

Inhaltsverzeichnis

Template API v1 Referenz	2
Versionen und unterstützter Umfang	2
Manifest	2
ZIP-Paket	3
ZIP-Import und Revisionsidentität	3
Unveränderlicher Speicher, Sicherung und Client-Modus	4
Import, Client-Modus und Wiederherstellung	4
Kompatibilität	4
Präsentationswurzeln	4
common	4
document	10
timesheet	13
settings	15
format	16
components	17
Anwendungseigene Komponenten	20
financial-summary	20
deductions	20
reminder-settlement	21
payment	21
legal-disclosures	21
timesheet-summary	21
Pebble und Ressourcen	21
CSS und XHTML	22
Diagnosen	22
Verantwortungsgrenze	23

Template API v1 Referenz

Diese Datei wird aus `TemplateApiV1Catalog` generiert. Führe zum Aktualisieren `./gradlew :app:generateTemplateApiDocumentation` aus.

Template API v1 ist experimentell und noch kein stabiler Kompatibilitätsvertrag. Sie kann sich zwischen Anwendungsversionen weiterentwickeln; Paketautoren müssen ihre Pakete für jede Version erneut testen.

Beginne mit der Anleitung für Verkaufsdokumente, der Anleitung für Zeitrachweise oder den professionellen Starterpaketen.

Versionen und unterstützter Umfang

Vertrag	Version
Paketformat	1
Template API	1

Vorlagenarten: `sales_document`, `timesheet`.

Verkaufsdokumentfamilien: `quote`, `order_confirmation`, `delivery_note`, `invoice`, `invoice_correction`, `payment_reminder`.

Berücksichtigte Einstellungsgruppen: `page.margins`, `fold_marks`, `header`, `footer`, `banner`, `metadata`, `text_blocks`, `tenant_address`, `table`.

Manifest

Der Paketstamm enthält `manifest.json`; validiere die Datei gegen `manifest-v1.schema.json`. Manifestfehler verwenden die Kompatibilitäts- und Ressourcendiagnosen.

Stabile Werte sind vollständige Vertragswerte. Beispiele veranschaulichen die Form der aktuellen Engine und sind keine zusätzlichen Einschränkungen.

`packageFormatVersion`

- **Vertrag:** Typ `integer`; erforderlich: ja
- **Stabile Werte:** 1
- **Beschreibung:** Hauptversion des ZIP-Paketformats.

`packageId`

- **Vertrag:** Typ `string`; erforderlich: ja
- **Beispielwerte:** `com.example.invoice`
- **Beschreibung:** Stabiler Reverse-DNS-Bezeichner für eine Designlinie.

`displayName`

- **Vertrag:** Typ `string`; erforderlich: ja
- **Beispielwerte:** `Example sales documents`
- **Beschreibung:** Für Menschen lesbarer Paketname.

`templateKind`

- **Vertrag:** Typ `string`; erforderlich: ja
- **Stabile Werte:** `sales_document`, `timesheet`
- **Beschreibung:** Die einzelne Vorlagenart des Pakets.

`entryPoint`

- **Vertrag:** Typ `string`; erforderlich: ja
- **Beispielwerte:** `document.peb`
- **Beschreibung:** Normalisierter paketrelativer Pebble-Einstiegspunkt.

`templateApiVersion`

- **Vertrag:** Typ `integer`; erforderlich: ja
- **Stabile Werte:** 1
- **Beschreibung:** Hauptversion des Vorlagenpräsentationsvertrags.

supportedDocumentFamilies

- **Vertrag:** Typ list; erforderlich: ja
- **Stabile Werte:** quote, order_confirmation, delivery_note, invoice, invoice_correction, payment_reminder
- **Beschreibung:** Verkaufsdokumentfamilien, die das Paket laut Deklaration rendern kann.

honoredSettings

- **Vertrag:** Typ list; erforderlich: ja
- **Stabile Werte:** page.margins, fold_marks, header, footer, banner, metadata, text_blocks, tenant_address, table
- **Beschreibung:** Layouteinstellungen, die das Paket tatsächlich verwendet.

resources

- **Vertrag:** Typ object; erforderlich: ja
- **Beispielwerte:** {"styles":["styles.css"],"images":[]}
- **Beschreibung:** Deklarierte paketrelative Stile, PNG-/JPEG-Bilder und statische TrueType-Schriftvarianten der unveränderlichen Revision.

ZIP-Paket

Autoren arbeiten in einem gewöhnlichen Quellverzeichnis und packen dessen Inhalt anschließend für den Import als ZIP. Ein ZIP steht für ein Design, eine Vorlagenart und einen Einstiegspunkt. `manifest.json` muss im Stamm des ZIPs liegen und nicht in einem umgebenden Verzeichnis. Ein typisches ZIP für Verkaufsdokumente enthält:

```
manifest.json
document.peb
partials/
styles.css
images/
fonts/
```

Die eingepackten Lernbeispiele und professionellen Starter sind Quellverzeichnisse, deren Inhalt aus ihrem Stamm heraus gepackt werden kann. Die Starter enthalten deklarierte paketlokale Bilder und statische TrueType-Varianten. Der Kunden-ZIP-Import validiert deklarierte Ressourcen und installiert die vollständige kanonische Revision im unveränderlichen Speicher.

ZIP-Import und Revisionsidentität

Containerregeln:

- Das Importartefakt ist eine ZIP-Datei mit `manifest.json` im Stamm; ein umgebendes Verzeichnis ist nicht erlaubt.
- Die Anwendung validiert und entpackt ein ZIP vor der Verwendung in den unveränderlichen Anwendungsdatenspeicher; Vorlagen werden niemals direkt aus dem ZIP gerendert.
- Eintragsreihenfolge, Komprimierungsmethode, Zeitstempel, Berechtigungen und andere ZIP-Metadaten sind nicht Teil der Revisionsidentität.
- Verzeichniseinträge mit null Bytes und macOS-Metadaten unter `__MACOSX`, mit dem Namen `.DS_Store` oder mit dem Präfix `.` werden nach einer begrenzten Pfadvalidierung ignoriert; andere Dateien unter gewöhnlichen Verzeichniseinträgen unterliegen weiterhin der normalen Paketvalidierung.
- Absolute Pfade, übergeordnete Pfadsegmente, Backslashes, symbolische Links, doppelte Einträge, nicht portable Pfade sowie Kollisionen durch Unicode-Normalisierung oder Groß-/Kleinschreibung werden abgelehnt.
- Der ZIP-Import ist auf 5242880 komprimierte Bytes, 20971520 entpackte Bytes, 128 Einträge, 4194304 Bytes je Datei, 8 Pfadsegmente und ein Komprimierungsverhältnis von 100:1 je Datei begrenzt.

Regeln der kanonischen Revision:

- Die Revision ist der kleingeschriebene SHA-256-Wert der validierten logischen Paketinhalte, nicht der rohen ZIP-Bytes.
- `manifest.json` wird geparkt, normalisiert und als kompaktes kanonisches JSON serialisiert; jede andere Datei trägt ihre exakten Bytes ohne Zeilenumbruch- oder Textnormalisierung bei.
- Pfade werden auf Unicode NFC normalisiert, verwenden Schrägstriche und werden nach ihrem kanonischen UTF-8-Pfad sortiert.

- Der Hashstrom beginnt mit der ASCII-Domänenmarke AGYNAMIX-INVOICER-TEMPLATE-PACKAGE, gefolgt von einem NUL-Byte sowie Paketformatversion und Dateianzahl als vorzeichenbehaftete Big-Endian-Werte.
- Jede sortierte Datei trägt eine vorzeichenbehaftete 32-Bit-Big-Endian-Länge des UTF-8-Pfads, die Pfadbytes, eine vorzeichenbehaftete 64-Bit-Big-Endian-Inhaltslänge und die exakten Inhaltsbytes bei.

Unveränderlicher Speicher, Sicherung und Client-Modus

- Validierte kanonische Dateien werden atomar im unveränderlichen, inhaltsadressierten Anwendungsdatenspeicher installiert; eine abgeschlossene Revision wird niemals direkt geändert.
- Revisionsmetadaten und Datensatzberechtigungen liegen in der aktiven Datenbank, während Paketbytes außerhalb der Invoicer-Datenbanksicherungsarchive verbleiben.
- Betreiber müssen den Paketspeicher der Anwendungsdaten durch Dateisystem- oder Systemsicherungen schützen, wenn die Notfallwiederherstellung Paketartefakte benötigt.
- Ein Reimport mit exakt gleichem Inhalt erzeugt denselben Revisionshash; fehlende oder beschädigte Bytes schlagen ausdrücklich fehl und wählen niemals automatisch eine andere Revision.
- Im Client-Modus bleibt der Host maßgeblich. Clients laden exakte kanonische Paketbytes mit Hashprüfung in einen lokalen, nach Datensatz und Generation abgegrenzten Vorschau-cache; endgültige Exporte werden auf dem Host gerendert.

Import, Client-Modus und Wiederherstellung

- Die Prüfung validiert das begrenzte ZIP und rendert vor der Installation jede anwendbare Vorschauseite. Sie listet akzeptierte paketlokale Schriftvarianten auf. Die Installation erfordert eine ausdrückliche Bestätigung für ein vom Kunden bereitgestelltes Layout; bei Paketen mit Schriften bestätigt sie zusätzlich die Rechte zur Einbettung und Weitergabe.
- Im Standalone-Modus wird die validierte kanonische Revision im unveränderlichen Paketspeicher des aktiven Datensatzes installiert. Anschließend wird das ausgewählte Layout an die exakte Paket-ID und den Revisionshash gebunden.
- Im Client-Modus erfolgt zunächst die lokale Prüfung. Danach wird dasselbe ZIP zur maßgeblichen Validierung und Installation an den Host übertragen. Layoutänderungen bleiben Sache des Hosts.
- Client-Vorschauen laden exakte, hashverifizierte Paketbytes in einen Cache, der nach Host-Datensatz-ID, Generations-ID und Revisionshash abgegrenzt ist. Eine exakt gecachte Revision bleibt offline verwendbar; eine nicht gecachte oder beschädigte Revision schlägt ausdrücklich fehl.
- Endgültige Exporte im Client-Modus werden auf dem Host aus der dort maßgeblichen Revision gerendert.
- Veröffentlichte Verkaufsdokumente und eingefrorene Zeitnachweiszeiträume behalten ihren vorgesehenen exakten Verweis. Fehlende Bytes wählen niemals automatisch eine andere Revision.
- Ein berechtigter Benutzer kann eine deutlich gewarnte Ersatzwiederherstellung ausdrücklich genehmigen. Rendering und Export bewahren die Herkunft von vorgesehenen und tatsächlich verwendeter Revision; der ursprüngliche eingefrorene Verweis wird nicht umgeschrieben.

Kompatibilität

- Unbekannte oder neuere Hauptversionen von Paketformat oder Template API werden vor dem Rendering abgelehnt.
- Eine ältere Hauptversion wird nur akzeptiert, solange ein ausdrücklicher Kompatibilitätsadapter und Vertragstests vorhanden sind.
- Ein Paket muss seine Vorlagenart und jede unterstützte Verkaufsdokumentfamilie deklarieren.
- Kompatibilitätsfehler sind ausdrücklich und führen niemals zu einem Rückfall auf ein anderes Paket.
- Template API v1 ist experimentell und noch kein stabiler Kompatibilitätsvertrag; Autoren müssen Pakete mit jeder Anwendungsversion erneut testen.

Präsentationswurzeln

Alle sechs Wurzelschlüssel sind immer vorhanden. Genau einer von `document` und `timesheet` ist nicht null. Anzeigewerte sind maßgeblich; Vorlagen dürfen finanzielle, Lebenszyklus-, Lokalisierungs- oder Rechtsentscheidungen nicht rekonstruieren.

Stabile Werte sind vollständige Vertragswerte. Wo keine vorhanden sind, enthält jeder Feldeintrag ein veranschaulichendes Beispiel der aktuellen Engine-Ausgabe.

common

Vorbereitete Parteien, Seite, Kopfzeile, Fußzeile, Beschriftungen und Anwendungsressourcen.

`common.locale`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: de-DE

`common.tenant.displayName`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: Example GmbH

`common.tenant.qualifiedContactPerson`

- Vertrag: Typ string; Vorkommen nullable; Semantik display
- Beispielwerte: Erika Muster

`common.tenant.addressLines[]`

- Vertrag: Typ string; Vorkommen empty_list; Semantik display
- Beispielwerte: Musterstraße 1

`common.tenant.isCompany`

- Vertrag: Typ boolean; Vorkommen required; Semantik display
- Beispielwerte: true

`common.tenant.isSmallBusiness`

- Vertrag: Typ boolean; Vorkommen required; Semantik display
- Beispielwerte: true

`common.tenant.customerNumber`

- Vertrag: Typ string; Vorkommen nullable; Semantik display
- Beispielwerte: C-1001

`common.tenant.vatId`

- Vertrag: Typ string; Vorkommen nullable; Semantik display
- Beispielwerte: DE123456789

`common.tenant.taxId`

- Vertrag: Typ string; Vorkommen nullable; Semantik display
- Beispielwerte: 12/345/67890

`common.recipient.displayName`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: Example GmbH

`common.recipient.qualifiedContactPerson`

- Vertrag: Typ string; Vorkommen nullable; Semantik display
- Beispielwerte: Erika Muster

`common.recipient.addressLines[]`

- Vertrag: Typ string; Vorkommen empty_list; Semantik display
- Beispielwerte: Musterstraße 1

`common.recipient.isCompany`

- Vertrag: Typ boolean; Vorkommen required; Semantik display
- Beispielwerte: true

`common.recipient.isSmallBusiness`

- Vertrag: Typ boolean; Vorkommen required; Semantik display
- Beispielwerte: true

`common.recipient.customerNumber`

- Vertrag: Typ string; Vorkommen nullable; Semantik display
- Beispielwerte: C-1001

`common.recipient.vatId`

- Vertrag: Typ string; Vorkommen nullable; Semantik display
- Beispielwerte: DE123456789

`common.recipient.taxId`

- Vertrag: Typ string; Vorkommen nullable; Semantik display
- Beispielwerte: 12/345/67890

`common.header.showLogo`

- Vertrag: Typ boolean; Vorkommen required; Semantik display
- Beispielwerte: true

`common.header.logoPlacement`

- Vertrag: Typ string; Vorkommen required; Semantik stable_enum
- Stabile Werte: left, center, right

`common.header.contentAlignment`

- Vertrag: Typ string; Vorkommen required; Semantik stable_enum
- Stabile Werte: left, center, right

`common.header.showCompanyName`

- Vertrag: Typ boolean; Vorkommen required; Semantik display
- Beispielwerte: true

`common.header.showDocumentType`

- Vertrag: Typ boolean; Vorkommen required; Semantik display
- Beispielwerte: true

`common.header.showDocumentTypeInMetadata`

- Vertrag: Typ boolean; Vorkommen required; Semantik display
- Beispielwerte: true

`common.header.leftColumnPercent`

- Vertrag: Typ integer; Vorkommen required; Semantik display
- Beispielwerte: 50

`common.header.rightColumnPercent`

- Vertrag: Typ integer; Vorkommen required; Semantik display
- Beispielwerte: 50

`common.footer.columns[].position`

- Vertrag: Typ string; Vorkommen empty_list; Semantik stable_enum
- Stabile Werte: left, center, right

`common.footer.columns[].widthPercent`

- Vertrag: Typ integer; Vorkommen empty_list; Semantik display
- Beispielwerte: 50

`common.footer.columns[].heading`

- Vertrag: Typ string; Vorkommen nullable; Semantik display
- Beispielwerte: Total

`common.footer.columns[].headingIconXhtml`

- Vertrag: Typ string; Vorkommen nullable; Semantik safe_xhtml
- Beispielwerte: <p>Prepared text</p>

`common.footer.columns[].rows[].fieldId`

- Vertrag: Typ string; Vorkommen empty_list; Semantik stable_enum
- Beispielwerte: general_email

`common.footer.columns[].rows[].label`

- Vertrag: Typ string; Vorkommen nullable; Semantik display
- Beispielwerte: Total

`common.footer.columns[].rows[].values[]`

- Vertrag: Typ string; Vorkommen empty_list; Semantik display
- Beispielwerte: Prepared display value

`common.footer.columns[].rows[].separator`

- Vertrag: Typ string; Vorkommen empty_list; Semantik display
- Beispielwerte: ·

`common.footer.columns[].rows[].labelStrong`

- Vertrag: Typ boolean; Vorkommen empty_list; Semantik display
- Beispielwerte: true

`common.footer.columns[].rows[].valueStrong`

- Vertrag: Typ boolean; Vorkommen empty_list; Semantik display
- Beispielwerte: true

`common.footer.columns[].rows[].iconXhtml`

- Vertrag: Typ string; Vorkommen nullable; Semantik safe_xhtml
- Beispielwerte: <p>Prepared text</p>

`common.footer.adaptiveLayout`

- Vertrag: Typ boolean; Vorkommen required; Semantik display
- Beispielwerte: true

`common.footer.splitLabelAndValue`

- Vertrag: Typ boolean; Vorkommen required; Semantik display
- Beispielwerte: true

`common.footer.showPageNumber`

- Vertrag: Typ boolean; Vorkommen required; Semantik display
- Beispielwerte: true

`common.footer.showTotalPages`

- Vertrag: Typ boolean; Vorkommen required; Semantik display
- Beispielwerte: true

`common.footer.pageLabel`

- Vertrag: Typ string; Vorkommen nullable; Semantik display
- Beispielwerte: Page

`common.footer.totalSeparator`

- Vertrag: Typ string; Vorkommen nullable; Semantik display
- Beispielwerte: of

`common.footer.pageIconXhtml`

- Vertrag: Typ string; Vorkommen nullable; Semantik safe_xhtml
- Beispielwerte: <p>Prepared text</p>

`common.footer.totalIconXhtml`

- Vertrag: Typ string; Vorkommen nullable; Semantik safe_xhtml
- Beispielwerte: <p>Prepared text</p>

`common.footer.reservedBottomMm`

- Vertrag: Typ integer; Vorkommen required; Semantik display
- Beispielwerte: 20

`common.footer.typography.fontSizeCss`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: 8pt

`common.footer.typography.iconSizeCss`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: 10pt

`common.footer.typography.iconGapCss`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: 4px

`common.footer.typography.bodyLineHeight`

- Vertrag: Typ decimal; Vorkommen required; Semantik display
- Beispielwerte: 1.2

`common.footer.typography.headingLineHeight`

- Vertrag: Typ decimal; Vorkommen required; Semantik display
- Beispielwerte: 1.2

`common.footer.typography.headingMarginBottomPx`

- Vertrag: Typ integer; Vorkommen required; Semantik display
- Beispielwerte: 20

`common.footer.colors.label`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: #2d3748

`common.footer.colors.value`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: #2d3748

`common.footer.colors.heading`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: #2d3748

`common.page.size`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: A4

`common.page.marginTopMm`

- Vertrag: Typ integer; Vorkommen required; Semantik display
- Beispielwerte: 20

`common.page.marginRightMm`

- Vertrag: Typ integer; Vorkommen required; Semantik display
- Beispielwerte: 20

`common.page.marginBottomMm`

- Vertrag: Typ integer; Vorkommen required; Semantik display
- Beispielwerte: 20

`common.page.marginLeftMm`

- Vertrag: Typ integer; Vorkommen required; Semantik display
- Beispielwerte: 20

`common.page.foldMarks`

- Vertrag: Typ boolean; Vorkommen required; Semantik display
- Beispielwerte: true

`common.resources.tenantLogo`

- Vertrag: Typ string; Vorkommen nullable; Semantik resource_handle
- Beispielwerte: data:image/png;base64,...

`common.resources.fontCss`

- Vertrag: Typ string; Vorkommen required; Semantik safe_css
- Beispielwerte: @font-face { ... }

`common.labels.from`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: Total

`common.labels.customerNumber`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: C-1001

`common.labels.documentNumber`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: INV-2026-001

`common.labels.documentDate`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: 2026-08-20

`common.labels.deliveryDate`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: 2026-08-20

`common.labels.actionByDate`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: 2026-08-20

`common.labels.orderNumber`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: INV-2026-001

`common.labels.contractNumber`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: INV-2026-001

`common.labels.buyerReference`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: Total

`common.labels.originalInvoice`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: Total

`common.labels.period`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: Total

`common.labels.project`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: Total

`common.labels.position`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: Total

`common.labels.description`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: Total

`common.labels.quantity`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: Total

`common.labels.unitPrice`

- **Vertrag:** Typ string; Vorkommen required; Semantik display
- **Beispielwerte:** 1,234.56 €

`common.labels.vat`

- **Vertrag:** Typ string; Vorkommen required; Semantik display
- **Beispielwerte:** Total

`common.labels.total`

- **Vertrag:** Typ string; Vorkommen required; Semantik display
- **Beispielwerte:** Total

`common.labels.date`

- **Vertrag:** Typ string; Vorkommen required; Semantik display
- **Beispielwerte:** 2026-08-20

`common.labels.time`

- **Vertrag:** Typ string; Vorkommen required; Semantik display
- **Beispielwerte:** Total

`common.labels.location`

- **Vertrag:** Typ string; Vorkommen required; Semantik display
- **Beispielwerte:** Berlin

`common.labels.duration`

- **Vertrag:** Typ string; Vorkommen required; Semantik display
- **Beispielwerte:** 3:30 h

`common.labels.alternatives`

- **Vertrag:** Typ string; Vorkommen required; Semantik display
- **Beispielwerte:** Total

`common.labels.option`

- **Vertrag:** Typ string; Vorkommen required; Semantik display
- **Beispielwerte:** Total

`common.labels.subtotal`

- **Vertrag:** Typ string; Vorkommen required; Semantik display
- **Beispielwerte:** 1,234.56 €

`common.sandboxWatermark`

- **Vertrag:** Typ boolean; Vorkommen required; Semantik display
- **Beispielwerte:** true

`document`

Vorbereitete Identität, Metadaten, Positionen, Alternativen und Texte des Verkaufsdocuments. Diese artspezifische Wurzel ist für die jeweils andere Vorlagenart null.

`document.family`

- **Vertrag:** Typ string; Vorkommen required; Semantik `stable_enum`
- **Stabile Werte:** `quote`, `order_confirmation`, `delivery_note`, `invoice`, `invoice_correction`, `payment_reminder`

`document.title`

- **Vertrag:** Typ string; Vorkommen required; Semantik display
- **Beispielwerte:** Invoice

`document.number`

- **Vertrag:** Typ string; Vorkommen required; Semantik display
- **Beispielwerte:** INV-2026-001

`document.date`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: 2026-08-20

`document.actionByDate`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: 2026-08-20

`document.deliveryDate`

- Vertrag: Typ string; Vorkommen nullable; Semantik display
- Beispielwerte: 2026-08-20

`document.descriptionXhtml`

- Vertrag: Typ string; Vorkommen nullable; Semantik safe_xhtml
- Beispielwerte: <p>Prepared text</p>

`document.parentNumber`

- Vertrag: Typ string; Vorkommen nullable; Semantik display
- Beispielwerte: INV-2026-001

`document.originalInvoiceNumber`

- Vertrag: Typ string; Vorkommen nullable; Semantik display
- Beispielwerte: INV-2026-001

`document.metadata[].id`

- Vertrag: Typ string; Vorkommen empty_list; Semantik display
- Beispielwerte: net_total

`document.metadata[].label`

- Vertrag: Typ string; Vorkommen empty_list; Semantik display
- Beispielwerte: Total

`document.metadata[].value`

- Vertrag: Typ string; Vorkommen empty_list; Semantik display
- Beispielwerte: Prepared display value

`document.showVatRateColumn`

- Vertrag: Typ boolean; Vorkommen required; Semantik display
- Beispielwerte: true

`document.items[].position`

- Vertrag: Typ integer; Vorkommen empty_list; Semantik display
- Beispielwerte: 1

`document.items[].displayPosition`

- Vertrag: Typ integer; Vorkommen nullable; Semantik display
- Beispielwerte: 20

`document.items[].descriptionXhtml`

- Vertrag: Typ string; Vorkommen empty_list; Semantik safe_xhtml
- Beispielwerte: <p>Prepared text</p>

`document.items[].kind`

- Vertrag: Typ string; Vorkommen empty_list; Semantik stable_enum
- Stabile Werte: priced, text

`document.items[].productSku`

- Vertrag: Typ string; Vorkommen nullable; Semantik display
- Beispielwerte: SKU-1001

`document.items[].productName`

- Vertrag: Typ string; Vorkommen nullable; Semantik display
- Beispielwerte: Consulting package

`document.items[].productImage`

- Vertrag: Typ string; Vorkommen nullable; Semantik resource_handle
- Beispielwerte: data:image/png;base64,...

`document.items[].productImageMaxWidthMm`

- Vertrag: Typ integer; Vorkommen empty_list; Semantik display
- Beispielwerte: 20

`document.items[].productImageMaxHeightMm`

- Vertrag: Typ integer; Vorkommen empty_list; Semantik display
- Beispielwerte: 20

`document.items[].quantityWithUnit`

- Vertrag: Typ string; Vorkommen empty_list; Semantik display
- Beispielwerte: 2.00 h

`document.items[].unitPrice`

- Vertrag: Typ string; Vorkommen empty_list; Semantik display
- Beispielwerte: 1,234.56 €

`document.items[].vatRate`

- Vertrag: Typ string; Vorkommen empty_list; Semantik display
- Beispielwerte: 19%

`document.items[].vatCategoryCode`

- Vertrag: Typ string; Vorkommen empty_list; Semantik stable_enum
- Beispielwerte: S

`document.items[].lineTotal`

- Vertrag: Typ string; Vorkommen empty_list; Semantik display
- Beispielwerte: 1,234.56 €

`document.alternativeGroups[].id`

- Vertrag: Typ string; Vorkommen empty_list; Semantik display
- Beispielwerte: net_total

`document.alternativeGroups[].number`

- Vertrag: Typ integer; Vorkommen empty_list; Semantik display
- Beispielwerte: 20

`document.alternativeGroups[].label`

- Vertrag: Typ string; Vorkommen nullable; Semantik display
- Beispielwerte: Total

`document.alternativeGroups[].items[].position`

- Vertrag: Typ integer; Vorkommen empty_list; Semantik display
- Beispielwerte: 1

`document.alternativeGroups[].items[].displayPosition`

- Vertrag: Typ integer; Vorkommen nullable; Semantik display
- Beispielwerte: 20

`document.alternativeGroups[].items[].descriptionXhtml`

- Vertrag: Typ string; Vorkommen empty_list; Semantik safe_xhtml
- Beispielwerte: <p>Prepared text</p>

`document.alternativeGroups[].items[].kind`

- Vertrag: Typ string; Vorkommen empty_list; Semantik stable_enum
- Stabile Werte: priced, text

`document.alternativeGroups[].items[].productSku`

- Vertrag: Typ string; Vorkommen nullable; Semantik display
- Beispielwerte: SKU-1001

`document.alternativeGroups[].items[].productName`

- Vertrag: Typ string; Vorkommen nullable; Semantik display
- Beispielwerte: Consulting package

`document.alternativeGroups[].items[].productImage`

- Vertrag: Typ string; Vorkommen nullable; Semantik resource_handle
- Beispielwerte: data:image/png;base64,...

`document.alternativeGroups[].items[].productImageMaxWidthMm`

- Vertrag: Typ integer; Vorkommen empty_list; Semantik display
- Beispielwerte: 20

`document.alternativeGroups[].items[].productImageMaxHeightMm`

- Vertrag: Typ integer; Vorkommen empty_list; Semantik display
- Beispielwerte: 20

`document.alternativeGroups[].items[].quantityWithUnit`

- Vertrag: Typ string; Vorkommen empty_list; Semantik display
- Beispielwerte: 2.00 h

`document.alternativeGroups[].items[].unitPrice`

- Vertrag: Typ string; Vorkommen empty_list; Semantik display
- Beispielwerte: 1,234.56 €

`document.alternativeGroups[].items[].vatRate`

- Vertrag: Typ string; Vorkommen empty_list; Semantik display
- Beispielwerte: 19%

`document.alternativeGroups[].items[].vatCategoryCode`

- Vertrag: Typ string; Vorkommen empty_list; Semantik stable_enum
- Beispielwerte: S

`document.alternativeGroups[].items[].lineTotal`

- Vertrag: Typ string; Vorkommen empty_list; Semantik display
- Beispielwerte: 1,234.56 €

`document.alternativeGroups[].subtotal`

- Vertrag: Typ string; Vorkommen empty_list; Semantik display
- Beispielwerte: 1,234.56 €

`document.textBlocksBefore[].contentXhtml`

- Vertrag: Typ string; Vorkommen empty_list; Semantik safe_xhtml
- Beispielwerte: <p>Prepared text</p>

`document.textBlocksAfter[].contentXhtml`

- Vertrag: Typ string; Vorkommen empty_list; Semantik safe_xhtml
- Beispielwerte: <p>Prepared text</p>

`document.fullCancellationDescriptionXhtml`

- Vertrag: Typ string; Vorkommen nullable; Semantik safe_xhtml
- Beispielwerte: <p>Prepared text</p>

timesheet

Vorbereitete Zeitraumidentität und vollständige Zeitnachweiseinträge. Diese artspezifische Wurzel ist für die jeweils andere Vorlagenart null.

timesheet.title

- Vertrag: Typ string; Vorkommen required; Semantik display

- Beispielwerte: Invoice

`timesheet.documentTitle`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: Invoice

`timesheet.number`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: INV-2026-001

`timesheet.reference`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: Project Apollo

`timesheet.periodStart`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: 2026-08-20

`timesheet.periodEnd`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: 2026-08-20

`timesheet.descriptionXhtml`

- Vertrag: Typ string; Vorkommen nullable; Semantik safe_xhtml
- Beispielwerte: <p>Prepared text</p>

`timesheet.metadata[].id`

- Vertrag: Typ string; Vorkommen empty_list; Semantik display
- Beispielwerte: net_total

`timesheet.metadata[].label`

- Vertrag: Typ string; Vorkommen empty_list; Semantik display
- Beispielwerte: Total

`timesheet.metadata[].value`

- Vertrag: Typ string; Vorkommen empty_list; Semantik display
- Beispielwerte: Prepared display value

`timesheet.entries[].date`

- Vertrag: Typ string; Vorkommen empty_list; Semantik display
- Beispielwerte: 2026-08-20

`timesheet.entries[].timeRange`

- Vertrag: Typ string; Vorkommen nullable; Semantik display
- Beispielwerte: 09:00-12:30

`timesheet.entries[].descriptionXhtml`

- Vertrag: Typ string; Vorkommen empty_list; Semantik safe_xhtml
- Beispielwerte: <p>Prepared text</p>

`timesheet.entries[].category`

- Vertrag: Typ string; Vorkommen nullable; Semantik display
- Beispielwerte: Consulting

`timesheet.entries[].location`

- Vertrag: Typ string; Vorkommen nullable; Semantik display
- Beispielwerte: Berlin

`timesheet.entries[].duration`

- Vertrag: Typ string; Vorkommen empty_list; Semantik display
- Beispielwerte: 3:30 h

`timesheet.entries[].billable`

- Vertrag: Typ boolean; Vorkommen `empty_list`; Semantik `display`
- Beispielwerte: `true`

`timesheet.hasLocations`

- Vertrag: Typ boolean; Vorkommen `required`; Semantik `display`
- Beispielwerte: `true`

`settings`

Die materialisierten Layouteinstellungen, die das Paket als berücksichtigt deklariert.

`settings.honored[]`

- Vertrag: Typ `string`; Vorkommen `empty_list`; Semantik `stable_enum`
- Stabile Werte: `page.margins`, `fold_marks`, `header`, `footer`, `banner`, `metadata`, `text_blocks`, `tenant_address`, `table`

`settings.page.size`

- Vertrag: Typ `string`; Vorkommen `required`; Semantik `display`
- Beispielwerte: `A4`

`settings.page.marginTopMm`

- Vertrag: Typ `integer`; Vorkommen `required`; Semantik `display`
- Beispielwerte: `20`

`settings.page.marginRightMm`

- Vertrag: Typ `integer`; Vorkommen `required`; Semantik `display`
- Beispielwerte: `20`

`settings.page.marginBottomMm`

- Vertrag: Typ `integer`; Vorkommen `required`; Semantik `display`
- Beispielwerte: `20`

`settings.page.marginLeftMm`

- Vertrag: Typ `integer`; Vorkommen `required`; Semantik `display`
- Beispielwerte: `20`

`settings.page.foldMarks`

- Vertrag: Typ boolean; Vorkommen `required`; Semantik `display`
- Beispielwerte: `true`

`settings.appearance.bannerBackground`

- Vertrag: Typ `string`; Vorkommen `required`; Semantik `display`
- Beispielwerte: `#2d3748`

`settings.appearance.bannerBorder`

- Vertrag: Typ `string`; Vorkommen `required`; Semantik `display`
- Beispielwerte: `#2d3748`

`settings.appearance.bannerText`

- Vertrag: Typ `string`; Vorkommen `required`; Semantik `display`
- Beispielwerte: `#2d3748`

`settings.appearance.metadataBackground`

- Vertrag: Typ `string`; Vorkommen `required`; Semantik `display`
- Beispielwerte: `#2d3748`

`settings.appearance.metadataText`

- Vertrag: Typ `string`; Vorkommen `required`; Semantik `display`
- Beispielwerte: `#2d3748`

`settings.appearance.metadataBorder`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: #2d3748

`settings.appearance.metadataPosition`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: Right

`settings.appearance.textBlockBackground`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: #2d3748

`settings.appearance.textBlockBorder`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: #2d3748

`settings.appearance.textBlockText`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: #2d3748

`settings.appearance.tenantAddressStyle`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: Full

`settings.appearance.tableHeaderStyle`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: Inverted

`settings.appearance.tableBorder`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: #2d3748

`settings.appearance.tableEvenBackground`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: #2d3748

`settings.appearance.tableOddBackground`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: #2d3748

`settings.appearance.tableText`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: #2d3748

`settings.appearance.bannerHeightMm`

- Vertrag: Typ integer; Vorkommen required; Semantik display
- Beispielwerte: 20

`settings.appearance.imageHeightMm`

- Vertrag: Typ integer; Vorkommen required; Semantik display
- Beispielwerte: 20

`format`

Aktive Sprach-, Währungs- und Formatierungsprofil-Metadaten.

`format.locale`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: de-DE

`format.currency`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: EUR

`format.dateStyle`

- **Vertrag:** Typ string; Vorkommen required; Semantik stable_enum
- **Stabile Werte:** medium

`components`

Vorbereitete Eingaben und Verfügbarkeit anwendungseigener Komponenten.

`components.financialSummary.available`

- **Vertrag:** Typ boolean; Vorkommen required; Semantik display
- **Beispielwerte:** true

`components.financialSummary.rows[] .id`

- **Vertrag:** Typ string; Vorkommen empty_list; Semantik display
- **Beispielwerte:** net_total

`components.financialSummary.rows[] .role`

- **Vertrag:** Typ string; Vorkommen empty_list; Semantik stable_enum
- **Stabile Werte:** line_net, rebate, net_after_rebate, vat_bucket_net, vat_bucket_rebate, vat_bucket_tax, vat_total, gross_total

`components.financialSummary.rows[] .label`

- **Vertrag:** Typ string; Vorkommen empty_list; Semantik display
- **Beispielwerte:** Total

`components.financialSummary.rows[] .value`

- **Vertrag:** Typ string; Vorkommen empty_list; Semantik display
- **Beispielwerte:** Prepared display value

`components.financialSummary.rows[] .emphasis`

- **Vertrag:** Typ string; Vorkommen empty_list; Semantik stable_enum
- **Stabile Werte:** normal, subtotal, total, warning

`components.financialSummary.rows[] .keepWithNext`

- **Vertrag:** Typ boolean; Vorkommen empty_list; Semantik display
- **Beispielwerte:** true

`components.deductions.available`

- **Vertrag:** Typ boolean; Vorkommen required; Semantik display
- **Beispielwerte:** true

`components.deductions.heading`

- **Vertrag:** Typ string; Vorkommen required; Semantik display
- **Beispielwerte:** Total

`components.deductions.rows[] .id`

- **Vertrag:** Typ string; Vorkommen empty_list; Semantik display
- **Beispielwerte:** net_total

`components.deductions.rows[] .label`

- **Vertrag:** Typ string; Vorkommen empty_list; Semantik display
- **Beispielwerte:** Total

`components.deductions.rows[] .value`

- **Vertrag:** Typ string; Vorkommen empty_list; Semantik display
- **Beispielwerte:** Prepared display value

`components.deductions.rows[] .emphasis`

- **Vertrag:** Typ string; Vorkommen empty_list; Semantik stable_enum
- **Stabile Werte:** normal, subtotal, total, warning

`components.deductions.rows[] .keepWithNext`

- Vertrag: Typ boolean; Vorkommen empty_list; Semantik display
- Beispielwerte: true

`components.reminderSettlement.available`

- Vertrag: Typ boolean; Vorkommen required; Semantik display
- Beispielwerte: true

`components.reminderSettlement.heading`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: Total

`components.reminderSettlement.rows[].id`

- Vertrag: Typ string; Vorkommen empty_list; Semantik display
- Beispielwerte: net_total

`components.reminderSettlement.rows[].label`

- Vertrag: Typ string; Vorkommen empty_list; Semantik display
- Beispielwerte: Total

`components.reminderSettlement.rows[].value`

- Vertrag: Typ string; Vorkommen empty_list; Semantik display
- Beispielwerte: Prepared display value

`components.reminderSettlement.rows[].emphasis`

- Vertrag: Typ string; Vorkommen empty_list; Semantik stable_enum
- Stabile Werte: normal, subtotal, total, warning

`components.reminderSettlement.rows[].keepWithNext`

- Vertrag: Typ boolean; Vorkommen empty_list; Semantik display
- Beispielwerte: true

`components.payment.available`

- Vertrag: Typ boolean; Vorkommen required; Semantik display
- Beispielwerte: true

`components.payment.payableLabel`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: Prepared display text

`components.payment.payable`

- Vertrag: Typ string; Vorkommen required; Semantik display
- Beispielwerte: 1,234.56 €

`components.payment.instructions[].type`

- Vertrag: Typ string; Vorkommen empty_list; Semantik stable_enum
- Stabile Werte: bank_transfer, paypal

`components.payment.instructions[].heading`

- Vertrag: Typ string; Vorkommen empty_list; Semantik display
- Beispielwerte: Total

`components.payment.instructions[].instructionXhtml`

- Vertrag: Typ string; Vorkommen empty_list; Semantik safe_xhtml
- Beispielwerte: <p>Prepared text</p>

`components.payment.instructions[].accountHolderLabel`

- Vertrag: Typ string; Vorkommen empty_list; Semantik display
- Beispielwerte: Prepared display text

`components.payment.instructions[].bankNameLabel`

- Vertrag: Typ string; Vorkommen empty_list; Semantik display
- Beispielwerte: Prepared display text

`components.payment.instructions[].emailLabel`

- Vertrag: Typ string; Vorkommen empty_list; Semantik display
- Beispielwerte: Prepared display text

`components.payment.instructions[].linkLabel`

- Vertrag: Typ string; Vorkommen empty_list; Semantik display
- Beispielwerte: Prepared display text

`components.payment.instructions[].accountHolder`

- Vertrag: Typ string; Vorkommen nullable; Semantik display
- Beispielwerte: Example GmbH

`components.payment.instructions[].iban`

- Vertrag: Typ string; Vorkommen nullable; Semantik display
- Beispielwerte: DE02120300000000202051

`components.payment.instructions[].bic`

- Vertrag: Typ string; Vorkommen nullable; Semantik display
- Beispielwerte: BYLADEM1001

`components.payment.instructions[].bankName`

- Vertrag: Typ string; Vorkommen nullable; Semantik display
- Beispielwerte: Example Bank

`components.payment.instructions[].email`

- Vertrag: Typ string; Vorkommen nullable; Semantik display
- Beispielwerte: billing@example.com

`components.payment.instructions[].link`

- Vertrag: Typ string; Vorkommen nullable; Semantik display
- Beispielwerte: https://example.com/pay

`components.payment.instructions[].qrResource`

- Vertrag: Typ string; Vorkommen nullable; Semantik resource_handle
- Beispielwerte: data:image/png;base64,...

`components.payment.skonto.heading`

- Vertrag: Typ string; Vorkommen nullable; Semantik display
- Beispielwerte: Total

`components.payment.skonto.discountedTotalLabel`

- Vertrag: Typ string; Vorkommen nullable; Semantik display
- Beispielwerte: Prepared display text

`components.payment.skonto.percentage`

- Vertrag: Typ string; Vorkommen nullable; Semantik display
- Beispielwerte: 19%

`components.payment.skonto.days`

- Vertrag: Typ integer; Vorkommen nullable; Semantik display
- Beispielwerte: 14

`components.payment.skonto.discountAmount`

- Vertrag: Typ string; Vorkommen nullable; Semantik display
- Beispielwerte: 1,234.56 €

`components.payment.skonto.discountedTotal`

- Vertrag: Typ string; Vorkommen nullable; Semantik display
- Beispielwerte: 1,234.56 €

`components.legalDisclosures.available`

- **Vertrag:** Typ boolean; Vorkommen required; Semantik display
- **Beispielwerte:** true

`components.legalDisclosures.notesXhtml[]`

- **Vertrag:** Typ string; Vorkommen empty_list; Semantik safe_xhtml
- **Beispielwerte:** <p>Prepared text</p>

`components.timesheetSummary.available`

- **Vertrag:** Typ boolean; Vorkommen required; Semantik display
- **Beispielwerte:** true

`components.timesheetSummary.billableLabel`

- **Vertrag:** Typ string; Vorkommen required; Semantik display
- **Beispielwerte:** Prepared display text

`components.timesheetSummary.billable`

- **Vertrag:** Typ string; Vorkommen required; Semantik display
- **Beispielwerte:** Prepared display text

`components.timesheetSummary.nonBillableLabel`

- **Vertrag:** Typ string; Vorkommen required; Semantik display
- **Beispielwerte:** Prepared display text

`components.timesheetSummary.nonBillable`

- **Vertrag:** Typ string; Vorkommen required; Semantik display
- **Beispielwerte:** Prepared display text

`components.timesheetSummary.totalLabel`

- **Vertrag:** Typ string; Vorkommen required; Semantik display
- **Beispielwerte:** Prepared display text

`components.timesheetSummary.total`

- **Vertrag:** Typ string; Vorkommen required; Semantik display
- **Beispielwerte:** Prepared display text

Anwendungseigene Komponenten

Includes in `invoicer/components/v1/` werden ausschließlich von der Anwendung aufgelöst. Komponenten lesen ihre vorbereitete Ansicht und akzeptieren in v1 keine beliebigen Pebble-Argumente. Internes Markup ist über die dokumentierten stabilen Hooks hinaus nicht festgeschrieben.

`financial-summary`

- **Include:** `invoicer/components/v1/financial-summary.peb`
- **Vorbereitete Ansicht:** `components.financialSummary`
- **Stabiler äußerer Hook:** `[data-invoicer-component="financial-summary"]`
- **Stabile Zeilen-Hooks:** `[data-row-role="<role>"], .inv-c-role-<role>`
- **Argumente:** none
- **Seitenumbruch:** Hält eine unteilbare Zeile zusammen und darf die vollständige Übersicht verschieben
- **Vertragsrichtlinie:** Liest die vorbereitete Ansicht `components.financialSummary`; beliebige Argumente werden in v1 nicht unterstützt.

`deductions`

- **Include:** `invoicer/components/v1/deductions.peb`
- **Vorbereitete Ansicht:** `components.deductions`
- **Stabiler äußerer Hook:** `[data-invoicer-component="deductions"]`
- **Argumente:** none
- **Seitenumbruch:** Trennt nur zwischen vorbereiteten Zeilen
- **Vertragsrichtlinie:** Liest die vorbereitete Ansicht `components.deductions`; beliebige Argumente werden in v1 nicht unterstützt.

reminder-settlement

- **Include:** `invoicer/components/v1/reminder-settlement.peb`
- **Vorbereitete Ansicht:** `components.reminderSettlement`
- **Stabiler äußerer Hook:** `[data-invoicer-component="reminder-settlement"]`
- **Argumente:** none
- **Seitenumbruch:** Hält mit `keepWithNext` markierte Zeilen zusammen
- **Vertragsrichtlinie:** Liest die vorbereitete Ansicht `components.reminderSettlement`; beliebige Argumente werden in v1 nicht unterstützt.

payment

- **Include:** `invoicer/components/v1/payment.peb`
- **Vorbereitete Ansicht:** `components.payment`
- **Stabiler äußerer Hook:** `[data-invoicer-component="payment"]`
- **Argumente:** none
- **Seitenumbruch:** Hält jeden Anweisungsblock zusammen und darf zwischen Anweisungen trennen
- **Vertragsrichtlinie:** Liest die vorbereitete Ansicht `components.payment`; beliebige Argumente werden in v1 nicht unterstützt.

legal-disclosures

- **Include:** `invoicer/components/v1/legal-disclosures.peb`
- **Vorbereitete Ansicht:** `components.legalDisclosures`
- **Stabiler äußerer Hook:** `[data-invoicer-component="legal-disclosures"]`
- **Argumente:** none
- **Seitenumbruch:** Trennt nur zwischen vorbereiteten rechtlichen Hinweisen
- **Vertragsrichtlinie:** Liest die vorbereitete Ansicht `components.legalDisclosures`; beliebige Argumente werden in v1 nicht unterstützt.

timesheet-summary

- **Include:** `invoicer/components/v1/timesheet-summary.peb`
- **Vorbereitete Ansicht:** `components.timesheetSummary`
- **Stabiler äußerer Hook:** `[data-invoicer-component="timesheet-summary"]`
- **Argumente:** none
- **Seitenumbruch:** Hält die kurze Übersicht zusammen
- **Vertragsrichtlinie:** Liest die vorbereitete Ansicht `components.timesheetSummary`; beliebige Argumente werden in v1 nicht unterstützt.

Ein Paket darf eine Komponente neu anordnen, umschließen, neu gestalten oder auslassen. Das Auslassen legt die zugrunde liegenden Geschäftsregeln nicht offen.

Pebble und Ressourcen

Erlaubte Pebble-Tags: `block`, `else`, `elseif`, `embed`, `endblock`, `endembed`, `endfor`, `endif`, `extends`, `for`, `if`, `include`, `set`.

Erlaubte Filter: `raw`. Funktionsaufrufe und dynamische Ressourcenpfade sind verboten. Strikte Variablen sind aktiviert und die gerenderte Ausgabe ist auf 8000000 Zeichen begrenzt.

- Paket-Includes werden nur innerhalb des Pakets aufgelöst; absolute Pfade und übergeordnete Pfadsegmente sind verboten.
- Der Namensraum `invoicer/components/v1` gehört der Anwendung und kann nicht überschattet werden.
- Dynamische Pfade für `include`, `extends` und `embed` sind verboten.
- Das Laden lokaler Dateien und Netzwerkressourcen gehört nicht zu Template API v1.
- Paketlokale PNG- und JPEG-Bilder müssen in `manifest.json` deklariert und über normalisierte paketrelative Pfade referenziert werden; sie werden aus den kanonischen Bytes materialisiert, ohne Dateisystempfade offenzulegen.
- Jedes Bild ist auf 4096x4096 Pixel und alle Bilder zusammen auf 40000000 Pixel begrenzt.
- Paketlokale Schriften müssen als normalisierte statische TrueType-.ttf-Varianten deklariert werden, deren Familie, Stärke und normaler oder kursiver Stil mit dem Schriftprogramm übereinstimmen.
- Ein Paket darf höchstens 8 Schriftvarianten mit jeweils höchstens 1048576 Bytes und zusammen höchstens 4194304 Bytes enthalten.

- Deklarierte Schriften müssen die Dokumenteinbettung erlauben. OTF, TTC, WOFF, WOFF2, variable, farbige, Bitmap- und Type-1-Schriften sowie Betriebssystem-, lokale Datei- und Netzwerkschriften werden nicht unterstützt.
- Die importierende Organisation muss bestätigen, dass sie die bereitgestellte Schriftsoftware in erzeugte Dokumente und Archive einbetten und weitergeben darf.
- Vom Paket erzeugte data-, file- und Netzwerkressourcen-URLs sind verboten; von der Anwendung vorbereitete Datenressourcen bleiben ausschließlich über dokumentierte Felder der Template API verfügbar.

CSS und XHTML

- Die Vorlagenausgabe muss wohlgeformtes XHTML sein; JavaScript und Browser-DOM-APIs sind nicht verfügbar.
- CSS wird von der Flying-Saucer-Druckpipeline der Anwendung und nicht von einer Browser-Engine gerendert.
- Template API v1 lehnt die von Flying Saucer nicht unterstützten HTML5-Elemente ab: `<article>`, `<aside>`, `<audio>`, `<canvas>`, `<datalist>`, `<details>`, `<dialog>`, `<figcaption>`, `<figure>`, `<footer>`, `<header>`, `<main>`, `<mark>`, `<meter>`, `<nav>`, `<output>`, `<picture>`, `<progress>`, `<section>`, `<summary>`, `<template>`, `<time>`, `<track>`, `<video>`, `<wbr>`.
- Template API v1 lehnt CSS-Deklarationen ab, die Flying Saucer als nicht unterstützt meldet; verwende XHTML- und CSS-2.1-kompatible Block-, Inline-, Float- oder Tabellenlayouts.
- Druck-CSS, @page-Regeln, Tabellen mit festem Layout, gewöhnlicher Blockfluss und begrenzte Floats werden unterstützt, wenn der Renderer sie akzeptiert.
- Binde `common.resources.fontCss` vor den Paketstilen ein. Es enthält anwendungseigene Deklarationen für die gepflegte IBM Plex Sans und jede validierte paketlokale Schriftvariante.
- Wähle eine deklarierte Paketvariante im Paket-CSS anhand ihrer Manifestfamilie, -stärke und ihres Stils; verwende IBM Plex Sans oder eine generische Familie als ausdrücklichen Fallback.
- Externe Stylesheets, Netzwerkschriften, lokale Datei-URLs und pixelgenaue Browserparität werden nicht unterstützt.

Diagnosen

ZIP-Import und Kanonisierung des Pakets schlagen mit diesen Codes deterministisch fehl:

- `ARCHIVE_MISSING`
- `ARCHIVE_NOT_REGULAR_FILE`
- `ARCHIVE_TOO_LARGE`
- `ARCHIVE_INVALID`
- `ARCHIVE_ENCRYPTED`
- `ENTRY_COUNT_EXCEEDED`
- `ENTRY_DUPLICATE`
- `ENTRY_DIRECTORY_FORBIDDEN`
- `ENTRY_SYMLINK_FORBIDDEN`
- `ENTRY_PATH_INVALID`
- `ENTRY_TYPE_UNSUPPORTED`
- `ENTRY_TOO_LARGE`
- `EXPANDED_SIZE_EXCEEDED`
- `COMPRESSION_RATIO_EXCEEDED`
- `MANIFEST_MISSING`
- `MANIFEST_WRAPPER_DIRECTORY`
- `MANIFEST_INVALID`
- `IMAGE_INVALID`
- `IMAGE_DIMENSIONS_EXCEEDED`
- `IMAGE_PIXEL_BUDGET_EXCEEDED`
- `FONT_INVALID`
- `FONT_UNSUPPORTED`
- `FONT_METADATA_MISMATCH`
- `FONT_EMBEDDING_RESTRICTED`
- `FONT_COUNT_EXCEEDED`
- `FONT_SIZE_EXCEEDED`
- `FONT_TOTAL_SIZE_EXCEEDED`
- `PACKAGE_FORMAT_UNSUPPORTED`
- `TEMPLATE_API_VERSION_UNSUPPORTED`
- `PACKAGE_ID_INVALID`
- `DISPLAY_NAME_INVALID`

- PATH_INVALID
- PATH_COLLISION
- MANIFEST_PATH_RESERVED
- ENTRY_POINT_MISSING
- RESOURCE_MISSING
- FONT_PATH_INVALID
- FONT_PATH_DUPLICATE
- FONT_FAMILY_INVALID
- FONT_WEIGHT_INVALID
- FONT_FACE_AMBIGUOUS
- FONT_RESOURCE_UNDECLARED
- DOCUMENT_FAMILY_INVALID
- HONORED_SETTING_INVALID

Validierung und Rendering des Vorlagenvertrags schlagen mit diesen Codes deterministisch fehl:

- API_VERSION_UNSUPPORTED
- KIND_MISMATCH
- DOCUMENT_FAMILY_UNSUPPORTED
- ENTRY_POINT_MISSING
- RESOURCE_MISSING
- RESOURCE_FORBIDDEN
- COMPONENT_NAMESPACE_SHADOWED
- PEBBLE_FEATURE_FORBIDDEN
- RENDERER_FEATURE_UNSUPPORTED
- PRESENTATION_INCOMPATIBLE
- OUTPUT_LIMIT_EXCEEDED
- EVALUATION_FAILED

Behandle Fehler bei Version, Art, Familie, Einstiegspunkt und Präsentation als Manifest- oder API-Probleme, Fehler bei Ressourcen, Namensräumen, Renderer-Kompatibilität und Pebble-Funktionen als Probleme der Paketquelle und Fehler bei Ausgabe oder Auswertung als Renderingprobleme.

Verantwortungsgrenze

Die Anwendung verantwortet vorbereitete Quelldaten, finanzielle Entscheidungen, Lebenszyklusentscheidungen, Anwendungskomponenten, strikte Ausführung, PDF/A-Verarbeitung und die Erzeugung eingebetteter E-Rechnungen. Das importierende Unternehmen verantwortet die visuelle, textliche und rechtliche Eignung eines kundenerstellten Layouts einschließlich bewusst ausgelassener Komponenten. Für bereitgestellte Schriftsoftware muss die importierende Organisation die erforderlichen Rechte besitzen und bestätigen, um die exakten Dateien in erzeugte Dokumente und Archive einzubetten und weiterzugeben.