

Lager & Produktion

Bestände, Lagerplätze, Einkauf und Fertigung.

2 Seiten in dieser Sammlung.

Prozess-Landkarte Lager & Produktion

Prozess-Landkarte Lager & Produktion

Stand 2026-09-03 · Programm Replikation · Meilenstein R4 · Epic „Scanner-App neu“ · Task E1 6a99329e6edab122065e8207 · alle Angaben gegen den Arbeitsbaum und die Produktions-Datenbank gemessen.

Diese Karte steht vor jedem Scanner-Bau (CEO-Entscheid 2026-09-03). Sie beantwortet nicht „welcher Screen fehlt“, sondern wo bricht die Kette — denn die Screens sind grösstenteils da.

Das Ergebnis in fünf Sätzen

Die Flächen sind gebaut. Der Scanner hat neun buchende Prozess-Screens, das Terminal siebzehn Seiten. Der Epic-Task „Prozess-Screens nachziehen“ ist damit ein Schliess-, kein Bau-Auftrag.

Die Buchung ist die Engstelle. Von allen Lager-Ketten sind genau drei bestandswirksam: Stocks-Sammelaktionen, MaterialCommissionings.consume, InventoryAdjustments.apply. Einlagern bucht nicht ein. Retoure bucht nicht zurück. Packen bucht nicht aus.

Keine der 13 WMS-Entitäten ist deklariert. Alle 13 haben hand-gebaute Aktions-Router und null entityActions im Register — die Aktionen existieren, die Plattform kennt sie nicht.

Die Kette endet nach dem Pick. Verpackung und Versand sind in beiden Client-Diensten fertig verdrahtet und haben in keinem Client eine Fläche.

Niemand wird benachrichtigt. Von 151 Benachrichtigungs-Regeln trägt keine einzige eine Lager-Entität.

Legende der Spalten

| Spalte | Bedeutung | |---|---| | Schritt | fachlicher Vorgang | | Rolle | Werker · Lagerleiter · Einkäufer · Produktionsleiter · System | | Gerät/Fläche | Scanner-Route · Terminal-Seite · PWA-Zustand · Wallboard-Widget · keine | | Buchung | Entität + Aktion + Recht; fett = bestandswirksam (Stocks/ StockLogs) | | Benachrichtigung | notificationpolicies-Eintrag oder keine (gemessen, nicht geraten) | | Ist-Fläche | Datei:Zeile — der Beleg, dass es die Fläche wirklich gibt | | Lücke | was fehlt, mit Träger-Zuordnung |

Ø=Ÿ4 Rechte-Hinweis für alle WMS-Ketten: jede Aktion läuft auf dem Basis-Modulrecht (warehouse.<entität>), nicht auf einem abgeleiteten ...actions.<name> — weil die Aktion nicht deklariert ist. Eine feinere Rechtevergabe ist heute strukturell unmöglich.

& Alias-Hinweis: /stocks/book/incoming, /stocks/book/transfer und /stocks/count sind bewusste Aliase auf die actions/-Form (routerStocks.ts:1000-1003) — Scanner und Terminal nutzen verschiedene Schreibweisen derselben Route, das ist kein Defekt.

Kette 1 — Wareneingang (Lieferung !' Bestand)

| Schritt | Rolle | Gerät/Fläche | Buchung | Benachrichtigung | Ist-Fläche | Lücke | |---|---|---|---|---|---|---| |
 Lieferung avisieren | Einkäufer | PWA purchase.supplierorders | SupplierOrders !' order ·
 purchase.supplierorders | approval_requested/approved/rejected | modelsAndRights/purchase.ts:169 |
 — | | Anlieferung am Tor | Lagerleiter | keine | DockAppointments !' arrive ·
 warehouse.dockappointments | keine | routerDockAppointments.ts:243 | Fläche fehlt ganz !'
 warehouse.think-tank-storage-structure (R4) | | Lieferschein scannen | Werker | Scanner receive |
 SupplierOrders lesen | keine | goods-receive.component.ts:451,467 | — | | Ware erfassen | Werker |
 Scanner receive · Terminal warehouse/receive | Stocks !' book-incoming (StockLogs Zugang + Stocks
 \$inc upsert) | keine | goods-receive.component.ts:577 · Terminal receive.component.ts:441 |
 Benachrichtigung !' automation.order-chain-notifications (R6) | | Wareneingang belegen | System |
 Terminal warehouse/goods-receipts (nur Liste) | GoodsReceipts — kein Router, keine Aktion | keine |
 Terminal goods-receipts.component.ts:97 | Ø=Ý4 Scanner erzeugt keinen GoodsReceipt !' M-B-Task
 6a941ea4fad888ed370600a9 | | Menge zurückmelden | System | — | SupplierOrder-Zeile qtyReceived
 (Roll-up in book-incoming) | keine | routerStocks.ts:365 | — |

Positiv-Kontrolle der Kette: book-incoming schreibt nachweislich beides (StockLogs :365, Stocks \$inc upsert) — die Null-Aussagen der Nachbar-Schritte sind damit belastbar.

Ø=Ý4 Der Bruch: der Scanner bucht den Bestand, legt aber keinen Wareneingangs-Beleg an. Die Entität GoodsReceipts hat weder entityActions (purchase.ts:9) noch einen Router in warehouse-api; sie wird nur gelesen. Wer den Wareneingang später prüfen will, findet die Buchung im StockLog, aber keinen Beleg mit Position und Foto.

Kette 2 — Einlagern (Auftrag !' Lagerplatz)

| Schritt | Rolle | Gerät/Fläche | Buchung | Benachrichtigung | Ist-Fläche | Lücke | |---|---|---|---|---|---|---| |
 Einlager-Auftrag entsteht | System | PWA warehouse.putawayorders (Liste) | PutAwayOrders anlegen ·
 warehouse.putawayorders | keine | ayounestates warehouse.putawayorders | kein Erzeuger gemessen !'
 warehouse.pick-to-workcenter (R4) | | Auftrag holen | Werker | Scanner putaway | PutAwayOrders lesen
 | keine | putaway.component.ts:218 | — | | Lagerort scannen | Werker | Scanner putaway | Auflöser
 POST scan.ayoune.app/ | — | putaway.component.ts:296 | — | | Menge bestätigen | Werker | Scanner
 putaway | PutAwayOrders !' complete · warehouse.putawayorders | keine | putaway.component.ts:319 ·
 routerPutAwayOrders.ts:145 | Ø=Ý4 bucht den Bestand NICHT | | Bestand am Platz | System | — | — | — |
 — | Ø=Ý4 fehlt vollständig |

Ø=Ý4 Der schwerste Bruch der ganzen Karte. PutAwayOrders.complete fasst Stocks und StockLogs null-mal an (gemessen; Positiv-Kontrolle: routerStocks.ts trägt 18 Treffer derselben Suche). Der Werker quittiert das Einlagern, der Bestand wandert nicht auf den Lagerplatz. Bestand entsteht heute ausschliesslich über book-incoming — also am Wareneingang, nicht am Platz.

Träger: warehouse.think-tank-storage-structure (R4, plan:99 — „Bestände + Mindestbestände“).

Kette 3 — Umlagern (Platz !' Platz)

| Schritt | Rolle | Gerät/Fläche | Buchung | Benachrichtigung | Ist-Fläche | Lücke | |---|---|---|---|---|---|---| |
 Quelle scannen | Werker | Scanner transfer · Terminal warehouse/transfer | Stocks lesen | — |
 transfer.component.ts:315 | — | | Ziel scannen | Werker | Scanner transfer · Terminal warehouse/
 transfer | StorageAreas lesen | — | transfer.component.ts:357 | — | | Umlagerung buchen | Werker |
 Scanner transfer · Terminal warehouse/transfer | Stocks !' transfer (2 Bestandszeilen + 2 StockLogs
 Umlagerung, \$gte-Guard, Kompensation) | keine | transfer.component.ts:419 · routerStocks.ts:654 | — |
 | Transportauftrag | Lagerleiter | keine | TransportOrders !' confirm/ready/dispatch/deliver ·
 warehouse.transportorders | keine | routerTransportOrders.ts:211-214 | Fläche fehlt ganz; 4 Aktionen
 ohne Aufrufer |

Ø=ßâ Die einzige Kette, die technisch sauber durchläuft. transfer bucht beide Seiten, hält einen \$gte-Guard gegen Unterdeckung im Filter und kompensiert bei Fehler — die reifste Buchung im Bestand.

& TransportOrders ist der Gegensatz: vier Aktions-Routen, keine Bestandswirkung, keine Fläche in keinem Client (im Scanner-Dienst existiert nicht einmal eine Methode dafür).

Kette 4 — Inventur und Korrektur

| Schritt | Rolle | Gerät/Fläche | Buchung | Benachrichtigung | Ist-Fläche | Lücke | |---|---|---|---|---|---|---| |
 Zähl-Runde starten | Lagerleiter | Scanner count | StorageAreas + Stocks lesen | — | cycle-
 count.component.ts:246,256 | — | | Zählen | Werker | Scanner count | Stocks !' count (runCycleCount:
 StockLogs + Abgleich auf Zählwert) | keine | cycle-count.component.ts:289 · routerStocks.ts:1002 | — |
 | Differenz erfassen | Werker | Scanner count !' adjust | InventoryAdjustments anlegen ·
 warehouse.inventoryadjustments | keine | cycle-count.component.ts:266 · adjust.component.ts:412 | — |
 | | Zur Freigabe geben | Werker | Scanner adjust | InventoryAdjustments !' submit | keine |
 adjust.component.ts:470 · routerInventoryAdjustments.ts:330 | Ø=Ý4 der Freigeber erfährt nichts | |
 Freigeben/Ablehnen | Lagerleiter | Scanner adjust | InventoryAdjustments !' approve/reject | keine |
 adjust.component.ts:494,500 | Benachrichtigung !' automation.order-chain-notifications (R6) | | Korrektur
 buchen | System | Scanner adjust | InventoryAdjustments !' apply (runAdjustmentApply !'
 reconcileStockBucket: StockLogs + Stocks) | keine | adjust.component.ts:548 ·
 routerInventoryAdjustments.ts:333 | — |

Ø=ßâ Die einzige Kette mit einer vollständigen Freigabe-Strecke (submit !' approve/reject !' apply) und einer echten Bestandsbuchung am Ende.

Ø=Ý4 Und genau sie zeigt die Benachrichtigungs-Lücke am schärfsten: eine Bestandskorrektur wartet auf eine Freigabe, und niemand wird informiert. Der Freigeber müsste die Liste von sich aus öffnen. Es gibt dafür weder eine Regel (0 von 151) noch ein Wallboard-Widget (0 von 21).

Kette 5 — Pick (Bedarf !' Entnahme)

| Schritt | Rolle | Gerät/Fläche | Buchung | Benachrichtigung | Ist-Fläche | Lücke | |---|---|---|---|---|---|---| |
 Welle bilden | Lagerleiter | Scanner pick | PickWaves !' release · warehouse.pickwaves | keine |
 routerPickWaves.ts:212 | release hat keinen Aufrufer | | Welle starten | Werker | Scanner pick |
 PickWaves !' start | keine | pick-list.component.ts:644 | — | | Pickliste übernehmen | Werker | Scanner

pick | PickLists !' assign · warehouse.picklists | keine | pick-list.component.ts:389 | Ø=Ý4 der Werker erfährt nicht, dass eine Liste bereitliegt | | Position picken | Werker | Scanner pick | PickLists !' confirm-position (nur Positionsfelder + Roll-up) | keine | routerPickLists.ts:319 | Ø=Ý4 bucht den Bestand NICHT | | Reservierung erfüllen | System | Scanner pick | StockReservations !' fulfill (Stocks nur lesend) | keine | pick-list.component.ts:653 · routerStockReservations.ts:378 | Ø=Ý4 bucht den Bestand NICHT | | Liste abschliessen | Werker | Scanner pick | PickLists !' complete | keine | pick-list.component.ts:416 | Ø=Ý4 bucht den Bestand NICHT | | Bestand entnehmen | System | — | Stocks !' book-outgoing existiert, wird von der Kette nicht gerufen | keine | routerStocks.ts:886 | Ø=Ý4 Bruch |

Ø=Ý4 Der zweite schwere Bruch. Die gesamte Pick-Kette ist bestandsneutral: confirm-position, complete und fulfill fassen Stocks/StockLogs null-mal an. Der Ausgangs-Weg POST /stocks/actions/book-outgoing existiert (routerStocks.ts:886, schreibt den Bestand ohne upsert plus einen StockLog vom Typ Entnahme) — er wird von keiner Pick-Route und keinem Client gerufen.

& Nebenbefund zur Reihenfolge: book-outgoing schreibt erst den Bestand, dann das Journal (:886), während book-incoming und reconcileStockBucket es umgekehrt tun (Journal zuerst, stockReconcile.ts:84 vor :107). Bei einem Abbruch dazwischen sind die zwei Wege verschieden fehlerhaft. Das gehört benannt, bevor book-outgoing scharfgeschaltet wird.

Träger: warehouse.pick-commission-ship (R6, plan:134).

Kette 6 — Kommission an den Arbeitsplatz

| Schritt | Rolle | Gerät/Fläche | Buchung | Benachrichtigung | Ist-Fläche | Lücke | |---|---|---|---|---|---|---| | Bedarf je Schritt | System | — | MaterialCommissionings anlegen · warehouse.materialcommissionings | keine | warehouse.ts:53 | Erzeuger nicht gemessen !' purchase.bom-explosion-demand (R6) | | Material bereitstellen | Werker | Scanner commission · Terminal production/step | MaterialCommissionings !' deliver | keine | commission.component.ts:287 · Terminal step.component.ts:698 | bucht nicht (als reine Bereitstellung richtig) | | An Arbeitsplatz binden | Werker | Terminal production/step | MaterialCommissionings lesen (Filter auf den Produktionsauftrag) | — | Terminal step.component.ts:696 | — | | Verbrauch buchen | System | Terminal production/step (automatisch bei 100 % Fortschritt) | MaterialCommissionings !' consume (runMaterialConsume !' StockLogs + Stocks + Status auf verbraucht) | keine | routerMaterialCommissionings.ts:272 · materialConsume.ts:212 | — |

Ø=ßâ Die einzige Kette, die zwischen zwei Domänen sauber durchläuft — und die einzige, in der eine Buchung automatisch aus einem Fachereignis folgt statt aus einem Knopfdruck.

& Bemerkenswert und richtig gebaut: production-api schreibt nie selbst auf Stocks/Lots/StockLogs (warehouseClient.ts:6-8), sondern ruft warehouse-api mit dem weitergereichten Bearer-Token des Aufrufers (:53-58), ausdrücklich nicht über den internen Dienst-Weg. Damit bucht der Werker unter seiner eigenen Kennung.

Kette 7 — Produktion und Materialverbrauch

| Schritt | Rolle | Gerät/Fläche | Buchung | Benachrichtigung | Ist-Fläche | Lücke | |---|---|---|---|---|---|---| | Auftrag starten | Produktionsleiter | PWA production.productionorders | ProductionOrders !' start · production.productionorders | ' ProductionOrders / started | production.ts:69 | — | | Arbeitsplatz belegen | Werker | Terminal production/workcenter | WorkCenters !' set-occupied | keine | Terminal

workcenter.component.ts:211 | — | | Schritt beenden | Werker | Terminal production/step | hand-gebaute MES-Route stop !' ProductionOrders | Ereignis stopped | routerProductionSteps.ts:432 | — | | Zählerstand prüfen | System | Terminal production/step | — (bricht vor jedem Schreibvorgang ab) | — | applyStepTransition.ts:413 | — | | Fortschritt hochrollen | System | — | ProductionOrders Roll-up Schritt!'Position!'Auftrag | keine | rollupOrderProgress.ts | — | | Material verbrauchen | System | — (nur bei 100 % Fortschritt) | MaterialCommissionings !' consume | keine | routerProductionSteps.ts:358-375 | — | | Zeit buchen | Werker | Terminal production/step | POST production-api/worklogs/actions/book | keine | Terminal step.component.ts:1443 | — | | Auftrag abschliessen | Produktionsleiter | PWA production.productionorders | ProductionOrders !' complete | ' ProductionOrders / completed | production.ts:69 | — |

Ø=ßâ Die reifste Kette der Plattform — und die einzige mit Benachrichtigungen (zwei Regeln, started und completed, in drei Mandanten-Kopien).

& Zwei Fallstricke, die in die Karte gehören:

Der Materialverbrauch hängt an Geschäftsnummern, nicht an Array-Positionen (warehouseConsume.ts:10-20). Fehlen sie, wird der Verbrauch still übersprungen, ohne Fehler.

Ein Buchungs-Problem wird zur Warnung, nie zum Fehler — foldConsumeOutcome (warehouseConsume.ts:69-72) übersetzt selbst eine Unterdeckung in eine Warnung, damit ein produzierter Schritt nie als Fehlschlag erscheint. Das ist fachlich richtig und heisst zugleich: eine fehlgeschlagene Materialbuchung ist am Terminal nicht als Fehler sichtbar.

Kette 8 — Verpackung und Versand

| Schritt | Rolle | Gerät/Fläche | Buchung | Benachrichtigung | Ist-Fläche | Lücke | |---|---|---|---|---|---| | Packauftrag entsteht | System | keine | PackConfirmations anlegen · warehouse.packconfirmations | keine | routerPackConfirmations.ts | Ø=Ý4 kein Erzeuger, keine Fläche | | Karton befüllen | Werker | keine | — | keine | — | Ø=Ý4 fehlt vollständig | | Karton versiegeln | Werker | keine | PackConfirmations !' seal-box | keine | routerPackConfirmations.ts:215 | Ø=Ý4 Aktion ohne Aufrufer in beiden Clients | | Packen abschliessen | Werker | keine | PackConfirmations !' complete | keine | routerPackConfirmations.ts:216 | Ø=Ý4 Aktion ohne Aufrufer, bucht nicht aus | | Versandkarton verwalten | Lagerleiter | Terminal warehouse/shipping-boxes | ShippingBoxes (purchase-api) CRUD | keine | Terminal shipping-boxes.component.ts:231,248 | die einzige Fläche der ganzen Kette | | Versand | System | — | Shippings | ' outfordelivery · delivered · exception | notificationpolicies | die Benachrichtigungen sind da, der Vorgang davor nicht |

Ø=Ý4 Die Kette endet nach dem Pick. Alle drei Packbestätigungs-Methoden sind in beiden Client-Diensten deklariert (createPackConfirmation, sealBox, completePackConfirmation) und haben in keinem Client eine Fläche. Der einzige gebaute Teil ist die Verwaltung der Versandkartons — als Stammdaten, nicht als Vorgang.

& Der Kontrast ist die Aussage: für den Versand danach existieren drei Benachrichtigungs-Regeln (Shippings), für das Packen davor keine einzige — und keine Fläche. Der Übergang vom Lager zum Versand ist die grösste zusammenhängende Lücke der Karte.

Träger: warehouse.pick-commission-ship (R6, plan:134).

Kette 9 — Retoure

| Schritt | Rolle | Gerät/Fläche | Buchung | Benachrichtigung | Ist-Fläche | Lücke | |---|---|---|---|---|---| |
 Retoure anmelden | System | keine | ReturnOrders anlegen · warehouse.returnorders | keine |
 warehouse.ts:119 | kein Erzeuger gemessen | | Prüfen und freigeben | Lagerleiter | Scanner returns |
 ReturnOrders !' approve/reject | keine | returns.component.ts:300,311 | — | | Ware annehmen | Werker |
 Scanner returns | ReturnOrders !' receive | keine | returns.component.ts:334 · routerReturnOrders.ts:273
 | Ø=Ý4 bucht den Bestand NICHT zurück | | Zustand begutachten | Werker | Scanner returns | ReturnOrders !'
 inspect | keine | returns.component.ts:340 | — | | Ins Sperrlager | Werker | keine | — | keine | — | Ø=Ý4 fehlt
 vollständig | | Abschliessen | Werker | Scanner returns | ReturnOrders !' complete | keine |
 returns.component.ts:403 | Ø=Ý4 bucht nicht |

Ø=Ý4 Der dritte schwere Bruch. routerReturnOrders.ts fasst Stocks/StockLogs über alle sechs Aktionen null-mal an (Positiv-Kontrolle: routerStocks.ts 18 Treffer derselben Suche). Eine zurückgenommene Ware wird begutachtet, freigegeben und abgeschlossen — und ist im Bestand nie angekommen. Ein Sperrlager-Begriff existiert nirgends im Bestand.

Träger: neu — siehe Lücken-Register L9.

Kette 10 — Nachschub

| Schritt | Rolle | Gerät/Fläche | Buchung | Benachrichtigung | Ist-Fläche | Lücke | |---|---|---|---|---|---| |
 Mindestbestand unterschritten | System | keine | — | keine | — | Ø=Ý4 kein Erzeuger !' purchase.reorder-
 demand (R4, plan:103) | | Auftrag übernehmen | Werker | Scanner replenish | ReplenishmentOrders !'
 assign | keine | replenish.component.ts:209 | — | | Nachfüllen | Werker | Scanner replenish |
 ReplenishmentOrders !' complete | keine | replenish.component.ts:251 ·
 routerReplenishmentOrders.ts:169 | Ø=Ý4 bucht den Bestand NICHT um |

Ø=Ý4 Ein Nachschub-Vorgang, der den Bestand nicht vom Reserve- auf den Kommissionierplatz bewegt, ist eine Quittung ohne Wirkung. Der Umlager-Weg dafür existiert (Stocks !' transfer, Kette 3) und wird nicht gerufen.

Kette 11 — Rückstand

| Schritt | Rolle | Gerät/Fläche | Buchung | Benachrichtigung | Ist-Fläche | Lücke | |---|---|---|---|---|---| |
 Rückstand entsteht | System | PWA warehouse.backorders (Liste) | Backorders anlegen ·
 warehouse.backorders | keine | warehouse.ts:9 | kein Erzeuger gemessen | | Rückstand auflösen |
 Lagerleiter | keine | Backorders !' fulfill (Stocks nur lesend) | keine | routerBackorders.ts:308 | Ø=Ý4 Aktion
 ohne Aufrufer; Methode im Scanner-Dienst deklariert, von keiner Seite gerufen | | Rückstand stornieren
 | Lagerleiter | keine | Backorders !' cancel | keine | routerBackorders.ts:307 | Ø=Ý4 Aktion ohne Aufrufer |

Kette 12 — Dock und Anlieferung

| Schritt | Rolle | Gerät/Fläche | Buchung | Benachrichtigung | Ist-Fläche | Lücke | |---|---|---|---|---|---| |
 Zeitfenster buchen | Einkäufer | keine | DockAppointments anlegen | keine | warehouse.ts:20 | Ø=Ý4 keine
 Fläche | | Ankunft bestätigen | Lagerleiter | keine | DockAppointments !' confirm/arrive | keine |

routerDockAppointments.ts:242,243 | Ø=Ý4 im Terminal-Dienst deklariert (confirmDockAppointment), ohne Aufrufer; im Scanner-Dienst gar nicht vorhanden | | Verladen | Werker | keine | DockAppointments !' start-loading/complete | keine | routerDockAppointments.ts:244,245 | Ø=Ý4 keine Fläche | | Tor belegen/freigeben | Lagerleiter | keine | Docks !' occupy/release | keine | routerDocks.ts:154,155 | Ø=Ý4 keine Fläche | | Nichterscheinen | Lagerleiter | keine | DockAppointments !' no-show | keine | routerDockAppointments.ts:246 | Ø=Ý4 keine Fläche |

Ø=Ý4 Die vollständig unerreichbare Kette. Elf Aktions-Routen über zwei Entitäten, keine einzige Fläche in Scanner, Terminal, PWA oder Wallboard. Für Think Tank Tegernsee heute ohne Bedarf — sie gehört in die Karte, weil sie sonst bei der nächsten Planung als „fehlt“ neu erfunden wird.

Lücken-Register — jede Lücke mit genau einem Träger

Träger-Arten: F = R-Programm-Feature · E = Task im Epic „Scanner-App neu“ · M = Task in den M-B-Sprints „Lager 1/2“ · N = neu (nur mit Begründung, CEO-Freigabe offen).

| # | Lücke | Kette | Träger | Art | |---|---|---|---|---| | L1 | Scanner bucht den Bestand, legt aber keinen Wareneingangs-Beleg an | 1 | Task 6a941ea4fad888ed370600a9 („Wareneingang im Scanner erzeugt einen GoodsReceipt mit Positionen und Foto“) | M | | L2 | PutAwayOrders.complete bucht den Bestand nicht auf den Platz | 2 | warehouse.think-tank-storage-structure (R4, plan:99) | F | | L3 | Pick-Kette vollständig bestandsneutral; book-outgoing existiert und wird nie gerufen — inkl. der abweichenden Schreib-Reihenfolge dort | 5 | warehouse.pick-commission-ship (R6, plan:134) | F | | L4 | Verpackung/Versand ohne jede Fläche, drei Aktionen ohne Aufrufer | 8 | warehouse.pick-commission-ship (R6, plan:134) | F | | L5 | ReturnOrders.receive bucht nicht zurück; kein Sperrlager-Begriff im Bestand | 9 | neu — die R6-Kette endet beim Versand, die Retoure ist der Rückweg und in keinem Feature enthalten (geprüft gegen alle 23) | N | | L6 | ReplenishmentOrders.complete bucht den Bestand nicht um | 10 | purchase.reorder-demand (R4, plan:103 — Mindestbestand! Bedarf) | F | | L7 | Null Benachrichtigungen für die gesamte WMS-Achse (0 von 151 Regeln) | 1,4,5,8,9,10 | automation.order-chain-notifications (R6, plan:136) | F | | L8 | Alle 13 WMS-Entitäten ohne entityActions — Prozess ohne Deklaration | alle | neu — Fundament-Arbeit, in keinem Feature und keinem Task enthalten (geprüft gegen 23 F, 8 E, 15 M) | N | | L9 | Kein Einstieg für Lager und Lagerorte (keine Liste, kein Baum) | alle | Task 6a9932b26edab122065e84f4 („Einstieg Lager & Lagerorte verwalten“) | E | | L10 | Raster-Generator-Oberfläche existiert, ist aber ohne Route unerreichbar | 2 | warehouse.scanner-grid-ui (R4, plan:96) | F | | L11 | storage-area-detail liest keinen Scan — die einzige Lagerort-Fläche | 2 | Task 6a9932a16edab122065e8293 (Scan-Bus auf alle Seiten) | E | | L12 | Einstellungen sind sieben Listenpunkte ohne Verhalten | alle | Task 6a9932b76edab122065e8580 | E | | L13 | Dock/Anlieferung: 11 Aktionen über 2 Entitäten, null Flächen | 12 | zurückgestellt — für Think Tank Tegernsee heute ohne Bedarf; in der Karte, damit sie nicht neu erfunden wird — | | L14 | Wallboard: 20 von 21 Endpunkten produktionsbezogen, für Lager genau einer (als „legacy“ markiert), kein Widget für Bestände/Picklisten | alle | reporting.production-floor-wallboards (R7, plan:155) | F | | L15 | TransportOrders/Backorders: Aktionen ohne jeden Aufrufer | 3,11 | purchase.reorder-demand (R4) für Backorders; TransportOrders mit L13 zurückgestellt | F |

Bilanz: 13 Lücken zugeordnet · 2 neu · 1 zurückgestellt. Die zwei neuen (L5, L8) sind gegen alle 23 Features, 8 Epic-Tasks und 15 M-B-Tasks geprüft. Ø=Ý4 Kein PM-Task angelegt — das braucht die CEO-Freigabe (Brief-Constraint).

& Beobachtung ohne Träger (gehört in den SDUI-Strang, nicht hierher): das Modul warehouse trägt 13

reine Listen-Zustände und keinen einzigen .edit.*-Detail-Reiter, während purchase 13 und production 22 tragen. Wer in der PWA einen Wareneingang oder eine Pickliste öffnen will, findet keine Bearbeitungsfläche.

Bau-Reihenfolge für Task 6a9932ba6edab122065e860c (REP3)

Ø=Ý4 Der Auftrag jenes Tasks lautet „Prozess-Screens nachziehen“ — gemessen ist er ein SCHLIESS-, kein Bau-Auftrag. Alle sieben genannten Screens (Wareneingang, Einlagern, Umlagern, Inventur, Pick, Kommission, Retoure) existieren im Scanner und rufen echte Aktions-Routen. Was fehlt, ist die Buchung dahinter.

Ø=ßâ Und die Buchungswege existieren ebenfalls alle — book-incoming, book-outgoing, transfer, count, apply, consume sind gebaut und getestet. Es ist Verdrahtung, kein Neubau.

Kriterium je Position: (a) Buchungsweg im Backend vorhanden · (b) Scan-Grundfunktion trägt den Schritt · (c) kein R4-Feature liefert den Screen ohnehin.

1. Die 13 Entitäten deklarieren (L8) — Fundament, vor allem anderen

Ohne entityActions im Register gibt es keine SDUI-Schaltfläche, keinen Eintrag in der Scan-Aktionsliste und kein abgeleitetes Recht — heute läuft jede der ~50 Aktionen auf dem Basis-Modulrecht. Jede folgende Position wird billiger, wenn das zuerst steht. & Die Stocks-Buchungen liegen auf Sammlungs-Ebene (/stocks/actions/*) und brauchen deshalb availability:'collection' — der Routen-Erzeuger kennt diese Form (belegt in CLAUDE.md, Kaskade 61). (a) ' · (b) entfällt · (c) '

2. Einlagern bucht ein (L2)

Ohne Bestand am Lagerplatz ist jede Folgekette wertlos — Pick, Nachschub und Inventur rechnen alle gegen einen Bestand, der heute nur am Wareneingang entsteht. Weg: PutAwayOrders.complete ruft Stocks '! transfer (Eingangszone '! Zielplatz). (a) ' · (b) ' (putaway scannt Platz und Menge) · (c) '

3. Einstieg Lager und Lagerorte (L9, L10, L11)

Der CEO-Befund Nr. 1 und zugleich die Voraussetzung für Position 2: man muss einen Lagerplatz sehen, auswählen und anschauen können. Die Raster-Oberfläche existiert bereits im Zusatzblatt und braucht nur einen Routen-Einstieg. (a) ' (generate-grid, print-cell-labels) · (b) & heute nein — storage-area-detail liest keinen Scan, das schliesst L11 mit · (c) & Überschneidung mit warehouse.scanner-grid-ui (R4) — dort bauen, hier nur verlinken, sonst doppelt.

4. Pick bucht aus (L3)

book-outgoing scharfschalten aus PickLists.complete. & Vorher die Schreib-Reihenfolge angleichen (heute Bestand vor Journal, überall sonst Journal vor Bestand) — sonst wird eine bestehende Abweichung zum ersten Mal wirksam. (a) ' · (b) ' · (c) '

5. Packen und Versand (L4)

Die erste Position, die eine neue Fläche braucht (Packauftrag, Karton befüllen, versiegeln). Die drei Aktionen sind gebaut, book-outgoing liefert die Ausbuchung, und die Benachrichtigungen für den

Versand danach existieren bereits. (a) ' · (b) ' (Karton- und Artikel-Scan) · (c) '

6. Retoure bucht zurück (L5)

Rückweg über book-incoming auf einen Sperrbereich. & Der Sperrlager-Begriff existiert nirgends — das ist der Teil, der eine Entscheidung braucht (eigener Lagerort oder Kennzeichen am Bestand), und der Grund, warum L5 als neu geführt wird. (a) ' · (b) ' · (c) '

7. Nachschub bucht um (L6)

ReplenishmentOrders.complete ruft Stocks !' transfer. Technisch die kleinste Position, fachlich erst sinnvoll, wenn Mindestbestände gepflegt sind (purchase.reorder-demand, R4). (a) ' · (b) ' · (c) '

8. Benachrichtigungen je Stufe (L7)

Erst hier, weil eine Benachrichtigung ein Ereignis braucht, das etwas bewirkt hat. Die vier lohnendsten sind gemessen: Bestandskorrektur wartet auf Freigabe · Pickliste liegt bereit · Material fehlt ·

Packauftrag offen. & Vier Sprachen (de/en/es/it) über ayounet translations, nie im Code. (a) — · (b) — · (c) ' (automation.order-chain-notifications, R6)

9. Wallboard-Widgets für das Lager (L14)

Zuletzt, weil ein Wallboard nur zeigt, was die Ketten davor erzeugen. Die Modelle Stocks und Warehouses sind im BFF bereits geladen (app.ts:126,127) und werden von keinem Widget gelesen. (a) ' · (b) — · (c) ' (reporting.production-floor-wallboards, R7)

Anforderungen an jeden Screen dieser Reihenfolge

Aus den geltenden CEO-Regeln, in jede Zeile des Lücken-Registers zu übernehmen, die einen Screen verlangt:

- Scan-Grundfunktionen gelten in jeder Geräterolle (CEO 2026-08-30). Eine Rolle bestimmt Startseite, Fokus und Menü — nie ob gescannt werden darf.
- Jeder Knopf trägt einen Loader (CEO 2026-08-23). Ein stiller Klick ist ein Defekt.
- Alles viersprachig (de/en/es/it), Übersetzungen ausschliesslich in ayounet translations (CEO 2026-08-23) — nie im Code.
- Keine Freitexte, wo die Datenbank die Werte führt (CEO 2026-08-26): Lagerort, Produkt, Arbeitsplatz und Charge sind atomare Referenz-Auswahlfelder mit Schnell-Anlage.
- Kein Hinweis auf laufende Entwicklung in der Oberfläche (CEO 2026-08-23).

Was diese Karte NICHT plant

- Scan-Bus und Hardware-Matrix — vergeben an Task 6a9932a16edab122065e8293, geteilter Wedge-Detektor @tolinax/ayounet-scan-ui. Die Karte hält nur fest, dass im Bestand drei Scan-Muster nebeneinander laufen (globaler Wedge · Seiten-Handler · Kamera).

- Mehrsprachigkeit und Interfaces-Pin des Scanners — beide seit TK20 erledigt, hier bewusst keine Lücke.

- UX, Informationsarchitektur, Lesbarkeit — Task 6a9932a66edab122065e8347.

Replikation — Eigenfertigung Ende-zu-Ende (Programm-Plan)

Replikation — Eigenfertigung Ende-zu-Ende (Programm-Plan, Stand 2026-09-02)

Zielbild: tolinax fertigt die eigenen Geräte (Wallboard, Terminal, Control Panel 7/10, Basic-Sensor) Ende-zu-Ende auf der eigenen Plattform — Dogfooding des Produktionsmoduls, Fabrikations-Erweiterung (3D-Druck, Laserschnitt, Lasergravur), Think-Tank-Fertigungsservice für Kunden und Vorlage für Demo-Mandanten. Endziel der Produktschiene: die vier Geräte verkaufbar im Shop auf tolinax.com, Gehäuse aus dem eigenen Druck (Prototyp im Think Tank, Serie über JLC3DP). Live-Abnahme Shop !' Fertigung !' Lieferung: Ende Oktober 2026.

CEO-Entscheide (2026-09-02): eigene Meilenstein-Reihe R1–R8 · VGT = die drei verwaisten Fabrik-Wallboards (Mandant 5d7fa3c4...) als Vorlage · Hardware Rev B: Wallboard Pi 5 + NVMe-SSD, Terminal Pi + 10-Zoll-Touch + Kamera + 2D-Scanner, Control Panel 7 und 10 Zoll, Basic-Sensor ESP32 + BME280 + LD2410 · Energie jetzt über Home Assistant, langfristig eigene offline-/self-host-fähige Shelly-Integration · Verkauf: volle Kette Bestellung !' Auftrag !' Produktionsauftrag !' Materialbestellungen !' Pick !' Kommission mit allen Benachrichtigungen, make-to-order/make-to-stock je Produkt · Demo: generischer Fertigungsbetrieb · Gehäuse-Generator als generisches, wiederholbares Schema · JLC3DP zuerst · E3/E8/E9/E10 ins Programm gezogen · Fertigungsservice deckt FDM-Druck, CO2-Laserschnitt und Dioden-Laser-Gravur (vier Produktionsarten inkl. Montage) · keine Fundament-Felder erfinden, die es schon gibt (jedes Add gegen den Vertrag gemessen).

Bestand vs. Lücke (gemessen 2026-09-02)

| Achse | Existiert | Fehlt | |---|---|---| | Eigenprodukte | 4 Geräte im PIM (Pi-3/4-BOMs), 4 Produktionsschemas, production[] ohne Titel | Control Panel/Sensor fehlen, keine Rev/BOM-Status, 0 Beschreibungen, Pi 5/SSD nicht abgebildet | | Produktion | Arbeitsplan = ProductionProcess.steps !' Produkt-Kopie !' Auftrags-Instanz; Terminal precheck/running/material/qc/booking; QC-Prüfpläne; OEE; WorkLogs; 445 Alt-Screenshots | Seriennummern-Vergabe, Serie!"Produktionsauftrag!"Gerät, Factory-Test-Plan je Produkt | | Energie/Telemetrie | HA-Sync live !' Sensors/SensorLogs; kWh!'€-Motor | IoT-Nachkalkulation + CostRates kind=energy | Shelly nur via HA; Sensor!"Maschine 0 befüllt; 0 Writer für Machine.monitoring.values; kein Rückfluss in Kosten/OEE; Bambu 0 Zeilen Code | | Fabrikation | 3D-Viewer-Lib, 3D-MIME in Uploads, IDrawings dxf/stl, FAB-Programm M1–M5/E1–E10 | Filament als Werkzeug (E3), Druck-Schritt + Kosten (E8), Gehäuse-Modelle | | Lager | Raster-Generator gebaut (generate-grid, Zellen-Etiketten-Sheet), Scanner 19 Routen, M-B Dymo | Scanner-UI für Raster/Sheet fehlt (Fach + deaktiviert) | | Wallboards | 20+ Produktions-Widgets, WallBoards je Halle mit _workcenter | VGT-Mandant gelöscht; 3 verwaiste Wallboards | | Shop | consumer-app /shop, /cart, /checkout mit Stripe/PayPal; Orders | keine Geräte-Kategorie/-Seite, Shops nur WooCommerce-Entwurf, Order!"Produktionsauftrag nicht verknüpft |

Fundament-Adds (gegen die Verträge geprüft)

- Neu: IProduct.hardwareRevision/bomRevision/bomStatus · ISerial.productionOrder/iotDevice/hardwareRevision · IMachine.iotDevice · IDrawings.step/threeMf · Entität CadTemplates · Entität PrintJobs (E3) · PageBlockType fabrication-order · nach Messung ggf. IOrder.assignment, IStock.minQty/reorderQty.
- Nutzen statt erfinden: material[].availableProductVariants (Alternativteile), material[].name (Bezeichner), procurement (Make/Buy), delivery_time, warranty, useSerials, package (= Verpackungseinheit, kein Footprint), ProductionOrder.assignment/internalOrder/internalProductionOrder/offer/invoice/shipping, CostRates(kind=energy, source=iot), Sensor.machine/workCenter, IIoTDeviceBlueprint, QualityControlTemplates, IFactoryTestProtocol, WMS picklists/materialcommissionings, Offers/Orders/Documents.

Abnahme-Gates

- R1 — Eigenprodukte stehen: Rev B, Stücklisten, Komponenten-Katalog, Kalkulation (6a97e53af7ba5c7579542285, 2026-09-07 !' 2026-09-20): Abnahme: fünf Produkte Rev B (Wallboard, Terminal, Control Panel 7/10, Basic-Sensor) mit gerolltem Preis aus der Stückliste, BOM released, Rev-A-Produkte als Auslauf.
- R2 — Werker-Kreislauf läuft: Produktionsschema, Arbeitspläne, QC, Seriennummern, Pilot-Auftrag (6a97e53bf7ba5c7579542310, 2026-09-07 !' 2026-10-04): Abnahme: ein Wallboard als interner Produktionsauftrag Ende-zu-Ende am Terminal mit Seriennummer, QC-Protokoll und WorkLog.
- R3 — Energie & Telemetrie zählen: Shelly/HA !' Maschinenkosten, OEE, Drucker-Fundament (6a97e53cf7ba5c7579542395, 2026-09-21 !' 2026-10-18): Abnahme: Nachkalkulation eines Auftrags zeigt gemessene kWh × Strompreis; H2S und Bibo 2 als FDM-Maschinen mit Filament als Werkzeug. Enthält E3 (verschoben).
- R4 — Think-Tank-Lager beschriftet: Raster, Etiketten, Kommission an den Arbeitsplatz (6a97e53ff7ba5c757954241a, 2026-09-07 !' 2026-10-04): Abnahme: alle Mikroelektronik-Kästen als Raster angelegt, gescannt und beschriftet; ein Pick an den Arbeitsplatz gebucht. Enthält E10 (verschoben). Dockt an M-B.
- R5 — Gehäuse aus dem eigenen Druck: Gehäuse-Generator, vier Prototypen, Druck-Schritt im Arbeitsplan (6a97e540f7ba5c757954249f, 2026-10-05 !' 2026-11-01): Abnahme: vier Gehäuse (Wallboard Pi5+SSD, Terminal umschließend, Control Panel 7 und 10) aus dem parametrischen Schema gedruckt, passend, als Revision freigegeben, Druckkosten je Stück im Costing. Enthält E8 (verschoben).
- R6 — Shop !' Fertigung !' Lieferung live, Think-Tank-Fertigungsservice für Kunden (6a97e543f7ba5c7579542524, 2026-10-05 !' 2026-11-01): Abnahme: Testkauf im Shop auf tolinax.com bis Versand und Rechnung durchgelaufen, plus ein Kunden-Auftrag je Verfahren (3D-Druck, Laserschnitt, Gravur) über das Website-Widget. Enthält E9 (verschoben).
- R7 — Produktionsflächen-Wallboards (VGT-Vorlage) & Demo-Mandant Fertigungsbetrieb (6a97e544f7ba5c75795425a9, 2026-10-05 !' 2026-11-01): Abnahme: Demo-Mandant mit drei Hallen-Wallboards (Produktionsschemas, Status, OEE, Andon) live, Superuser-Demo-Zugang. Dockt an M-F Demo-Fabrik.
- R8 — Serie & Anbindungen: JLC3DP, eigene Shelly-Integration, Lieferanten-APIs (Nachfolge) (6a97e545f7ba5c757954262e, 2026-11-02 !' 2026-11-29): Nachfolge-Paket nach der Oktober-Abnahme:

Serienfertigung der Gehäuse über JLC3DP-API, offline-/self-host-fähige Shelly-Integration, weitere Lieferanten-Anbindungen.

Sprints

- Replikation 1 — Eigenprodukte, Lager-Raster, Fundament-Kaskade (6a97e566f7ba5c7579542ee5, 2026-09-07 !' 2026-09-20)
- Replikation 2 — Werker-Kreislauf, Energie-Messung, Pilot-Auftrag (6a97e567f7ba5c7579542f69, 2026-09-21 !' 2026-10-04)
- Replikation 3 — Gehäuse-Generator, Shop-Aufbau, Fertigungsservice (6a97e56af7ba5c7579542fee, 2026-10-05 !' 2026-10-18)
- Replikation 4 — Auftragskette live, Demo-Mandant, Wallboards, Abnahme (6a97e56bf7ba5c7579543072, 2026-10-19 !' 2026-11-01)
- Replikation 5 — Serie: JLC3DP, Shelly nativ (6a97e56cf7ba5c75795430f6, 2026-11-02 !' 2026-11-15)
- Replikation 6 — Anbindungen abschließen, Nachlauf (6a97e56ff7ba5c757954317a, 2026-11-16 !' 2026-11-29)

Epics, Features, Tasks

Produktstammdaten Rev B & Komponenten-Katalog (6a97e548f7ba5c75795426b1, R1, Modul pim)

Vier Geräte plus Basic-Sensor als Produkte Rev B mit vollständigen Stücklisten (Gehäuse als Halbfabrikat), Komponenten-Katalog mit Lieferantenvereinbarungen, Kalkulationsschema Gerätemontage mit Stücklisten-Kostenrollup, Zeichnungen und Datenblätter. Fundament: hardwareRevision/bomRevision/bomStatus (geprüft: existiert nicht; Alternativteile = material[].availableProductVariants, Make/Buy = procurement).

- Produkt- und Stücklisten-Revision pim.product-revisions (6a97e570f7ba5c75795431fd): Fundament: hardwareRevision, bomRevision, bomStatus (draft/released) an IProduct; PWA-BOM-Reiter; Feld-Register.
- [Feature/critical/REP1] R1: Fundament-Kaskade Replikation — hardwareRevision/bomRevision/bomStatus an IProduct, ISerial.productionOrder/iotDevice/hardwareRevision, IMachine._iotDevice, IDrawings.step/threeMf, Entität CadTemplates, PageBlockType fabrication-order (6a97e5e9f7ba5c75795449cb)
- [Aufgabe/high/REP1] R1: PWA-Stücklisten-Reiter mit Revision, Status, Alternativteilen und Rollup-Spalte; Feld-Register nachziehen (6a97e5ebf7ba5c7579544a5e)
- Komponenten-Katalog Eigenfertigung pim.component-catalog (6a97e571f7ba5c7579543281): Alle Bauteile der Geräte als Einkaufsartikel mit Lieferantenvereinbarung, Bild, Shop-URL, Preis, Gewicht/Maße (Pi 5, NVMe-HAT, SSD, Netzteile, 7/10-Zoll-Touch, Kamera, 2D-Scanner, ESP32, BME280, LD2410, Kleinteile, Filament).
- [Aufgabe/high/REP1] R1: Komponenten-Katalog anlegen — alle Bauteile der fünf Geräte als Einkaufsartikel mit Lieferantenvereinbarung, Bild, Shop-URL, Preis, Gewicht/Maße (6a97e5f0f7ba5c7579544af1)

- Geräte-Produkte Rev B pim.device-products-rev-b (6a97e576f7ba5c757954330c): Wallboard, Terminal, Control Panel 7, Control Panel 10, Basic-Sensor als Produkte Rev B mit Texten de/en/es/it, Kategorie Geräte, Lieferzeit, Garantie, isPublic; Rev-A-Produkte als Auslauf; Gehäuse als Halbfabrikat-Produkte.
- [Feature/critical/REP1] R1: Fünf Geräte-Produkte Rev B anlegen (Wallboard, Terminal, Control Panel 7, Control Panel 10, Basic-Sensor) + Gehäuse als Halbfabrikate; Rev A als Auslauf (6a97e5f6f7ba5c7579544b97)
- [Aufgabe/high/REP1] R1: Basic-Sensor Firmware-Blueprint (ESP32 + BME280 + LD2410 mmWave) im Device Studio anlegen und mit dem Produkt verknüpfen (6a97e5fcf7ba5c7579544cb9)
- Stücklisten-Kostenrollup & Verkaufspreis pim.bom-cost-rollup (6a97e577f7ba5c7579543396): Kalkulationsschema Gerätemontage (Material + Lohn + Maschinenstunde + Energie + GK), Rollup aus material[] über Halbfabrikate, Marge !' Shop-Preis.
- [Feature/high/REP1] R1: Stücklisten-Kostenrollup bauen — Materialkosten aus material[] über Halbfabrikate + Lohn + Maschinenstunde + Energie + GK !' total_costs/sellPrice (6a97e601f7ba5c7579544d4c)
- [QS/medium/REP1] R1-Abnahme: fünf Produkte Rev B mit gerolltem Preis, BOM released, Rev A Auslauf — Beleg-Handoff (6a97e605f7ba5c7579544ddf)
- Zeichnungen, Maßblätter, Datenblätter am Produkt pim.drawings-datasheets (6a97e579f7ba5c757954341a): Dokumente mit Revisionskette (Documents.version) am Produkt und je Arbeitsschritt; Firmware-Blueprint-Bezug am Sensor.
- [Aufgabe/medium/REP1] R1: Maßblätter, Datenblätter, Zeichnungen an Produkte und Arbeitsschritte hängen (Documents mit Version, IDrawings) (6a97e60df7ba5c7579544ec1)

Produktionsschemas, Arbeitspläne, QC & Seriennummern der Gerätefertigung (6a97e549f7ba5c7579542734, R2, Modul production)

Produktionsschema Geräte-Montage (Kommission !' Montage !' Flash !' Factory-Test !' Endkontrolle !' Verpackung), detaillierte Arbeitspläne je Gerät als ProductionProcess.steps mit Zeiten, Material, Zeichnung, QC in-process/after und Abnahme; QC-Prüfpläne; Seriennummern-Kette Serie!"Produktionsauftrag!"IoTDevice; Arbeitsplätze im Think Tank; Pilot-Auftrag am Terminal.

- Produktionsschema Geräte-Montage config.production-schema-device-assembly (6a97e57cf7ba5c757954352d): Phasen Kommission !' Montage !' Flash/Provisionierung !' Factory-Test !' Endkontrolle !' Verpackung mit Abteilungen, Produktionsregeln (Planzeit) und Costing-Zuordnung.
- [Feature/critical/REP1] R2: Produktionsschema Geräte-Montage anlegen — Phasen Kommission !' Montage !' Flash/Provisionierung !' Factory-Test !' Endkontrolle !' Verpackung (6a97e615f7ba5c7579544f78)
- Arbeitspläne je Gerät (detailliert) config.work-plans-devices (6a97e57df7ba5c75795435b1): ProductionProcess.steps je Gerät: Arbeitsplatz/Maschine, Rüst-/Stück-/Nachzeit, Material je Schritt, Zeichnung je Schritt, QC in-process/after, Abnahme durch Rolle, Zählerstände. Referenz: Alt-Screenshots production/productionorders, config/productionprocesses.
- [Feature/critical/REP2] R2: Arbeitspläne je Gerät detailliert ausarbeiten (ProductionProcess.steps): Arbeitsplatz/Maschine, Rüst-/Stück-/Nachzeit, Material je Schritt, Zeichnung, QC, Abnahme, Zählerstände (6a97e619f7ba5c7579545011)
- QC-Prüfpläne Gerätefertigung config.qc-plans-devices (6a97e57ef7ba5c7579543635): QualityControlTemplates: Sichtprüfung, Funktionstest (Boot/Netz/Touch/Kamera/Scanner), Factory-Test-

Protokoll als QC-Schritt, Sensor-Kalibrierung, Maßhaltigkeit Gehäuse.

- [Feature/high/REP2] R2: QC-Prüfpläne (QualityControlTemplates) — Sichtprüfung, Funktionstest Boot/Netz/Touch/Kamera/Scanner, Factory-Test-Protokoll als QC-Schritt, Sensor-Kalibrierung, Maßhaltigkeit Gehäuse (6a97e61ef7ba5c757954513b)
 - Seriennummern-Kette Serie !" Produktionsauftrag !" Gerät config.serial-number-chain (6a97e581f7ba5c75795436b9): Fundament: ISerial.productionOrder, iotDevice, hardwareRevision; Nummernkreis über die bestehende Nummern-Mechanik (messen); Vergabe bei Abnahme; Serien-Etikett mit QR.
 - [Feature/high/REP2] R2: Seriennummern-Kette — Nummernkreis je Produkt/Revision über die bestehende Nummern-Mechanik, Vergabe bei Abnahme, Serie !" Produktionsauftrag !" IoTDevice, Serien-Etikett mit QR (6a97e623f7ba5c75795451ce)
 - Arbeitsplätze, Maschinen, Ressourcen Think Tank production.think-tank-workcenters (6a97e582f7ba5c7579543744): WorkCenters (Montage, Flash, Endkontrolle, Verpackung), Maschinen (H2S, Bibo 2, K40, Sculpfun, Lötstation) und ProductionResources angelegt und verknüpft.
 - [Aufgabe/high/REP1] R2: Arbeitsplätze (Montage, Flash, Endkontrolle, Verpackung), Maschinen (H2S, Bibo 2, K40, Sculpfun, Lötstation) und Ressourcen im Think Tank anlegen und verknüpfen (6a97e628f7ba5c757954525a)
 - Pilot: Wallboard als interner Produktionsauftrag am Terminal production.pilot-order-wallboard (6a97e585f7ba5c75795437cf): Ende-zu-Ende: Start, Material-Consume, QC, Badge-Abnahme, WorkLog, Report, Seriennummer; Beleg-Handoff.
 - [QS/critical/REP2] R2-Abnahme: Ein Wallboard als interner Produktionsauftrag Ende-zu-Ende am Werker-Terminal — Start, Material-Consume, QC, Badge-Abnahme, WorkLog, Report, Seriennummer (6a97e62df7ba5c75795452ed)

Maschinen-Telemetrie & Energiekosten: Shelly/HA !" Nachkalkulation, OEE, Wallboard (6a97e54bf7ba5c75795427b7, R3, Modul iot)

Shelly-Steckdose über den lebenden Home-Assistant-Sync (Sensors/SensorLogs), Sensor!"Maschine-Zuordnung (Felder existieren, 0 befüllt), Rückfluss der iot-Nachkalkulation ($\text{kWh} \times \text{CostRate kind=energy}$) in Auftrags-Costing und Maschinenkosten, erster Schreiber für Machine.monitoring.values, echte OEE, Wallboard-Kachel. Zusammen mit E3 (Drucker-Fundament, verschoben).

- Energie-Messung Shelly über Home Assistant iot.energy-metering-shelly-ha (6a97e587f7ba5c7579543853): Shelly Plug am H2S in HA, HA-Sync-Credentials tolinax, Sensor erscheint in Sensors; Sensor!"Maschine/Arbeitsplatz-Zuordnung als PWA-Fläche (Felder existieren, 0 befüllt).
- [Aufgabe/high/REP2] R3: Shelly Plug S am H2S in Home Assistant einbinden, HA-Sync für tolinax konfigurieren, Sensor erscheint in Sensors/SensorLogs (6a97e635f7ba5c7579545394)
- [Feature/high/REP2] R3: Sensor !" Maschine/Arbeitsplatz-Zuordnung als PWA-Fläche (Felder Sensor.machine/workCenter existieren, 0 befüllt) + Zuordnung H2S/Bibo 2/K40/Sculpfun (6a97e63af7ba5c7579545420)
- Energiekosten in Nachkalkulation und Maschinenkosten production.energy-cost-rollback (6a97e589f7ba5c75795438d7): CostRate kind=energy (Strompreis tolinax); iot-Nachkalkulation je Produktionsauftrag persistieren (Auftrags-Costing, CostRates source=iot); Maschinen-Kostensatz aus

Messung.

- [Feature/critical/REP2] R3: Energiekosten-Rückfluss — CostRate kind=energy (Strompreis), iot-Nachkalkulation je Produktionsauftrag persistieren (Auftrags-Costing, CostRates source=iot), Maschinen-Kostensatz aus Messung (6a97e63ff7ba5c7579545542)
- [QS/high/REP3] R3-Abnahme: Nachkalkulation eines echten Auftrags zeigt gemessene kWh × Strompreis; Filamentverbrauch je Druck gebucht (6a97e654f7ba5c7579545760)
- Maschinen-Telemetrie & echte OEE production.machine-telemetry-writer (6a97e58bf7ba5c7579543962): Erster Schreiber für Machine.components[].monitoring.values (Leistung/Temperatur) aus Sensoren, Laufzeit-Erkennung aus Leistungsschwelle, OEE aus Messung; Alert bei Überschreitung.
- [Feature/high/REP3] R3: Erster Schreiber für Machine.components[].monitoring.values aus Sensoren (Leistung/Temperatur), Laufzeit-Erkennung über Leistungsschwelle, echte OEE, Alert bei Überschreitung (6a97e647f7ba5c7579545616)
- Wallboard-Kachel Maschine & Energie reporting.wallboard-machine-energy (6a97e58df7ba5c75795439e6): company-wallboard: Maschinen-Widget um Live-Leistung, Energie je Auftrag und Druckerstatus erweitern; Andon bei Störung.
- [Aufgabe/medium/REP3] R3: Wallboard-Kachel Maschine/Energie — Live-Leistung, Energie je Auftrag, Druckerstatus, Andon (6a97e64ff7ba5c75795456cd)

Lagerplatz-Raster, Beschriftung & Kommission im Scanner (6a97e54ef7ba5c757954283a, R4, Modul warehouse)

Scanner-UI für den bereits gebauten Raster-Generator (generate-grid, Zellen-Etiketten-Sheet), Lagerstruktur Think Tank (Regal !' Kasten !' Zelle), Einlagerung der Mikroelektronik, Picklisten und Kommissionsbuchung an den Arbeitsplatz, Bedarfsmeldung bei Mindestbestand. Zusammen mit E10 (Druck & Ausgabe, verschoben); dockt an M-B LAGER1/2.

- Scanner-UI Raster anlegen & Zellen-Etiketten warehouse.scanner-grid-ui (6a97e590f7ba5c7579543a95): Box/Raster anlegen (Zeilen, Spalten, Namensschema, Typ), Fach + aktivieren !' generate-grid, Zellen-Etiketten-Sheet und Dymo-Einzeletikett aus der Scanner-App (Backend gebaut, FE fehlt).
- [Feature/critical/REP1] R4: Scanner-App — Box/Raster anlegen (Zeilen, Spalten, Namensschema, Typ), Fach + aktivieren !' generate-grid, Zellen-Etiketten-Sheet + Dymo-Einzeletikett (6a97e658f7ba5c75795457f2)
- [QS/high/REP2] R4-Abnahme: alle Think-Tank-Kästen als Raster angelegt, gescannt und beschriftet; ein Pick an den Arbeitsplatz gebucht (6a97e66cf7ba5c7579545ace)
- Lagerstruktur Think Tank & Einlagerung Mikroelektronik warehouse.think-tank-storage-structure (6a97e595f7ba5c7579543b68): Regal !' Kasten !' Zelle als Raster, alle Bauteile eingelagert (Batch-Modus aus M-B), Bestände je Bauteil, Mindestbestände.
- [Aufgabe/high/REP2] R4: Lagerstruktur Think Tank anlegen (Regal !' Kasten !' Zelle als Raster), alle Mikroelektronik-Bauteile einlagern, Bestände und Mindestbestände pflegen, Etiketten drucken (6a97e65af7ba5c757954587f)
- Picklisten & Kommission an den Arbeitsplatz warehouse.pick-to-workcenter (6a97e598f7ba5c7579543bf9): Materialbedarf aus Produktionsauftrag !' Pickliste !' Scanner-Pick !' Materialkommissionierung an WorkCenter (bestehende WMS-Entitäten picklists/

materialcommissionings).

- [Feature/high/REP2] R4: Picklisten aus Produktionsauftrag !' Scanner-Pick !'

Materialkommissionierung an den Arbeitsplatz (WMS-Entitäten picklists/materialcommissionings) (6a97e65ef7ba5c757954599a)

- Bedarfsmeldung & interne Materialbestellung purchase.reorder-demand

(6a97e59cf7ba5c7579543d0c): Unterschreitung Mindestbestand !' Bedarf !' interne Bestellung !' SupplierRequest/SupplierOrder mit Anbindungs-Slot (Bambu Store, Reichelt, JLC); Feld-Messung IStock min/reorder.

- [Feature/medium/REP2] R4: Bedarfsmeldung — Mindestbestand unterschritten !' Bedarf !' interne Materialbestellung !' Lieferantenanfrage/-bestellung; Feld-Messung IStock (kein min/reorder vorhanden) (6a97e665f7ba5c7579545a3b)

Gehäuse-Generator (parametrisches Schema) & Prototyp-Druck der vier Gehäuse (6a97e54ff7ba5c75795428bd, R5, Modul production)

Generisches, wiederholbares Schema für Gehäuse: Entität CadTemplates (Parameter, Presets, Quellcode OpenSCAD/CadQuery, Ausgabeformate STL/3MF/STEP/DXF), Generator-Worker mit Render, Bauteil-Maßblätter als Presets, vier Gehäuse als Prototypen auf dem H2S, Freigabe als Revision. Zusammen mit E8 (Druck-Schritt im Arbeitsplan, Kosten je Druck, QC, Wallboard, verschoben).

- Gehäuse-Generator: CadTemplates & Generator-Worker production.cad-templates

(6a97e5a1f7ba5c7579543dc2): Fundament: Entität CadTemplates (Parameter-Definition, Presets, Quellcode OpenSCAD/CadQuery, Ausgabeformate), IDrawings.step/threeMf; worker-cad rendert STL/3MF/STEP/DXF + Thumbnail nach Documents, Bindung an Produkt/Arbeitsschritt, 3D-Viewer.

- [Feature/critical/REP3] R5: Gehäuse-Generator — Entität CadTemplates (Parameter-Definition, Presets, Quellcode OpenSCAD/CadQuery, Ausgabeformate) + worker-cad (Render STL/3MF/STEP/DXF + Thumbnail nach Documents) + Bindung an Produkt/Arbeitsschritt + 3D-Viewer (6a97e670f7ba5c7579545b61)

- Bauteil-Maßblätter als Parameter-Presets production.cad-part-presets

(6a97e5a4f7ba5c7579543e46): Wiederverwendbare Bibliothek: Pi 5, NVMe-HAT, 7/10-Zoll-Touch, Kamera-Modul, 2D-Scanner, ESP32-Board, Sensor-Module — Maße, Bohrbilder, Anschlussseiten.

- [Aufgabe/high/REP3] R5: Bauteil-Maßblätter als Parameter-Presets — Pi 5, NVMe-HAT, 7/10-Zoll-Touch, Kamera-Modul, 2D-Scanner, ESP32-Board, Sensor-Module (Maße, Bohrbilder, Anschlussseiten) (6a97e675f7ba5c7579545c01)

- Vier Gehäuse aus dem Schema: Prototyp, Passprobe, Freigabe production.device-cases-rev-b (6a97e5a9f7ba5c7579543efd): Wallboard (Pi 5 + SSD, höher), Terminal (umschließend, Kamerafenster, Rückseite geschlossen, Clip), Control Panel 7 (Kabelführung links), Control Panel 10 (Kabel hinten); Prototyp-Druck auf dem H2S, Iteration, Freigabe als Revision, Renderings als Produktbilder.

- [Feature/critical/REP3] R5: Vier Gehäuse aus dem Schema erzeugen und auf dem H2S drucken — Wallboard (Pi 5 + SSD, höher), Terminal (umschließend, Kamerafenster, Rückseite geschlossen, Clip), Control Panel 7 (Kabel links), Control Panel 10 (Kabel hinten) (6a97e679f7ba5c7579545c8d)

- [QS/high/REP4] R5-Abnahme: vier Gehäuse gedruckt, passen, als Revision freigegeben, Druckkosten je Stück im Costing (6a97e67df7ba5c7579545da8)

- Halbfabrikat-Kalkulation Gehäuse pim.case-semi-finished-costing (6a97e5abf7ba5c7579543f81): Filament-Gramm × Preis + Maschinenstunde + Energie je Gehäuse als Halbfabrikat im Rollup der Geräte.

- [Aufgabe/medium/REP4] R5: Halbfabrikat-Kalkulation Gehäuse (Filament-Gramm × Preis + Maschinenstunde + Energie) im Rollup der Geräte (6a97e681f7ba5c7579545e34)

Geräte im Shop auf tolinax.com (6a97e550f7ba5c7579542940, R6, Modul ecommerce)

Produktkategorie Geräte, Shop-Seite und Navigation auf tolinax.com über die gebaute consumer-app-Shop-Strecke, Produktseiten mit 3D-Viewer und Varianten, Preise/Lieferzeit/Versandzonen/Steuern, nativer Shop statt WooCommerce-Entwurf, Stripe/PayPal live, Marketing-Texte de/en/es/it.

- Geräte-Shop auf tolinax.com ecommerce.device-shop-tolinax (6a97e5b0f7ba5c757954400c): Kategorie Geräte, Shop-Seite/Navigation, Produktseiten mit 3D-Viewer, Varianten (Control Panel 7/10), Preise, Lieferzeit, Versandzonen/-klassen, Steuern, Marketing-Texte.

- [Feature/critical/REP3] R6: Geräte-Shop auf tolinax.com — Kategorie Geräte, Shop-Seite + Navigation, Produktseiten mit 3D-Viewer und Varianten (Control Panel 7/10), Preise, Lieferzeit, Versandzonen/-klassen, Steuern, Marketing-Texte de/en/es/it (6a97e68df7ba5c7579545f41)

- Nativer tolinax-Shop & Checkout live ecommerce.native-shop-checkout (6a97e5b2f7ba5c7579544090): Shops-Entität tolinax nativ (statt WooCommerce-Entwurf), Stripe/PayPal live, AGB/Widerruf, Bestellbestätigung.

- [Aufgabe/high/REP3] R6: Nativer tolinax-Shop (Shops-Entität, WooCommerce-Entwurf ablösen), Stripe/PayPal live, AGB/Widerruf, Bestellbestätigung (6a97e692f7ba5c7579545fd4)

Auftragskette Ende-zu-Ende mit Benachrichtigungen (6a97e553f7ba5c75795429c3, R6, Modul sale)

Bestellung !' Auftrag !' Produktionsauftrag (make-to-order / make-to-stock je Produkt über procurement) !' Stücklisten-Auflösung !' interner Produktionsauftrag / interne Materialbestellung / externe Lieferantenbestellung !' Pickauftrag !' Kommission an Arbeitsplatz !' Produktion !' Abnahme mit Serie !' Verpackung !' Versand mit Tracking !' Rechnung; Benachrichtigungs-Policies für Kunde und Team je Stufe; Businessplanungs-Szenario Eigenfertigung.

- Bestellung !' Auftrag !' Produktionsauftrag sale.order-to-production-chain (6a97e5b3f7ba5c7579544114): Shop-Order !' Assignment !' ProductionOrder.assignment automatisch (make-to-order) bzw. Lagerabgang (make-to-stock) je Produkt über procurement; Messung der heutigen Order!Assignment-Strecke, ggf. IOrder._assignment.

- [Feature/critical/REP3] R6: Bestellung !' Auftrag !' Produktionsauftrag — Shop-Order !' Assignment !' ProductionOrder.assignment automatisch (make-to-order) bzw. Lagerabgang (make-to-stock) je Produkt über procurement (6a97e696f7ba5c757954606d)

- Stücklisten-Auflösung & Bedarfe purchase.bom-explosion-demand (6a97e5b7f7ba5c757954419e): BOM-Explosion je Produktionsauftrag !' Bedarfe !' interner Produktionsauftrag (Gehäuse, internalProductionOrder) / interne Materialbestellung / externe Lieferantenbestellung.

- [Feature/high/REP4] R6: Stücklisten-Auflösung je Produktionsauftrag !' Bedarfe !' interner Produktionsauftrag (Gehäuse) / interne Materialbestellung / externe Lieferantenbestellung

(6a97e69bf7ba5c7579546108)

- Pick !' Kommission !' Produktion !' Abnahme !' Versand !' Rechnung warehouse.pick-commission-ship (6a97e5bdf7ba5c75795442dc): Pickauftrag, Kommission an Arbeitsplatz, Produktion, Abnahme mit Serie, Packbestätigung, Versand mit Tracking (Shipping), Rechnung.
- [Feature/high/REP4] R6: Pickauftrag !' Kommission an Arbeitsplatz !' Produktion !' Abnahme mit Serie !' Packbestätigung !' Versand mit Tracking !' Rechnung (6a97e6a1f7ba5c757954622a)
- Benachrichtigungen der Auftragskette automation.order-chain-notifications (6a97e5bff7ba5c7579544360): NotificationPolicies je Stufe: Kunde (Bestätigung, in Fertigung, versendet, Rechnung), intern (neuer Auftrag, Material fehlt, Pick bereit, QC rot, Abnahme offen) über Bell/Mail/Chat/Wallboard, de/en/es/it.
- [Feature/high/REP4] R6: Benachrichtigungs-Policies der Auftragskette — Kunde (Bestätigung, in Fertigung, versendet, Rechnung) und intern (neuer Auftrag, Material fehlt, Pick bereit, QC rot, Abnahme offen) über Bell/Mail/Chat/Wallboard, de/en/es/it (6a97e6a6f7ba5c75795462be)
- Businessplanung Eigenfertigung Geräte planning.device-business-plan (6a97e5c3f7ba5c75795443eb): PlanningModel mit Dimensionen Produkt × Monat: Absatz × Preis, Materialkosten aus Rollup, Kapazität Arbeitsplätze/Drucker, Szenarien; Energiekosten aus Messung.
- [Aufgabe/medium/REP4] R6: Businessplanung Eigenfertigung — PlanningModel Produkt × Monat mit Absatz × Preis, Materialkosten aus Rollup, Kapazität Arbeitsplätze/Drucker, Szenarien, Energiekosten aus Messung (6a97e6abf7ba5c7579546350)

Think-Tank-Fertigungsservice: Kunden bestellen 3D-Druck, Laserschnitt und Gravur (6a97e557f7ba5c7579542ab2, R6, Modul cms)

CMS-Block fabrication-order auf tolinax.com: Modell-Upload (STL/3MF bzw. DXF/SVG), Verfahren (FDM-Druck H2S/Bibo 2, CO2-Laser K40 Schnitt, Dioden-Laser Sculpfun Gravur), Material, Sofort-Kalkulation je Verfahren, Bestellung in den Shop, Produktionsauftrag mit Maschine. Lokale Kunden im Tegernseer Tal; tolinax als erster Kunde. Zusammen mit E9 (Print-on-Demand, verschoben).

- Website-Block Fertigungsauftrag (Upload, Verfahren, Kalkulation, Bestellung) cms.fabrication-order-block (6a97e5c7f7ba5c7579544476): CMS-Blocktyp fabrication-order: Datei-Upload (STL/3MF, DXF/SVG), Verfahren FDM-Druck / Laserschnitt / Lasergravur, Material, Sofort-Kalkulation, Bestellung in den Shop-Warenkorb; consumer-app-Renderer.
- [Feature/critical/REP3] R6: Website-Block fabrication-order für tolinax.com — Upload (STL/3MF, DXF/SVG), Verfahren FDM-Druck / Laserschnitt / Lasergravur, Material, Sofort-Kalkulation, Bestellung in den Warenkorb; consumer-app-Renderer (6a97e6b0f7ba5c75795463e3)
- [QS/critical/REP4] R6-Abnahme: Testkauf eines Geräts im Shop bis Versand und Rechnung + je ein Kunden-Auftrag über das Widget für 3D-Druck, Laserschnitt und Gravur (tolinax als Kunde) (6a97e6c9f7ba5c7579546740)
- Sofort-Kalkulation je Verfahren production.fabrication-quote-calculator (6a97e5c9f7ba5c75795444fa): Druck: Volumen/Gewicht/Zeit-Schätzung aus STL (Slicer-Schätzung) × Filamentpreis + Maschinenstunde + Energie; Laser: Schnittlänge/Gravurfläche aus DXF/SVG × Zeit × Satz; Ergebnis als Offer/Position.
- [Feature/critical/REP3] R6: Sofort-Kalkulation je Verfahren — Druck: Volumen/Gewicht/Zeit aus STL × Filamentpreis + Maschinenstunde + Energie; Laser: Schnittlänge/Gravurfläche aus DXF/SVG × Zeit × Satz; Ergebnis als Angebot/Position (6a97e6b5f7ba5c757954647c)

- Produktionsschemas FDM-Druck, Laserschnitt, Lasergravur config.production-schemas-laser (6a97e5d0f7ba5c75795445b5): Drei weitere Produktionsschemas mit Arbeitsplänen, QC (Maßhaltigkeit, Kantenqualität) und Costing je Verfahren; Kunden-Auftrag !' Produktionsauftrag mit Maschine (H2S/Bibo 2, K40, Sculpfun).

- [Feature/high/REP3] R6: Produktionsschemas FDM-Druck, Laserschnitt, Lasergravur mit Arbeitsplänen, QC (Maßhaltigkeit, Kantenqualität) und Costing je Verfahren; Kunden-Auftrag !' Produktionsauftrag mit Maschine (H2S/Bibo 2, K40, Sculpfun) (6a97e6baf7ba5c757954650f)

Produktionsflächen-Wallboards (VGT-Vorlage) & Demo-Mandant Fertigungsbetrieb (6a97e55bf7ba5c7579542bca, R7, Modul reporting)

Die drei verwaisten Fabrik-Wallboards (Management, Aufbereitung, Gießerei+Stampferei, Mandant 5d7fa3c4...) als Vorlagen übernehmen; Produktionsflächen-Wallboards je Halle mit Arbeitsplatz-Bindung; Demo-Mandant generischer Fertigungsbetrieb anonymisiert aus R1–R6 über den Demo-Fabrik-Seeder (M-F); Superuser-Demo-Verwaltung.

- Produktionsflächen-Wallboards nach VGT-Vorlage reporting.production-floor-wallboards (6a97e5d7f7ba5c7579544689): Die drei verwaisten Wallboards exportieren und als Vorlagen übernehmen; Hallen-Wallboards mit Produktionsschemas, Auftragsstatus, OEE, Störungen, Andon, Arbeitsplatz-Bindung.

- [Aufgabe/high/REP3] R7: Die drei verwaisten VGT-Wallboards (Management, Aufbereitung, Gießerei+Stampferei, Mandant 5d7fa3c4...) exportieren und als Vorlagen übernehmen (6a97e6cef7ba5c75795467f8)

- [Feature/high/REP4] R7: Produktionsflächen-Wallboards je Halle — Produktionsschemas, Auftragsstatus, OEE, Störungen, Andon, Arbeitsplatz-Bindung (_workcenter) (6a97e6d1f7ba5c7579546891)

- Demo-Mandant generischer Fertigungsbetrieb su.demo-tenant-manufacturing (6a97e5daf7ba5c757954470d): Anonymisiert aus R1–R6: Hallen (Gießerei, Aufbereitung, Montage), Maschinen, Arbeitsplätze, Schemas/Arbeitspläne, Aufträge in allen Status, Lager-Raster, Demo-Energie, Businessplan — über den Demo-Fabrik-Seeder; Superuser-Demo-Zugang, Reset, Webshots.

- [Feature/critical/REP4] R7: Demo-Mandant generischer Fertigungsbetrieb — Hallen (Gießerei, Aufbereitung, Montage), Maschinen, Arbeitsplätze, Schemas/Arbeitspläne aus R2 abgeleitet, Aufträge in allen Status, Lager-Raster, Demo-Energie, Businessplan; anonymisiert über den Demo-Fabrik-Seeder (6a97e6d6f7ba5c7579546925)

- [QS/high/REP4] R7-Abnahme: Demo-Mandant mit drei Hallen-Wallboards live, Superuser-Zugang, Reset funktioniert (6a97e6dcf7ba5c75795469b8)

Serie & Anbindungen: JLC3DP, eigene Shelly-Integration, Lieferanten-APIs (6a97e55cf7ba5c7579542c4d, R8, Modul purchase)

JLC3DP-API (Antrag, Quote/Order/Status) als Plattform-Anbindung für die Serienfertigung der Gehäuse, eigene offline-/self-host-fähige Shelly-Integration (Gen2 RPC/MQTT lokal !' MCU-Ingest-Bridge), Bambu-Restglieder und weitere Lieferanten-APIs (M-E).

- JLC3DP-API-Anbindung für Serien-Gehäuse purchase.jlc3dp-integration (6a97e5def7ba5c757954482d): API-Antrag, Quote/Order/Status über api-hooks/Purchase, Serienbestellung aus Bedarf; manuelle SupplierOrder als Fallback.

- [Feature/high/REP5] R8: JLC3DP-API — Antrag stellen, Quote/Order/Status als Plattform-Anbindung über api-hooks/Purchase, Serienbestellung der Gehäuse aus Bedarf; manuelle SupplierOrder als Fallback (6a97e6e5f7ba5c7579546b05)
 - Eigene Shelly-Integration offline/self-host `iot.shelly-native-integration` (6a97e5e1f7ba5c75795448b2): Shelly Gen2 RPC/MQTT lokal !' MCU-Ingest-Bridge, HA-unabhängig, Kill-Switch, Mandanten-Zuordnung.
 - [Feature/high/REP5] R8: Eigene Shelly-Integration offline/self-host — Gen2 RPC/MQTT lokal !' MCU-Ingest-Bridge, HA-unabhängig, Kill-Switch, Mandanten-Zuordnung (6a97e6e9f7ba5c7579546b91)
 - Weitere Lieferanten-Anbindungen `purchase.supplier-api-connectors` (6a97e5e5f7ba5c757954493e): Bambu-Restglieder (E1/E2 falls offen), Preis-/Bestell-Anbindungen (Reichelt/Mouser) im Anschluss an M-E.
 - [Aufgabe/medium/REP6] R8: Weitere Lieferanten-Anbindungen — Bambu-Restglieder (E1/E2 falls offen), Preis-/Bestell-Anbindungen Reichelt/Mouser im Anschluss an M-E (6a97e6edf7ba5c7579546c24)

Umgehängte FAB-Epics

- E3 Drucker-Fundament !' R3 · E8 Produktionsintegration !' R5 · E9 3D-Marktplatz & Print-on-Demand !' R6 · E10 Druck & Ausgabe als Plattform-Dienst !' R4 (Marktplatz-Anteil von E9 bleibt M5-Inhalt).

Abhängigkeiten

- H2S-Lieferung (R3/R5) · Bambu-Bridge E1/E2 bleibt FAB M1 (R3 baut HA/Shelly unabhängig davon) · Webflasher W3b (blocked P1) für den Sensor-Flash-Schritt · M-B LAGER1 Dymo-Drucker · Foundation-Slot für die gebündelte Kaskade (REP1) · Demo-Fabrik-Seeder (M-F) für R7.

Arbeitsweise

- Jeder Task wird ein Brief mit Task-ID (Briefe-1:1-PM-Regel); Fundament-Adds nur über die eine Kaskade; jede Nullmessung mit Positiv-Kontrolle; Verträge und Schemas als Typ/Feldliste, nie Casts.

Hinweise zur Quelle

Dieses Dokument stammt aus der Dokumentation von tolinax UG und wurde am 2026-09-15 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