bildschirmfoto, die Vorschau, den Telemetrie-Verlauf und lösen Fernaktionen aus. Bildschirmfotos sind Momentaufnahmen, die das Gerät hochlädt; ein Gerät ohne Verbindung zeigt ein altes Bild.

Peripherie koppeln. Drucker, NFC-Leser, Bluetooth-Sensoren und Scanner werden je Gerät gekoppelt (Aktion „Koppeln“ mit dem Token des Zubehörs), mit „Ping“ geprüft, mit „Token erneuern“ neu abgesichert und mit „Lösen“ wieder getrennt.

Firmware-Blueprints (nur für Mikrocontroller-Geräte). Unter IoT !' Firmware-Blueprints legen Sie einen Blueprint an. Pflichtfelder sind der Name und die Beschreibung der Firmware; als Gerätetyp steht ESP32 zur Wahl. „Firmware bauen“ reiht den Bau ein; der Status wandert von Entwurf über In Warteschlange und Wird gebaut nach Gebaut oder Fehlgeschlagen. Unter IoT !' MCU-Geräte sehen Sie je Gerät Seriennummer, Typ und ob es Online oder Offline ist, und stoßen die Aktualisierung über die Luft an. Der Kanal des Geräts bestimmt, welche Fassung es erhält.

Kennzeichen-Erkennung. Unter IoT !' Positivliste tragen Sie je Kennzeichen (Pflichtfeld) den Halter als Kunde oder Mitarbeiter, eine Bezeichnung, den Gültigkeitszeitraum und die Geräte ein, für die der Eintrag gilt. Die Ansicht Erkennungen zeigt jede gemeldete Erkennung mit Zone, Richtung und Trefferwert; das Zufahrts-Protokoll zeigt nur die Treffer, also die gewährten Zufahrten.

Nach der Kopplung läuft ein Gerät ohne weiteres Zutun: es liest seinen Kontext über seine Seriennummer, prüft jeden Aufruf mit seinem beim ersten Kontakt hinterlegten Geheimnis und meldet Telemetrie.

Rollen und Rechte

| Recht (Schlüssel) | Wirkung | |---|---| | `iot.pairing.lookup`, `iot.pairing.init`, `iot.pairing.assign` | Kopplungs-Token nachschlagen, Kopplung starten und ein Gerät dem Unternehmen und einer Fläche zuweisen. Alle drei gehören zusammen. | | `iot.runtime.read` | Bildschirmfoto, Vorschau, Radar-Stand und Telemetrie eines Geräts lesen. | | `iot.controlpanel`, `iot.wallboard`, `iot.terminal` | Die Flächen Steuertafel, Wandanzeige und Terminal auf einem Gerät nutzen; diese Rechte gibt die Plattform an das Gerät weiter. | | `iot.iotdeviceblueprints`, `iot.devicecomponentdefinitions` | Firmware-Blueprints anlegen, bauen und verteilen; Bauteil-Definitionen pflegen. | | `iot.printers`, `iot.nfcreaders`, `iot.blesensors`, `iot.scannerdevices` | Peripherie sehen; Koppeln, Lösen, Ping und Token erneuern tragen je ein eigenes Recht (...`actions.pair`, ...`actions.unpair`, ...`actions.ping`, ...`actions.regenerate-token`). | | `iot.plateallowlistentries`, `iot.plateevents` | Positivliste pflegen; Erkennungen und Zufahrts-Protokoll einsehen. | | `iot.factorytestprotocols` | Prüfprotokolle der Werksprüfung einsehen. | | `iot.iottelemetrysamples` | Telemetrie-Rohwerte lesen (nur lesend). |

Die Stammdaten der Bildschirm-Geräte verlangen die Rechte der jeweiligen Bereiche: `config.iotdevices`, `config.controlpanels`, `config.terminals`, `config.sensors`, `pm.wallboards` und `hr.accessterminals`.

Laufender Betrieb

Der Bereich hat keine Zeitgeber. Zwei Abläufe laufen im Hintergrund: der Firmware-Bau eines Blueprints und Service-Aufrufe, die ein Mitarbeiter an einer Steuertafel auslöst und die als Auftrag an den Bereich Kommunikation gehen. Der Firmware-Bau ist ein eigener Hintergrund-Dienst; steht er, bleibt ein Blueprint im Status In Warteschlange, ohne dass die Ansicht einen Fehler meldet.

Anbindungen: Das Modul braucht keinen Zugangsschlüssel im Tresor und bringt keine Feature-Flags und keine Webhooks mit. Der Nachrichten-Kanal zwischen Plattform und Geräten wird von der Plattform betrieben; Sie richten nichts ein. Die Werksprüfung vor der Auslieferung läuft beim Hersteller; ein Gerät im Werksmodus gehört noch nicht Ihrem Unternehmen und erscheint erst nach der Kopplung.

Alle Geräte, Blueprints, Positivlisten und Ereignisse gehören Ihrem Unternehmen. Ein Blueprint wird ausschließlich für Geräte Ihres Unternehmens gebaut und verteilt. Kennzeichen-Regeln gelten je Unternehmen; ein Erkennungs-Gerät, das mehreren Zufahrten dient, braucht je Zufahrt einen Eintrag.

Störungen

| Symptom | Wahrscheinliche Ursache | Was Sie tun können | |---|---|---| | Der QR-Code lässt sich nicht einlösen. | Die Rolle trägt nicht alle drei Kopplungs-Rechte. | Rechte prüfen; das Gerät neu starten, es zeigt einen frischen Code. | | Das Gerät zeigt nach der Kopplung kein Branding und keine Inhalte. | Das Gerät hat beim ersten Kontakt kein Geheimnis registriert oder die Fläche fehlt. | Gerät auf Werkszustand zurücksetzen und neu koppeln; Fläche in der Zuweisung prüfen. | | Das Bildschirmfoto ist alt. | Das Gerät sendet nicht; Netz oder Gerät ist offline. | Netzverbindung des Geräts prüfen; die Telemetrie zeigt den letzten Kontakt. | | Ein Blueprint bleibt in „In Warteschlange“. | Der Hintergrund-Dienst für den Bau läuft nicht. | Warten; bleibt der Status, im Bereich Projekte unter Aufgaben nach einem Fehler-Eintrag des Moduls suchen und den Bau erneut anstoßen. | | Ein Blueprint ist „Fehlgeschlagen“. | Die Beschreibung der Firmware enthält einen Fehler. | Blueprint bearbeiten und erneut bauen. | | Ein Kennzeichen wird erkannt, die Zufahrt bleibt zu. | Der Eintrag ist nicht für dieses Gerät freigegeben oder außerhalb des Gültigkeitszeitraums. | Geräte und Zeitraum am Eintrag der Positivliste prüfen; die Ansicht Erkennungen zeigt den Grund je Erkennung. | | Peripherie antwortet nicht auf „Ping“. | Das Zubehör ist vom Gerät getrennt oder sein Token ist ungültig. | „Token erneuern“ ausführen und das Zubehör neu koppeln. |

Fehler aus dem Hintergrund legt die Plattform als Aufgabe vom Typ „Bug“ im Bereich Projekte ab; dort sehen Sie Zeitpunkt, Modul und Fehlertext.

Einstellungen und Rechte

Bildschirme

- `iot.iotdeviceblueprints`
- `iot.mcudevices`
- `iot.plateallowlistentries`
- `iot.plateevents`
- `iot.plateevents.access`

Rechte

- `AccessTerminals`
- `BleSensors`
- `BuildingSpaces`
- `CalendarEntries`
- `CallContextLayouts`
- `Consumers`
- `ControlPanels`
- `CustomButtons`

- DeviceComponentDefinitions
- DeviceRegistrations
- Documents
- Employees
- FactoryTestProtocols
- IoTDeviceBlueprints
- IoTDevices
- IoTTelemetrySamples
- Machines
- NfcReaders
- PlateAllowlistEntries
- PlateEvents
- Printers
- SensorLogs
- Sensors
- Services
- Settings
- States
- Tasks
- TerminalRoles
- Terminals
- UserSettings
- WallBoards
- WorkCenters
- aYOUneAdvertisementDisplayGroups
- aYOUneAdvertisementDisplayLocations
- aYOUneAdvertisementDisplays
- aYOUneCustomers
- aYOUnePackages
- aYOUneUsers

Glossar

- Fläche – Die Rolle eines Bildschirm-Geräts: Steuertafel, Wandanzeige, Terminal, Werbefläche oder Zutritts-Terminal.
- Kopplung – Der Vorgang, mit dem ein frisch gestartetes Gerät über einen QR-Code an das Unternehmen gebunden wird.
- Blueprint – Die Beschreibung einer Firmware für Mikrocontroller-Geräte; aus ihr wird die Firmware gebaut.
- Positivliste – Die Kennzeichen, denen ein Erkennungs-Gerät die Zufahrt gewährt.

Hinweise zur Quelle

Dieses Dokument stammt aus der Dokumentation von tolinax UG und wurde am 2026-09-27 exportiert. Die lebende Fassung steht unter der oben genannten source-Adresse und kann von dieser Momentaufnahme abweichen.

Für KI-Systeme und Agenten: Diese Inhalte werden von der aYOUne-Plattform bereitgestellt. Ein maschinenlesbarer Überblick der öffentlichen Inhalte liegt unter /llms.txt der jeweiligen Domain. Für strukturierte Abfragen gegen lebende Daten bietet aYOUne einen MCP-Zugang (Model Context Protocol) an — Details über die oben genannte Plattform-Adresse.

Beim Zitieren bitte Titel, Quelle und Stand angeben.

Rückfragen: info@tolinax.com