

# **Entwicklung komplexer betrieblicher Anwendungssysteme**

Dokumentation

(\$SS25\_RAP4CINEMA)



**Verfasser der Arbeit**

Kamal Badawi - 5236028

**Betreuer**

Prof. Dr. Sören Aguirre Reid

# Inhaltsverzeichnis

Abbildungsverzeichnis .....	4
Tabellenverzeichnis .....	5
1. Einleitung.....	6
1.1. Einführung in das Thema .....	6
1.2. Idee .....	6
1.3. Zielsetzung .....	7
1.4. Vorgehensweise .....	7
2. Architektur .....	9
2.1. ER-Modell .....	9
2.2. Klassendiagramm.....	14
2.3. Datenflussdiagramm .....	17
2.4. Use-Case-Diagramm.....	19
2.5. Zustandsdiagramm.....	22
3. Technische Umsetzung in SAP RAP .....	28
3.1. Packages in SAP RAP .....	28
3.2. Database Tables.....	32
3.3. Dateneingabe in SAP Logon mit SE16N .....	41
3.4. Basis View.....	45
3.5. Interface View (Reusable View) .....	61
3.6. Projection View (Consumption View).....	71
3.7. Service Definition .....	82
3.8. Service Binding .....	87
3.9. Metadata Extension (UI-Annotationen) .....	91
3.10. Behavior Definition.....	112
3.11. Draft Database Tables .....	125
4. Erweiterungen und neue Funktionalitäten in SAP RAP .....	125
4.1. Pflichtfelder (Obligatorische Felder) .....	125
4.2. Verwendung der QuickInfo-Annotation.....	126
4.3. Unterstützung von Datumsbereichen .....	127
4.4. Farbige Kennzeichnung von Datensätzen.....	129
4.5. Integration von Bildern und Symbolen.....	132
4.6. Erstellung von Drop Down.....	140

4.7.	Definition von Kennzahlen im Kopfbereich der Object Page .....	148
4.8.	Umsetzung der UI-Gruppierung auf der Object Page.....	151
4.9.	Umsetzung mehrerer UI-Gruppierungen auf der Object Page .....	154
5.	Auswertungen und Kennzahlen.....	161
5.1.	Verkaufsleistung der Mitarbeiter.....	161
5.2.	Treuekunden- und Top-Kundenanalyse.....	165
6.	Fiori-Anwendungen in ABAP RAP .....	170
6.1.	Von RAP zu Fiori: Implementierung und Anpassung einer Kinoverwaltungs-App	170
6.2.	Fiori CSS-Einbindung in Fiori-Anwendungen .....	190
7.	CRUD-Tests der Kinoverwaltungs-App.....	192
7.1.	Anlegen von Einträgen (Create).....	192
7.2.	Bearbeiten von Einträgen (Update).....	196
7.3.	Löschen von Einträgen (Delete).....	198
8.	Herausforderungen und Lösungsansätze.....	204
8.1.	Durchführung von CRUD-Operationen in hierarchischen Datenmodellen	204
8.2.	Prinzip der eindeutigen Root-Zuordnung in hierarchischen Datenmodellen	210
8.3.	Einschränkungen bei der Unterstützung von Datentypen im Service Binding	211
9.	GitHub-Integration mit abapGit.....	214
9.1.	SSL-Zertifikat im SAP-System konfigurieren.....	214
9.2.	Erstellung eines Personal Access Tokens in GitHub .....	223
9.3.	Vorgehensweise zur Installation von abapGit im SAP-System.....	224
9.4.	Code mit abapGit auf GitHub pushen.....	233
10.	Fazit .....	235
	GitHub .....	237
	Literaturverzeichnis .....	237

# Abbildungsverzeichnis

Abbildung 1: RAP4Cinema - ER-Modell .....	14
Abbildung 2: RAP4Cinema - Klassendiagramm .....	16
Abbildung 3: RAP4Cinema - Datenflussdiagramm (Kino-Management) .....	18
Abbildung 4: RAP4Cinema - Datenflussdiagramm (Besucher & Mitarbeiter) .....	19
Abbildung 5: Use-Case-Diagramm (Mitarbeiter, Besucher und Zahlungsanbieter) ..	21
Abbildung 6: Use-Case-Diagramm (Systemadministrator) .....	22
Abbildung 7: Zustandsdiagramm - Ticket-Lebenszyklus.....	23
Abbildung 8: Zustandsdiagramm - Besucher.....	24
Abbildung 9: Zustandsdiagramm - Verkauf.....	25
Abbildung 10: Zustandsdiagramm - Aufführung eines Films.....	26
Abbildung 11: Zustandsdiagramm - Verwaltung .....	27
Abbildung 12: Zustandsdiagramm - Mitarbeiter (Arbeitsprozess).....	28
Abbildung 13: ER-Modell Kino - Raum - Vorstellung - Ticket.....	210

# Tabellenverzeichnis

Tabelle 1: RAP4Cinema - Beziehungen zwischen den Tabellen..... 13

# 1. Einleitung

## 1.1. Einführung in das Thema

In modernen Kinobetriebssystemen stellt die Verwaltung von Kinos, Sälen, Filmen, Vorstellungen, Besuchern, Tickets und Personal eine komplexe Herausforderung dar. Klassische monolithische Anwendungen stoßen dabei häufig an ihre Grenzen - insbesondere, wenn es um Datenkonsistenz, Modularität, Erweiterbarkeit und Sicherheit geht. Um diesen Anforderungen gerecht zu werden, setzt die heutige Systementwicklung zunehmend auf serviceorientierte und cloudfähige Architekturen. Das SAP RESTful Application Programming Model (RAP) bietet hierfür einen modernen, standardisierten Ansatz zur modellbasierten Entwicklung von Anwendungen auf der SAP Business Technology Platform (BTP).

## 1.2. Idee

Ziel des Projekts RAP4Cinema ist die Entwicklung eines vollständig modularisierten und cloud-fähigen Kino-Management-Systems auf Basis des SAP RESTful Application Programming Model (RAP). Die Grundidee von RAP4Cinema besteht darin, sämtliche relevanten Entitäten eines Kinosystems - wie Kinos, Räume, Filme, Vorstellungen, Besucher, Tickets, Mitarbeiter und Verkäufe - in eigenständigen Core Data Services (CDS) Views zu modellieren. Jede dieser Entitäten bildet die Grundlage für weiterführende Reusable Views, die komplexe Beziehungen zwischen den Objekten abbilden, beispielsweise die Verknüpfung von Filmen mit Vorstellungen, Räumen und Kinos oder die Zuordnung von Tickets zu Besuchern und Vorstellungen. Diese modulare Architektur ermöglicht eine effiziente Wiederverwendung, eine konsistente Datenhaltung und eine klare Trennung der Verantwortlichkeiten innerhalb des Systems. Auf Basis der Reusable Views werden Consumption Views entwickelt, die speziell für den Einsatz in SAP Fiori/UI5-Anwendungen sowie für Reporting-Zwecke optimiert sind. Sie bündeln Daten aus verschiedenen Entitäten und stellen sie in einer konsolidierten, benutzerfreundlichen Form dar. Parallel dazu werden in RAP4Cinema Behavior Definitions implementiert, die die zulässigen CRUD-Operationen (Create, Read, Update, Delete) sowie geschäftsspezifische Aktionen - etwa den Ticketkauf - definieren. Diese Schicht stellt sicher, dass alle Datenmanipulationen zentral gesteuert, konsistent, sicher und nachvollziehbar ausgeführt werden. Die Daten werden über Service Definitions und OData V4 Service Bindings bereitgestellt, sodass sie modular, standardisiert und technologieübergreifend konsumierbar sind. Ergänzend kommen Metadata Extensions zum Einsatz, die UI-relevante Annotationen bereitstellen. Dadurch lässt sich die Darstellung in Fiori-Anwendungen gezielt optimieren, ohne die zugrunde liegenden Datenmodelle zu verändern. Das Projekt RAP4Cinema folgt konsequent dem Prinzip der Clean Architecture, das eine klare Trennung zwischen Datenmodell, Geschäftslogik, Service-Exposition und UI-Annotationen sicherstellt. Dies führt zu einer hohen Wartbarkeit, Erweiterbarkeit und Wiederverwendbarkeit der einzelnen Komponenten. Durch den Einsatz von SAP RAP profitiert RAP4Cinema

zudem von transaktionaler Konsistenz, standardisierten Serviceschnittstellen und einer nahtlosen Integration in die SAP Business Technology Platform (BTP). Insgesamt entsteht mit RAP4Cinema ein robustes, skalierbares und übersichtliches Kino-Management-System, das sämtliche wesentlichen Geschäftsprozesse eines Kinos abbildet - von der Verwaltung von Räumen und Filmen über den Ticketverkauf bis hin zur Personal- und Umsatzverwaltung - und dabei moderne, SAP-native Entwicklungsparadigmen nutzt.

### 1.3. Zielsetzung

Das Ziel des Projekts besteht darin, ein referenzfähiges, erweiterbares und praxisnahes Kino-Management-System zu entwickeln, das als Grundlage für die Demonstration moderner ABAP-RAP-Architekturen dienen kann.

Im Detail sollen folgende Ziele erreicht werden:

- Modellierung eines vollständigen, logisch konsistenten Datenmodells auf Basis von CDS-Views.
- Implementierung von Behavior Definitions zur Abbildung zentraler Geschäftsprozesse wie Ticketverkauf, Vorstellungspflege und Besucherregistrierung.
- Bereitstellung standardisierter OData-Services für externe Anwendungen und Fiori-Frontends.
- Integration von UI-Annotationen zur automatischen Generierung benutzerfreundlicher Fiori-Oberflächen.
- Sicherstellung von Modularität, Wiederverwendbarkeit und Erweiterbarkeit im Sinne einer Clean Architecture.
- Demonstration der Vorteile von SAP RAP im Hinblick auf Performance, Wartbarkeit und Integration in die SAP BTP.

Langfristig soll das System nicht nur als Beispielprojekt dienen, sondern auch als Grundlage für zukünftige Erweiterungen wie dynamische Preisgestaltung, Auslastungsanalysen oder KI-gestützte Besucherprognosen.

### 1.4. Vorgehensweise

In Kapitel 2 wird die Architektur der Anwendung RAP4Cinema detailliert beschrieben. Dieses Kapitel legt die Grundlage für das Verständnis des Projekts, da es die Struktur, den Aufbau und das Zusammenspiel der einzelnen Komponenten erläutert. Zunächst wird das Entity-Relationship-Modell (ER-Modell) vorgestellt, das die zugrunde liegende Datenstruktur und die Beziehungen zwischen den Entitäten beschreibt. Darauf folgt das Klassendiagramm, das die Interaktion der einzelnen Klassen und die Umsetzung der objektorientierten Modellierung zeigt. Das Datenflussdiagramm verdeutlicht den Informationsaustausch zwischen den Komponenten und definiert die Systemgrenzen,

während das Use-Case-Diagramm die zentralen Anwendungsfälle aus Sicht der Benutzer darstellt. Ergänzend beschreibt das Zustandsdiagramm die möglichen Zustände einzelner Objekte und deren Übergänge im Lebenszyklus des Systems. Kapitel 3 widmet sich der technischen Umsetzung in SAP RAP und zeigt, wie das theoretische Wissen aus dem Modul „Entwicklung komplexer betrieblicher Anwendungssysteme“ praktisch angewendet wurde. Zunächst werden die grundlegenden Packages in SAP RAP vorgestellt. Anschließend werden die Datenbanktabellen beschrieben und die Dateneingabe über SAP Logon mit SE16N erläutert. Auf dieser Basis werden die verschiedenen View-Ebenen behandelt: Die Basis View definiert die Grunddaten, die Interface View dient als wiederverwendbare Schicht, und die Projection View stellt die konsumierbaren Daten dar. Darauf aufbauend wird erklärt, wie die Anwendung über die Service Definition und das Service Binding als OData-Service bereitgestellt wird. Metadata Extensions werden zur Gestaltung der Benutzeroberfläche eingesetzt, während die Behavior Definition das Anwendungsverhalten festlegt. Abschließend werden Draft Database Tables vorgestellt, die Entwurfs- und Zwischenspeicherfunktionen unterstützen. Kapitel 4 beschreibt die Erweiterungen und neuen Funktionalitäten in SAP RAP. Dazu gehören die Implementierung von Pflichtfeldern, die Verwendung der QuickInfo-Annotation, die Unterstützung von Datumsbereichen, die farbige Kennzeichnung von Datensätzen, die Integration von Bildern und Symbolen sowie die Erstellung von Drop-Downs. Außerdem werden Kennzahlen im Kopfbereich der Object Page definiert und die Umsetzung von UI-Gruppierungen auf der Object Page, auch mehrfach, erläutert. Kapitel 5 behandelt die Auswertungen und Kennzahlen, wie die Verkaufsleistung der Mitarbeiter sowie Treuekunden- und Top-Kundenanalysen. Kapitel 6 beschreibt die Integration von SAP Fiori in die Anwendung, insbesondere die Umsetzung und Anpassung der Kinoverwaltungs-App sowie die Einbindung von Fiori CSS. Kapitel 7 widmet sich den CRUD-Tests der Anwendung, also dem Anlegen, Bearbeiten und Löschen von Einträgen. In Kapitel 8 werden die während der Umsetzung aufgetretenen Herausforderungen und Lösungsansätze beschrieben, insbesondere bei CRUD-Operationen in hierarchischen Datenmodellen, der eindeutigen Root-Zuordnung und den Einschränkungen bei der Unterstützung von Datentypen im Service Binding. Kapitel 9 erläutert die GitHub-Integration mittels abapGit. Hierzu gehören die Konfiguration von SSL-Zertifikaten im SAP-System, die Erstellung von Personal Access Tokens in GitHub, die Installation von abapGit und das Pushen von Code auf GitHub. Abschließend fasst Kapitel 10 die Ergebnisse und Erkenntnisse des Projekts zusammen. Im Fazit werden die Zielerreichung bewertet, die wichtigsten Erfahrungen reflektiert und mögliche Ansätze für zukünftige Erweiterungen oder Optimierungen aufgezeigt.

## 2. Architektur

### 2.1. ER-Modell

Das Entity-Relationship-Modell (ERM) beschreibt die logische Struktur des entwickelten Kino-Management-Systems RAP4Cinema. Es definiert die zentralen Entitäten, deren Attribute sowie die Beziehungen untereinander. Ziel der Modellierung ist es, eine konsistente, normalisierte und erweiterbare Datenbasis zu schaffen, die alle relevanten Geschäftsobjekte eines Kinobetriebs abbildet - von Kinos und Räumen über Filme und Vorstellungen bis hin zu Besuchern, Tickets, Verkäufen und Mitarbeitern.

Das Modell folgt den Prinzipien der Datenbanknormalisierung (bis zur dritten Normalform) und verwendet Primär- und Fremdschlüsselbeziehungen, um Redundanzen zu vermeiden und die referentielle Integrität sicherzustellen.

Die folgende Übersicht zeigt die Struktur der Datenbanktabellen des Projekts RAP4Cinema. Sie beschreibt die wichtigsten Attribute sowie deren Beziehungen zu anderen Tabellen.

**ZSS25\_R4C\_KINO - Kino:** Die Tabelle speichert Stammdaten zu den einzelnen Kinos.

- **Primärschlüssel:** MANDT, KINO\_ID
- **Attribute:**
  - NAME
  - ADRESSE
  - STADT
  - PLZ
  - ANZAHL\_RAEUME
  - EROEFFNUNGSDATUM
  - KONTAKTINFO
- **Beziehung:** Ein Kino kann mehrere Räume besitzen (1:n-Beziehung zu ZSS25\_R4C\_RAUM)

**ZSS25\_R4C\_RAUM - Raum (Kinosaal):** Die Tabelle enthält Informationen zu den Räumen innerhalb eines Kinos.

- **Primärschlüssel:** MANDT, RAUM\_ID
- **Fremdschlüssel:** KINO\_ID → ZSS25\_R4C\_KINO.KINO\_ID
- **Attribute:**
  - NAME
  - STUEHL\_ANZAHL\_PRO\_REIHE
  - SITZPLAETZE
  - AUSSTATTUNG
- **Beziehungen:**

- Jeder Raum gehört genau einem Kino (n:1)
- In einem Raum finden mehrere Vorstellungen statt (1:n-Beziehung zu ZSS25\_R4C\_VORST)

**ZSS25\_R4C\_FILM - Film:** Die Tabelle beschreibt die im Kino gezeigten Filme.

- **Primärschlüssel:** MANDT, FILM\_ID
- **Attribute:**
  - TITEL
  - GENRE
  - DAUER\_MINUTEN
  - ALTERSFREIGABE
  - PRODUKTIONSJAHR
  - REGISSEUR
- **Beziehung:**
  - Jeder Film wird genau in einer Vorstellung gezeigt (1:1-Beziehung zu ZSS25\_R4C\_VORST)

**ZSS25\_R4C\_VORST - Vorstellung:** Diese Tabelle modelliert einzelne Filmvorführungen.

- **Primärschlüssel:** MANDT, VORSTELLUNG\_ID
- **Fremdschlüssel:**
  - FILM\_ID → ZSS25\_R4C\_FILM.FILM\_ID
  - RAUM\_ID → ZSS25\_R4C\_RAUM.RAUM\_ID
- **Attribute:**
  - SPRACHE
  - DATUM
  - STARTZEIT
  - ENDZEIT
  - UNTERTITEL
  - VORSTELLUNG\_FORMAT
  - STATUS
- **Beziehungen:**
  - Eine Vorstellung findet in genau einem Raum statt (n:1)
  - Eine Vorstellung bezieht sich auf genau einen Film (1:1)
  - Zu jeder Vorstellung gehören mehrere Tickets (1:n-Beziehung zu ZSS25\_R4C\_TICKET)

**ZSS25\_R4C\_BESUCH - Besucher:** Speichert die Stammdaten der registrierten Kinobesucher.

- **Primärschlüssel:** MANDT, BESUCHER\_ID
- **Attribute:**
  - NAME
  - GEBURTSDATUM
  - EMAIL
  - TELEFON
  - ADRESSE
  - REGISTRIERUNG
- **Beziehung:**
  - Ein Besucher kauft ein Ticket für eine Vorstellung (1:1-Beziehung zu ZSS25\_R4C\_TICKET)

**ZSS25\_R4C\_TICKET - Ticket:** Die Tabelle bildet die ausgestellten Eintrittskarten ab.

- **Primärschlüssel:** MANDT, TICKET\_ID
- **Fremdschlüssel:**
  - VORSTELLUNG\_ID → ZSS25\_R4C\_VORST.VORSTELLUNG\_ID
  - BESUCHER\_ID → ZSS25\_R4C\_BESUCH.BESUCHER\_ID
- **Attribute:**
  - SITZPLATZNUMMER
  - PREIS
  - TICKET\_TYP
  - QR\_CODE
  - STATUS
- **Beziehungen:**
  - Ein Ticket gehört genau einem Besucher (1:1)
  - Ein Ticket bezieht sich auf genau eine Vorstellung (1:n)
  - Jedes Ticket ist Bestandteil eines Verkaufs (1:1-Beziehung zu ZSS25\_R4C\_VERKF)

**ZSS25\_R4C\_MITARB - Mitarbeiter:** Diese Tabelle enthält die Stammdaten der Mitarbeiter des Kinos.

- **Primärschlüssel:** MANDT, MITARBEITER\_ID
- **Attribute:**
  - NAME
  - GEBURTSDATUM
  - ROLLE, ABTEILUNG
  - EINSTELLUNGSDATUM
  - GEHALT
  - KONTAKTINFO

- **Beziehung:**
  - Ein Mitarbeiter kann mehrere Verkäufe tätigen (1:n-Beziehung zu ZSS25\_R4C\_VERKF)

**ZSS25\_R4C\_VERKF - Verkauf (Faktentabelle):** Diese Faktentabelle dokumentiert die Verkaufsprozesse im Kino.

**Primärschlüssel:** MANDT, VERKAUF\_ID

- **Fremdschlüssel:**
  - TICKET\_ID → ZSS25\_R4C\_TICKET.TICKET\_ID
  - MITARBEITER\_ID → ZSS25\_R4C\_MITARB.MITARBEITER\_ID
- **Attribute:**
  - VERKAUFSDATUM
  - ZAHLUNGSART
  - RABATT
- **Beziehungen:**
  - Jeder Verkauf wird von genau einem Mitarbeiter durchgeführt (n:1)
  - Jeder Verkauf betrifft genau ein Ticket (1:1)

In der folgenden Tabelle werden die Beziehungen zwischen den Datenbanktabellen des Projekts RAP4Cinema dargestellt.

Beziehung	Kardinalität	Beschreibung
Kino - Raum	1:n	Ein Kino besitzt mehrere Räume, ein Raum gehört zu genau einem Kino.
Raum - Vorstellung	1:n	In einem Raum finden mehrere Vorstellungen statt, jede Vorstellung findet in genau einem Raum statt.
Film - Vorstellung	1:1	Jeder Film wird genau in einer Vorstellung gezeigt, jede Vorstellung bezieht sich auf genau einen Film.
Vorstellung - Ticket	1:n	Jede Vorstellung hat mehrere Tickets, jedes Ticket gehört zu genau einer Vorstellung.
Besucher - Ticket	1:1	Ein Besucher kauft ein Ticket für eine Vorstellung, jedes Ticket gehört zu genau einem Besucher.
Ticket - Verkauf	1:1	Jedes Ticket wird in genau einem Verkaufstransaktion erfasst, jeder Verkauf betrifft genau ein Ticket.
Mitarbeiter - Verkauf	1:n	Ein Mitarbeiter kann mehrere Verkäufe durchführen, jeder Verkauf wird von genau einem Mitarbeiter abgewickelt.

Tabelle 1: RAP4Cinema - Beziehungen zwischen den Tabellen

Das folgende Diagramm zeigt das Entity-Relationship-Modell des Projekts RAP4Cinema. Es veranschaulicht die Entitäten, ihre Attribute sowie die Beziehungen zwischen den zentralen Geschäftsobjekten des Kino-Management-Systems.

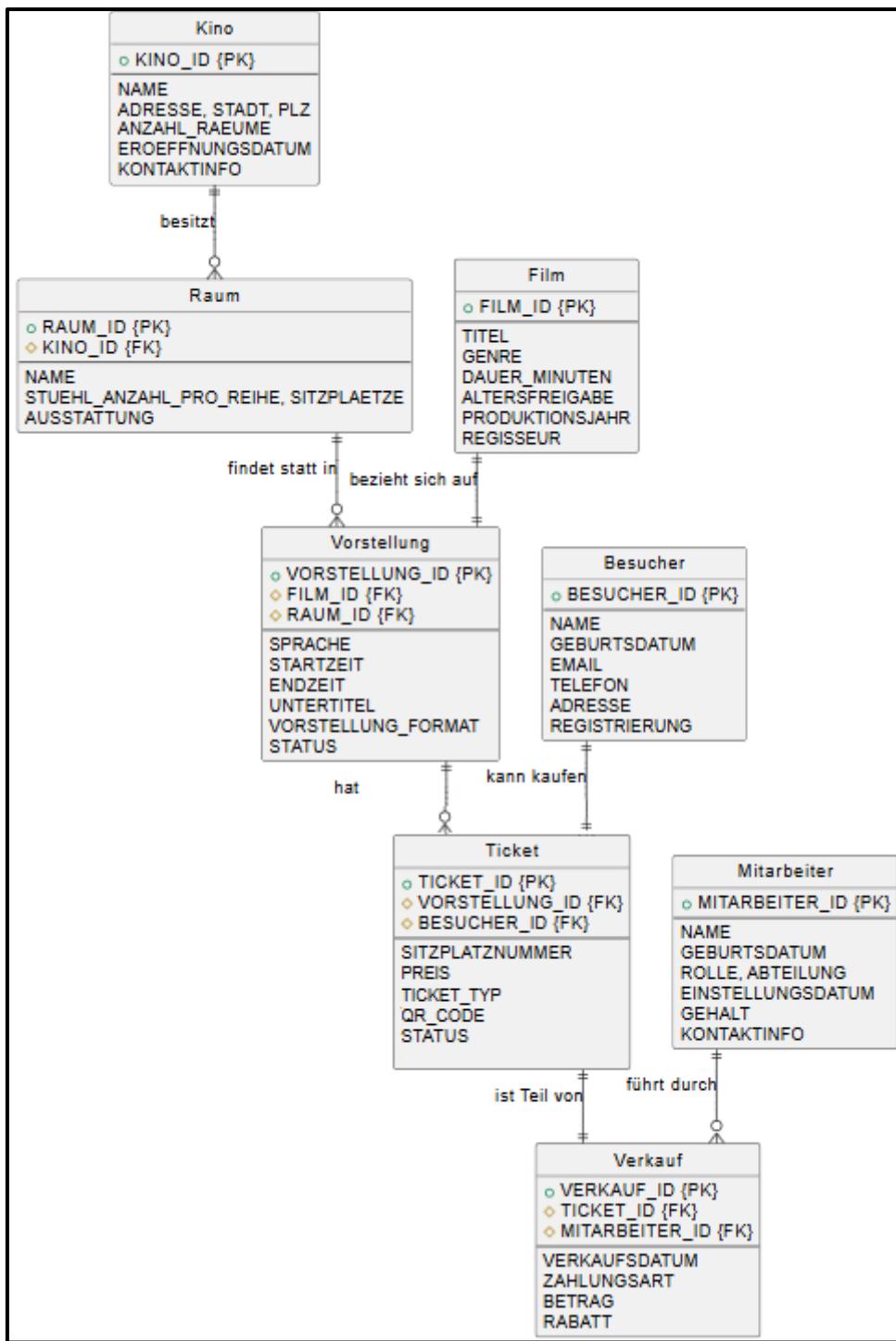


Abbildung 1: RAP4Cinema - ER-Modell

## 2.2. Klassendiagramm

Das folgende Klassendiagramm beschreibt die grundlegende Struktur des Systems (RAP4Cinema), seine zentralen Klassen mit Attributen und Methoden sowie die

Beziehungen zwischen diesen Klassen. Es dient als Modellierung der objektorientierten Architektur und zeigt, wie die verschiedenen Objekte miteinander interagieren.

Die Klasse Kino repräsentiert ein einzelnes Kino innerhalb des Systems. Sie enthält Attribute wie die Kino-ID, den Namen, die Adresse, die Stadt, die Postleitzahl, die Anzahl der Räume, das Eröffnungsdatum und die Kontaktinformationen. Über ihre Methoden können die mit dem Kino verbundenen Räume abgerufen oder hinzugefügt werden, außerdem ist es möglich, die allgemeinen Kinodaten zu aktualisieren.

Ein Kino besteht aus mehreren Räumen, die durch die Klasse Raum modelliert werden. Diese Klasse umfasst Attribute wie Raum-ID, Name, Sitzplatzzahl und Ausstattung. Sie stellt Methoden bereit, um Vorstellungen eines bestimmten Raumes abzurufen oder neue Vorstellungen hinzuzufügen. Zudem kann über eine Methode der Sitzplan eines Raumes abgerufen werden.

Die Klasse Film bildet die im Kino gezeigten Filme ab. Sie enthält Attribute wie Film-ID, Titel, Genre, Dauer, Altersfreigabe, Produktionsjahr, Regisseur und Originalsprache. Methoden dieser Klasse ermöglichen es, Filminformationen abzurufen oder Filmdaten zu aktualisieren.

Eine Vorstellung beschreibt die konkrete Aufführung eines Films in einem bestimmten Raum und zu einer bestimmten Zeit. Die Klasse Vorstellung beinhaltet Attribute wie Vorstellung-ID, Start- und Endzeit, Sprache, Untertitel, Format und Status. Über ihre Methoden können die zugehörigen Tickets abgefragt oder hinzugefügt und der Status der Vorstellung geändert werden.

Die Klasse Besucher repräsentiert registrierte Kinobesucher. Sie enthält Attribute wie Besucher-ID, Name, Geburtsdatum, E-Mail-Adresse, Telefonnummer, Wohnadresse und Registrierungsdatum. Besucher können über Methoden Tickets für Vorstellungen kaufen, bereits gekaufte Tickets stornieren oder ihre persönlichen Daten aktualisieren.

Ein Ticket steht für den Kauf oder die Reservierung eines Sitzplatzes in einer bestimmten Vorstellung. In der Klasse Ticket werden Informationen wie Ticket-ID, Sitzplatznummer, Preis, Kaufdatum, Tickettyp, QR-Code und Status gespeichert. Methoden wie validate() dienen der Überprüfung der Gültigkeit, während stornieren() eine Rückabwicklung ermöglicht. Über getTicketInfo() können alle relevanten Ticketdetails abgerufen werden.

Die Klasse Mitarbeiter bildet die Angestellten des Kinos ab. Zu ihren Attributen zählen Mitarbeiter-ID, Name, Geburtsdatum, Rolle, Abteilung, Einstellungsdatum, Gehalt und Kontaktinformationen. Mitarbeiter können Verkäufe durchführen, Tickets im Auftrag von Besuchern stornieren oder ihre eigenen Daten im System aktualisieren.

Die Klasse Verkauf dokumentiert abgeschlossene Transaktionen. Sie enthält Attribute wie Verkaufs-ID, Verkaufsdatum, Zahlungsart, Betrag und gewährten Rabatt. Methoden ermöglichen die Bestätigung eines Verkaufs sowie die Berechnung des endgültigen Zahlbetrags nach Abzug eventueller Rabatte.

Für die Zahlungsabwicklung ist ein eigenes Interface `IZahlung` vorgesehen, das die grundlegenden Zahlungsfunktionen definiert. Es stellt Methoden bereit, um einen Betrag zu bezahlen oder eine Rückerstattung für ein Ticket durchzuführen. Dieses Interface kann von verschiedenen Implementierungen genutzt werden, beispielsweise Barzahlung, Kreditkartenzahlung oder Onlinezahlung.

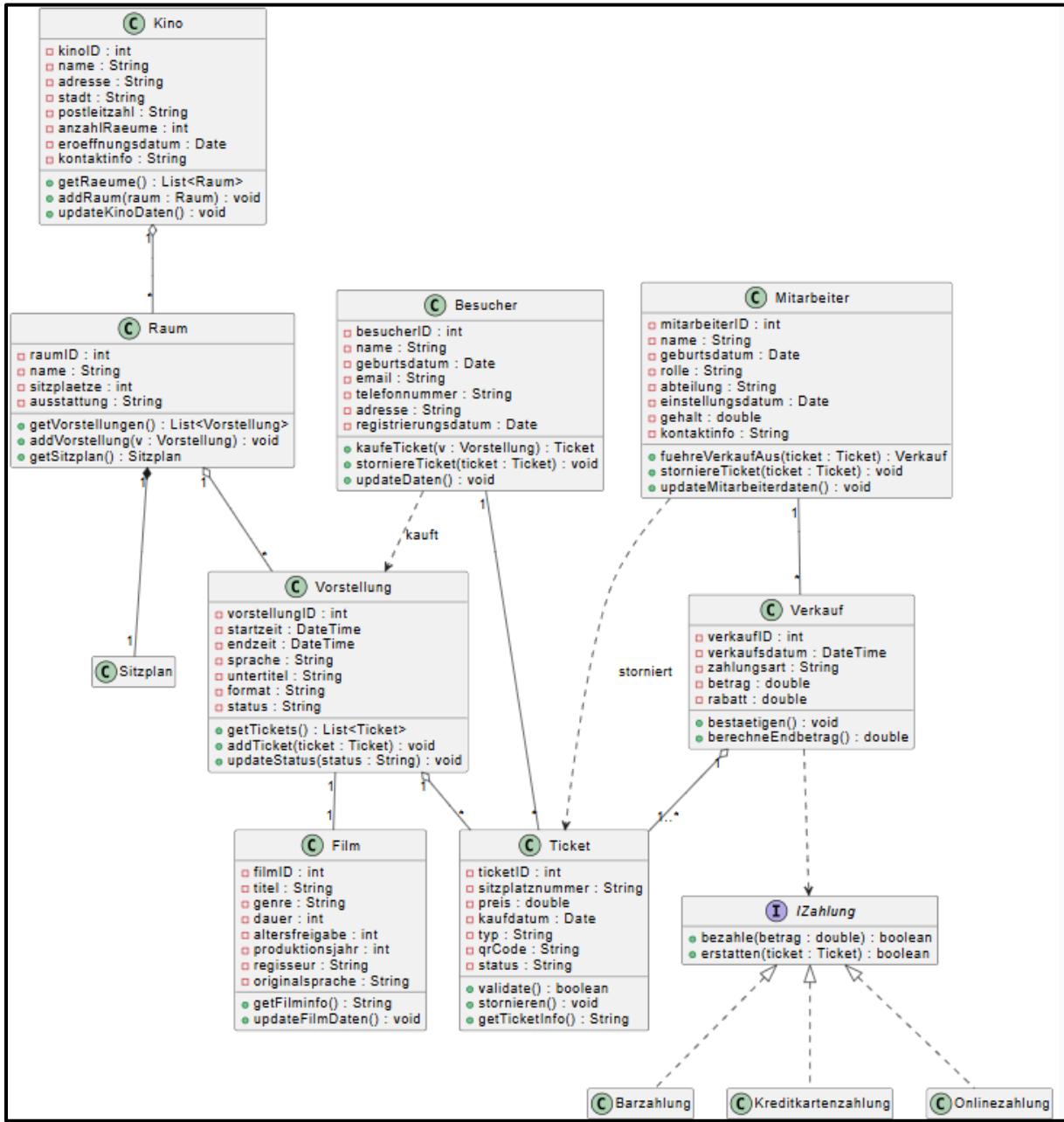


Abbildung 2: RAP4Cinema - Klassendiagramm

## 2.3. Datenflussdiagramm

Das Kino-System (RAP4Cinema) umfasst mehrere externe Akteure, interne Prozesse sowie verschiedene Datenspeicher, die über klar definierte Datenflüsse miteinander verbunden sind.

**Externe Akteure:** Zu den externen Entitäten des Systems gehören Besucher, Mitarbeiter, das Kino-Management sowie externe Zahlungsanbieter.

- Besucher kaufen Tickets, reservieren Sitzplätze, sehen sich Vorstellungen an und verwalten ihre persönlichen Daten.
- Mitarbeiter verkaufen Tickets, bearbeiten Stornierungen und verwalten Verkaufsdaten.
- Das Kino-Management ist für die Verwaltung von Kinos, Räumen, Filmen, Vorstellungen und Mitarbeitern zuständig.
- Externe Zahlungsanbieter übernehmen die Abwicklung von Zahlungen, z. B. über Kreditkarten oder Online-Bezahldienste.

**Hauptprozesse im System:** Innerhalb des Systems werden mehrere zentrale Prozesse ausgeführt:

- Ticketkauf abwickeln: Entgegennahme von Ticketanfragen, Sitzplatzwahlen und Zahlungsinformationen; Erstellung bestätigter Tickets und Weitergabe der Verkaufsdaten.
- Ticket reservieren oder stornieren: Verarbeitung von Reservierungs- oder Stornierungsanfragen und Aktualisierung des Ticketstatus.
- Vorstellungen anzeigen: Bereitstellung einer Übersicht verfügbarer Vorstellungen nach Film, Datum, Sprache oder Raum.
- Film verwalten: Erfassung und Pflege von Filmdaten (Titel, Genre, Dauer, Altersfreigabe, Produktionsjahr, Regisseur).
- Raum verwalten: Verwaltung von Raumdaten wie Name, Sitzplatzzahl und Ausstattung.
- Kino verwalten: Pflege allgemeiner Kinodaten (Name, Adresse, Stadt, Eröffnungsdatum, Kontaktinformationen, Anzahl der Räume).
- Mitarbeiter verwalten: Verwaltung von Mitarbeiterdaten einschließlich ID, Name, Rolle, Abteilung, Gehalt und Kontaktdaten.
- Verkäufe verarbeiten: Erfassung aller Verkaufs- und Zahlungsinformationen sowie Aktualisierung des Ticketstatus (z. B. bezahlt oder storniert).

**Datenspeicher:** Die Prozesse greifen auf verschiedene Datenbanken zu:

- Ticket-Datenbank: Enthält alle Tickets und deren aktuellen Status.
- Vorstellungs-Datenbank: Speichert sämtliche Vorstellungen mit Film-, Raum- und Zeitzuordnung.

- Film-Datenbank: Beinhaltet Filminformationen wie Titel, Dauer, Genre und Altersfreigabe.
- Raum-Datenbank: Verwaltet Details zu Räumen und deren Ausstattung.
- Kino-Datenbank: Enthält Stammdaten der Kinos.
- Besucher- und Mitarbeiter-Datenbanken: Erfassen Benutzer- und Personaldaten.
- Verkaufs-Datenbank: Dokumentiert alle Transaktionen und zugehörigen Zahlungsdetails.

**Datenflüsse:** Die Datenflüsse zwischen den Komponenten verlaufen wie folgt:

- Besucher senden Anfragen an den Prozess Ticketkauf abwickeln, der mit der Ticket- und Verkaufs-Datenbank interagiert.
- Reservierungen und Stornierungen führen zu Aktualisierungen in der Ticket-Datenbank.
- Besucher, die Vorstellungen anzeigen lassen, greifen über den entsprechenden Prozess auf die Vorstellungs-Datenbank zu und erhalten die Ergebnisse direkt zurück.
- Mitarbeiter verarbeiten Verkäufe, wodurch sowohl die Verkaufs- als auch die Ticket-Datenbank aktualisiert werden.
- Das Kino-Management pflegt Filme, Räume, Kinos und Mitarbeiter, wobei die jeweiligen Datenbanken aktualisiert werden.
- Im Rahmen der Verkaufsverarbeitung findet ein Datenaustausch mit externen Zahlungsanbietern statt, um Zahlungen sicher abzuwickeln und die Verkaufsdaten im System zu speichern.

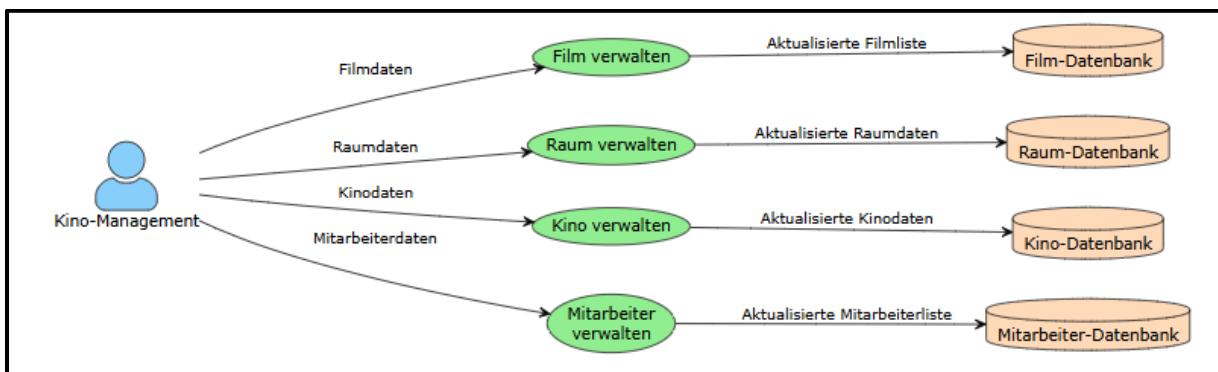


Abbildung 3: RAP4Cinema - Datenflussdiagramm (Kino-Management)

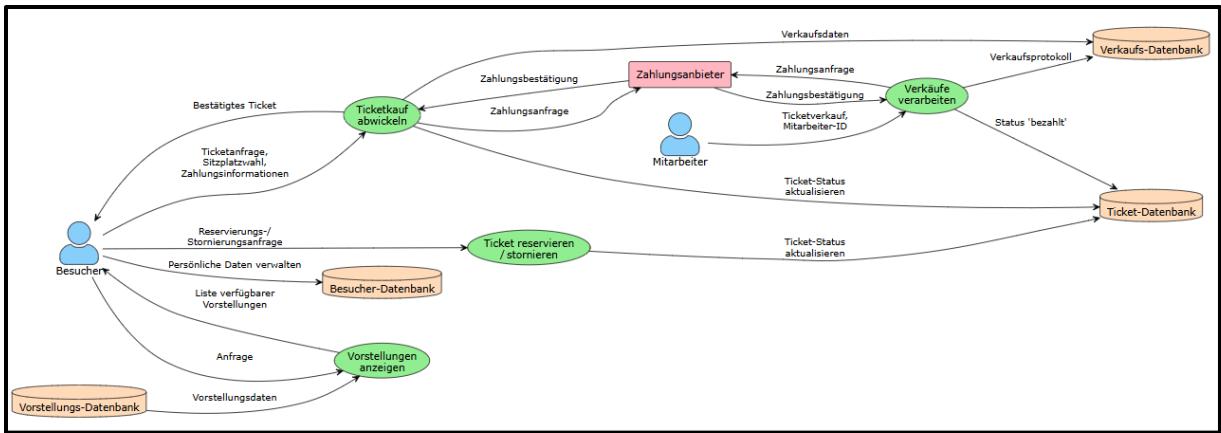


Abbildung 4: RAP4Cinema - Datenflussdiagramm (Besucher & Mitarbeiter)

## 2.4. Use-Case-Diagramm

Im Rahmen des Projekts RAP4Cinema wird ein Kino-Verwaltungssystem entwickelt, das zentrale Geschäftsprozesse wie Ticketverkauf, Reservierung und Stammdatenpflege digital abbildet. Die folgende Use-Case-Diagramme des Kino-Systems (RAP4Cinema) beschreiben die Interaktionen zwischen den verschiedenen Akteuren und dem System selbst. Dabei werden die wesentlichen Anwendungsfälle abgebildet, die das Verhalten und die Funktionalität des Systems aus Sicht der Benutzer veranschaulichen. Zu den Akteuren gehören der Besucher, der Mitarbeiter und der Systemadministrator. Der Besucher interagiert mit dem System, um Tickets zu kaufen oder Reservierungen vorzunehmen. Er kann sich registrieren, ein persönliches Benutzerkonto anlegen und seine persönlichen Daten wie Adresse, E-Mail oder Telefonnummer verwalten. Der Mitarbeiter nutzt das System zur Durchführung von Ticketverkäufen und Stornierungen sowie zur Abwicklung von Zahlungen. Darüber hinaus kann er Rabatte oder Sonderpreise gewähren, etwa im Rahmen spezieller Aktionen. Der Systemadministrator hat die umfassendsten Rechte im System. Er verwaltet sämtliche Stammdaten, darunter Filme, Vorstellungen, Kinos und Räume, und ist für die Konfiguration und Pflege des gesamten Systems verantwortlich. Das Use-Case-Diagramm umfasst eine Reihe von Anwendungsfällen, die sich den jeweiligen Akteuren zuordnen lassen. Für Besucher stehen mehrere grundlegende Funktionen zur Verfügung. Ein Besucher kann sich im System registrieren, indem er ein persönliches Konto mit seinen Daten anlegt. Nach erfolgreicher Registrierung ist eine Anmeldung möglich, um auf weitere Systemfunktionen zuzugreifen. Besucher können Filme anzeigen lassen und Vorstellungen nach bestimmten Kriterien wie Datum, Kino, Film oder Sprache suchen. Sie können Tickets für eine gewünschte Vorstellung kaufen und dabei Sitzplätze auswählen. Alternativ ist es möglich, Tickets zunächst zu reservieren, ohne sie sofort zu bezahlen. Auch die Stornierung eines bereits erworbenen oder reservierten Tickets ist vorgesehen, sofern die Systemrichtlinien dies zulassen. Darüber hinaus können Besucher ihre persönlichen Daten jederzeit anpassen, beispielsweise bei Änderungen von Adresse, Telefonnummer oder E-Mail-Adresse. Mitarbeiter haben eigene Anwendungsfälle, die

vor allem den operativen Kinobetrieb betreffen. Sie verkaufen Tickets direkt an der Kinokasse und können auf Wunsch der Besucher Stornierungen durchführen. Im Rahmen der Zahlungsabwicklung erfassen sie die gewählte Zahlungsart, etwa Barzahlung, Kartenzahlung oder Online-Bezahlung. Bei Bedarf können sie außerdem Rabatte oder Sonderpreise anwenden, um Aktionen oder Preisnachlässe korrekt abzubilden. Der Systemadministrator besitzt administrative Zugriffsrechte und ist für das gesamte Stammdatenmanagement verantwortlich. Er kann neue Kinos im System anlegen oder bestehende Einträge bearbeiten, einschließlich Name, Adresse und Anzahl der Räume. Ebenso verwaltet er die einzelnen Räume, wobei Sitzplätze und Ausstattungsmerkmale hinterlegt werden. Die Pflege der Filmdaten umfasst Angaben wie Titel, Dauer, Genre und Altersfreigabe. Darüber hinaus plant der Administrator Vorstellungen, indem er Filme bestimmten Räumen zuordnet und Start- sowie Endzeiten definiert. Schließlich verwaltet er die Mitarbeiter, legt neue Benutzerkonten an, weist Rollen zu und aktualisiert relevante Personaldaten. Die Systemgrenze des Kino-Systems umfasst alle zentralen Geschäftsprozesse rund um die Ticketverwaltung (Kauf, Reservierung und Stornierung), die Verwaltung von Besuchern und Mitarbeitern sowie das Management von Filmen, Räumen und Vorstellungen. Auch die Verkaufs- und Zahlungsabwicklung sind Teil des Systems. Externe Systeme wie Zahlungsanbieter, etwa Kreditkartenunternehmen oder Online-Bezahldienste, sind nicht Teil des eigentlichen Kino-Systems, können jedoch als eigenständige Akteure im Use-Case-Diagramm dargestellt werden, wenn die Integration mit diesen externen Diensten visualisiert werden soll.

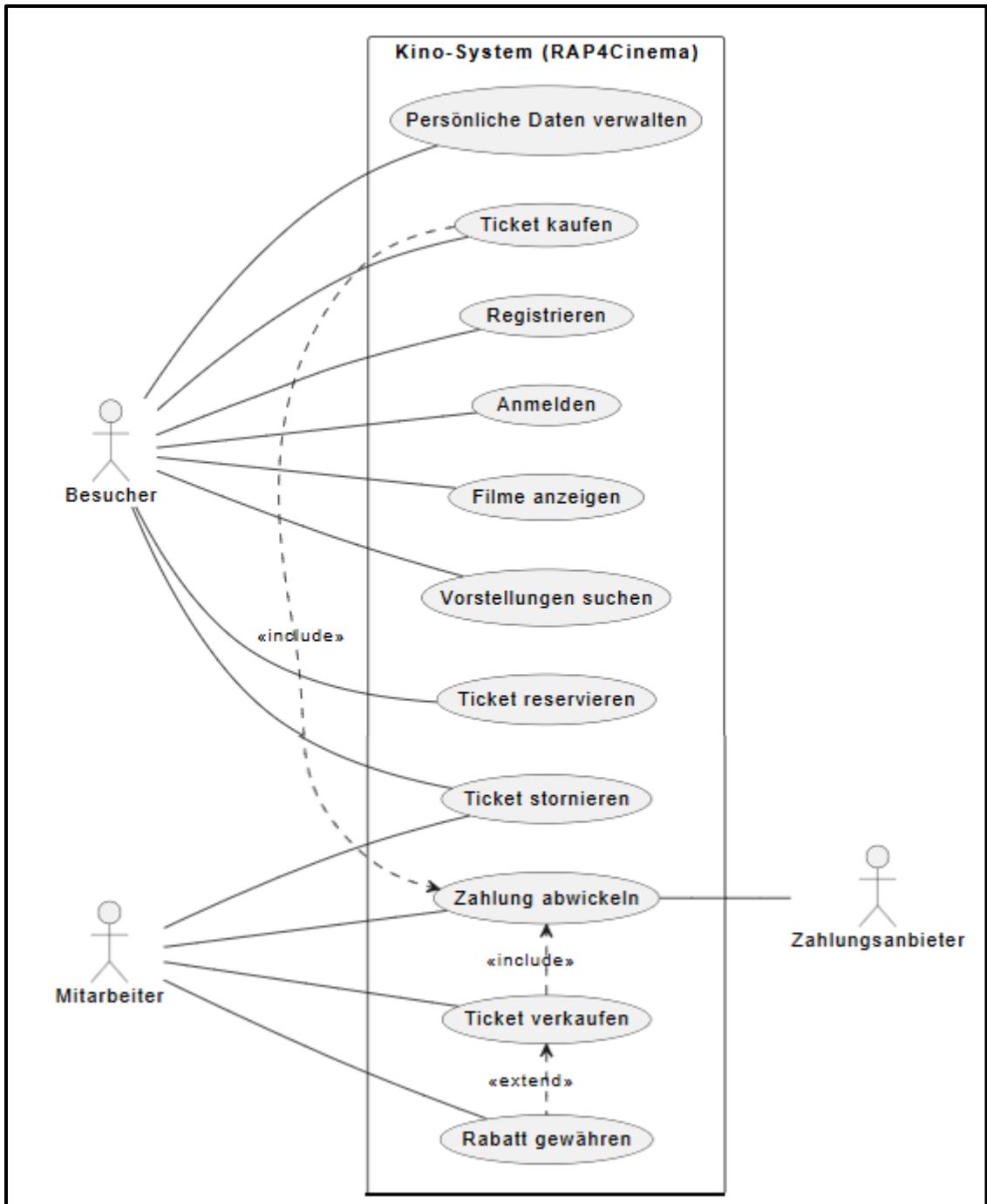


Abbildung 5: Use-Case-Diagramm (Mitarbeiter, Besucher und Zahlungsanbieter)

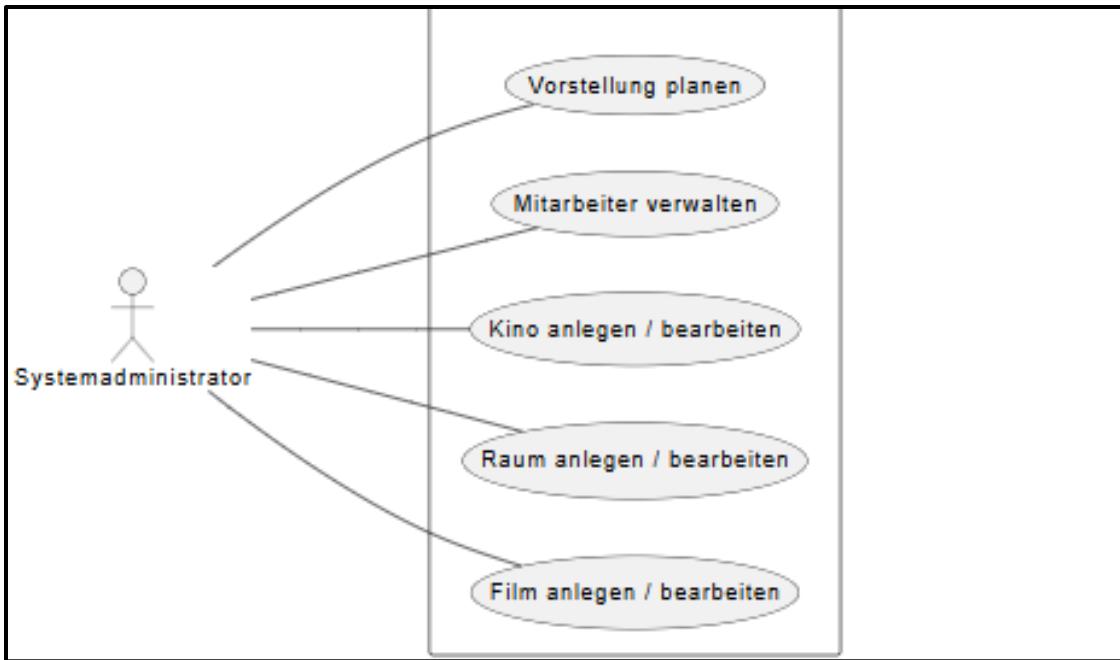


Abbildung 6: Use-Case-Diagramm (Systemadministrator)

## 2.5. Zustandsdiagramm

Im Rahmen des Projekts RAP4Cinema werden verschiedene Geschäftsprozesse eines Kinos digital abgebildet und automatisiert. Um die Abläufe innerhalb des Systems klar zu strukturieren und deren Dynamik zu veranschaulichen, wurden Zustandsdiagramme erstellt. Diese Diagramme zeigen, in welchen Zuständen sich zentrale Entitäten wie Tickets, Vorstellungen, Besucher, Mitarbeiter, Verkaufsprozesse und Räume befinden können, sowie die Übergänge, die durch bestimmte Aktionen oder Ereignisse ausgelöst werden. Die Zustandsdiagramme dienen als Grundlage für die technische Umsetzung im RESTful ABAP Programming Model (RAP). Sie unterstützen sowohl die Modellierung der Geschäftslogik als auch das Verständnis des Systemverhaltens während des gesamten Lebenszyklus einer Entität. Durch diese visuelle Darstellung werden Abhängigkeiten, Prozessschritte und Systemreaktionen transparent und ermöglichen eine konsistente Implementierung der Anwendungslogik im RAP-Framework.

Das folgende Zustandsdiagramm veranschaulicht den Lebenszyklus eines Tickets. Ein Ticket wird zunächst erstellt, sobald es im System angelegt wird. Anschließend kann es reserviert werden, wodurch der zugehörige Platz vorläufig blockiert ist, während die Zahlung noch aussteht. Nach erfolgreicher Bezahlung wechselt der Status zu bezahlt, womit das Ticket gültig und für den Einlass berechtigt ist. Beim Eintritt zur Veranstaltung wird das Ticket eingelöst und gilt damit als verwendet. Alternativ kann ein Ticket, solange es sich im reservierten oder bezahlten Zustand befindet, storniert

werden. Wenn die Veranstaltung endet, ohne dass das Ticket genutzt wurde, wechselt es automatisch in den Zustand abgelaufen.

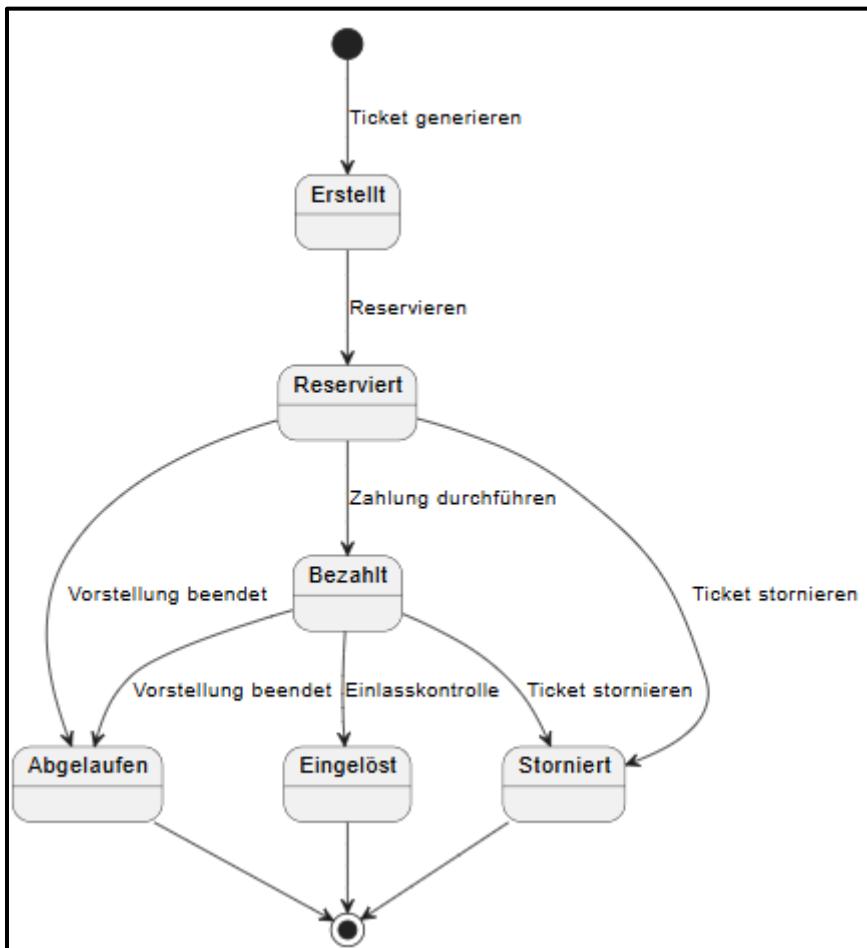


Abbildung 7: Zustandsdiagramm - Ticket-Lebenszyklus

Das folgende Zustandsdiagramm beschreibt die Interaktion eines Besuchers im Kino-System. Ein Besucher befindet sich zunächst im Zustand unregistriert, solange er dem System nicht bekannt ist. Legt der Besucher ein Konto an, wechselt der Status zu registriert. Meldet er sich anschließend im System an, geht der Zustand in eingeloggt über. Während der Besucher Aktionen wie den Kauf eines Tickets oder die Änderung seiner persönlichen Daten durchführt, befindet er sich im Zustand aktiv. Meldet er sich ab oder bleibt über einen bestimmten Zeitraum inaktiv, wird der Status zu inaktiv geändert.

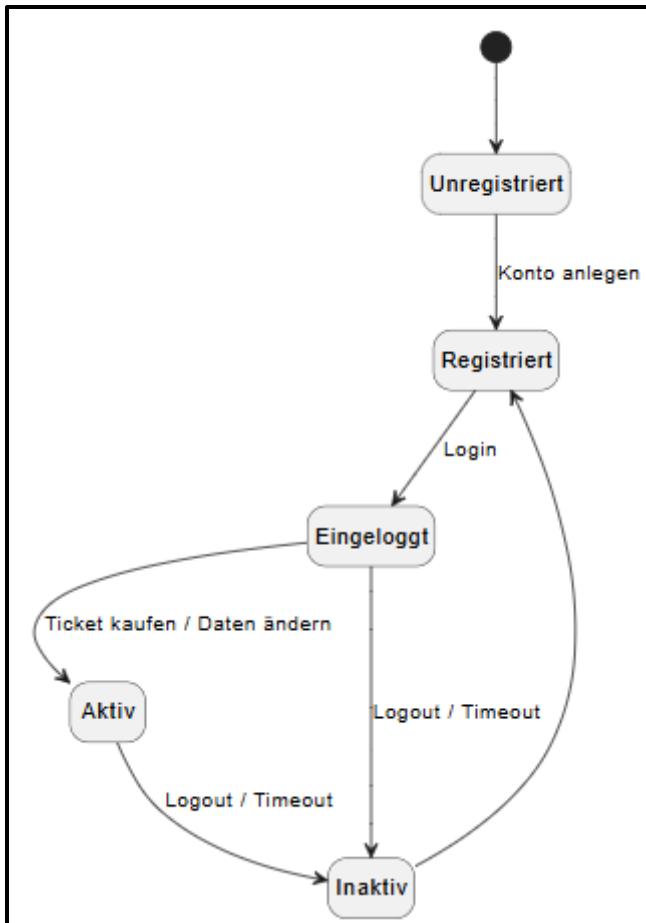


Abbildung 8: Zustandsdiagramm - Besucher

Das folgende Zustandsdiagramm beschreibt den Ablauf eines Verkaufsprozesses im Kino-System. Ein Verkaufsvorgang beginnt im Zustand offen, sobald der Kunde den Kaufprozess startet. Nach Eingabe der Zahlungsinformationen wechselt der Status zu in Bearbeitung, da die Zahlungsdaten geprüft werden. Wenn die Zahlung erfolgreich abgeschlossen wird, ändert sich der Zustand zu bestätigt, womit der Verkauf als abgeschlossen gilt. Schlägt die Zahlung hingegen fehl, geht der Prozess in den Zustand fehlgeschlagen über. Wird der Verkaufsvorgang vom Kunden oder System vorzeitig beendet, befindet sich der Verkauf im Zustand storniert.

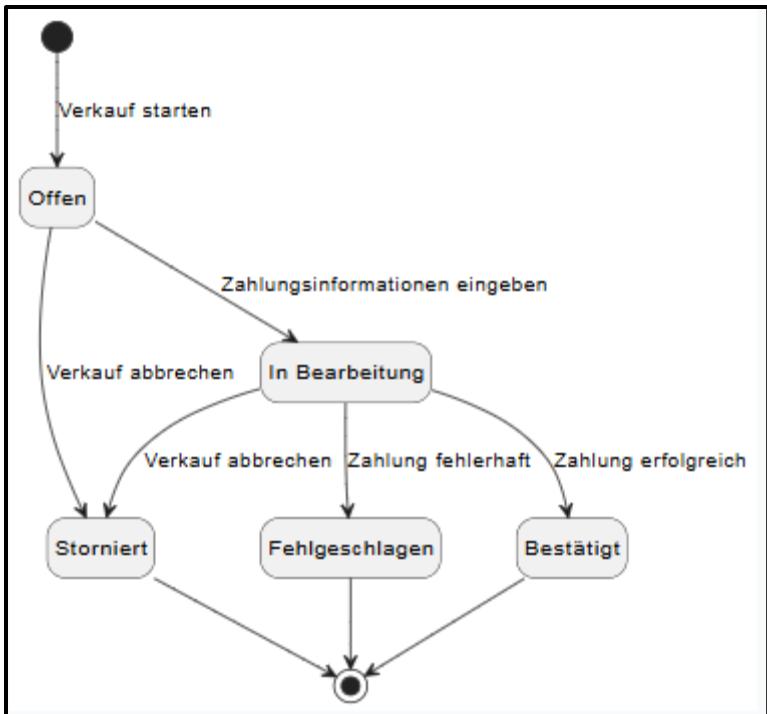


Abbildung 9: Zustandsdiagramm - Verkauf

Das folgende Zustandsdiagramm beschreibt den Lebenszyklus einer Filmvorstellung im Kino-System. Eine Vorstellung befindet sich zunächst im Zustand geplant, sobald sie im System angelegt wurde, jedoch noch nicht für den Ticketverkauf freigegeben ist. Wird der Verkauf geöffnet, wechselt der Status zu buchbar, wodurch Kunden Tickets reservieren oder kaufen können. Sobald alle Plätze vergeben sind, geht die Vorstellung in den Zustand ausverkauft über. Mit Beginn der Aufführung ändert sich der Status zu gestartet, und nach deren Abschluss zu beendet. Sollte die Vorstellung aus organisatorischen oder technischen Gründen nicht stattfinden, wird sie abgesagt, unabhängig davon, ob sie sich zuvor im geplanten, buchbaren oder ausverkauften Zustand befand.

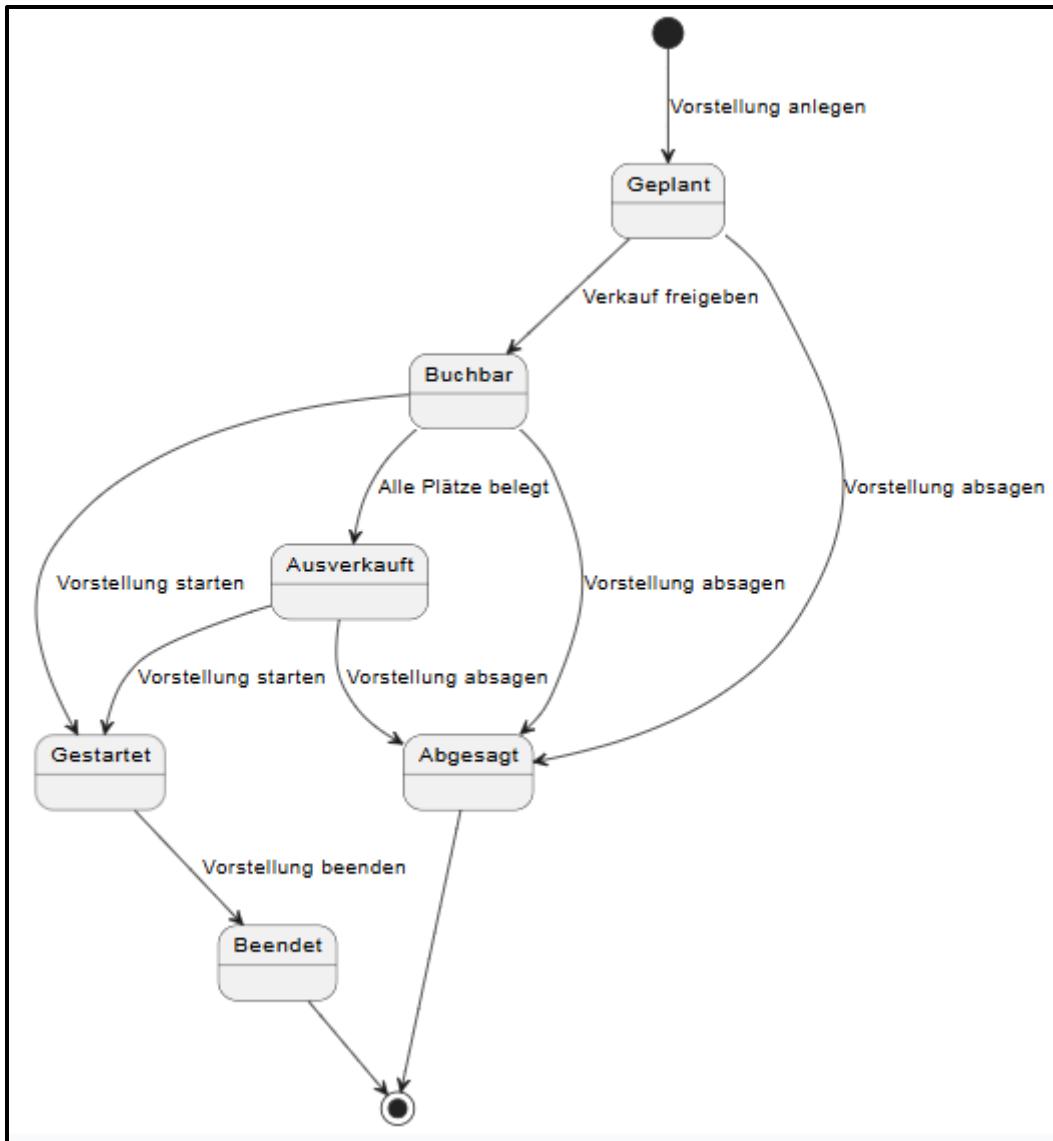


Abbildung 10: Zustandsdiagramm - Aufführung eines Films

Das folgende Zustandsdiagramm beschreibt den Lebenszyklus eines Kinoraums innerhalb des Systems. Ein Raum ist zunächst verfügbar, sobald er im System angelegt wurde und für Vorstellungen genutzt werden kann. Wird in diesem Raum eine Vorstellung gestartet, ändert sich der Zustand zu belegt, da der Raum aktuell genutzt wird. Nach dem Ende der Vorstellung wechselt der Status wieder zu verfügbar. Soll der Raum aufgrund technischer Arbeiten oder Reinigung vorübergehend gesperrt werden, wird er in den Zustand in Wartung versetzt. Während dieser Zeit ist keine Nutzung möglich. Nach Abschluss der Wartungsarbeiten wird der Raum wieder verfügbar und steht erneut für Vorstellungen zur Verfügung.

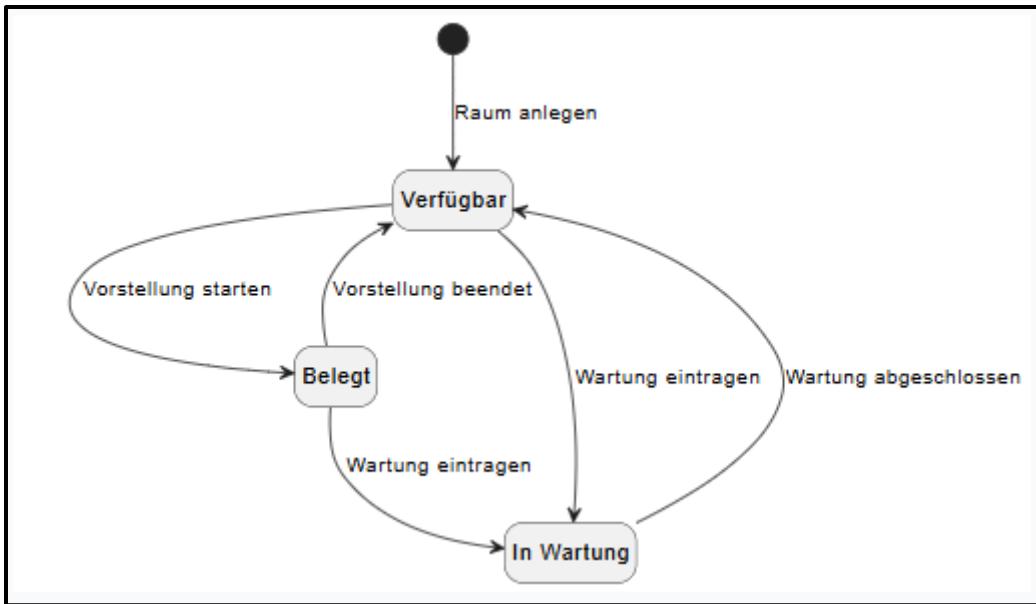


Abbildung 11: Zustandsdiagramm - Verwaltung

Das folgende Zustandsdiagramm zeigt den Arbeitsprozess eines Mitarbeiters im Kino-System. Ein Mitarbeiter befindet sich zunächst im Zustand angestellt, sobald er im System erfasst wurde. Meldet sich der Mitarbeiter am System an, wechselt der Status zu eingeloggt. Sobald er aktive Tätigkeiten wie den Verkauf von Tickets oder die Durchführung administrativer Aufgaben übernimmt, geht der Zustand in aktiv über. Nach Abschluss der Arbeit oder bei einer Pause meldet sich der Mitarbeiter ab und befindet sich damit im Zustand inaktiv. Verlässt der Mitarbeiter schließlich das Unternehmen, wechselt der Status zu ausgeschieden, womit kein Zugriff oder keine Tätigkeit im System mehr möglich ist.

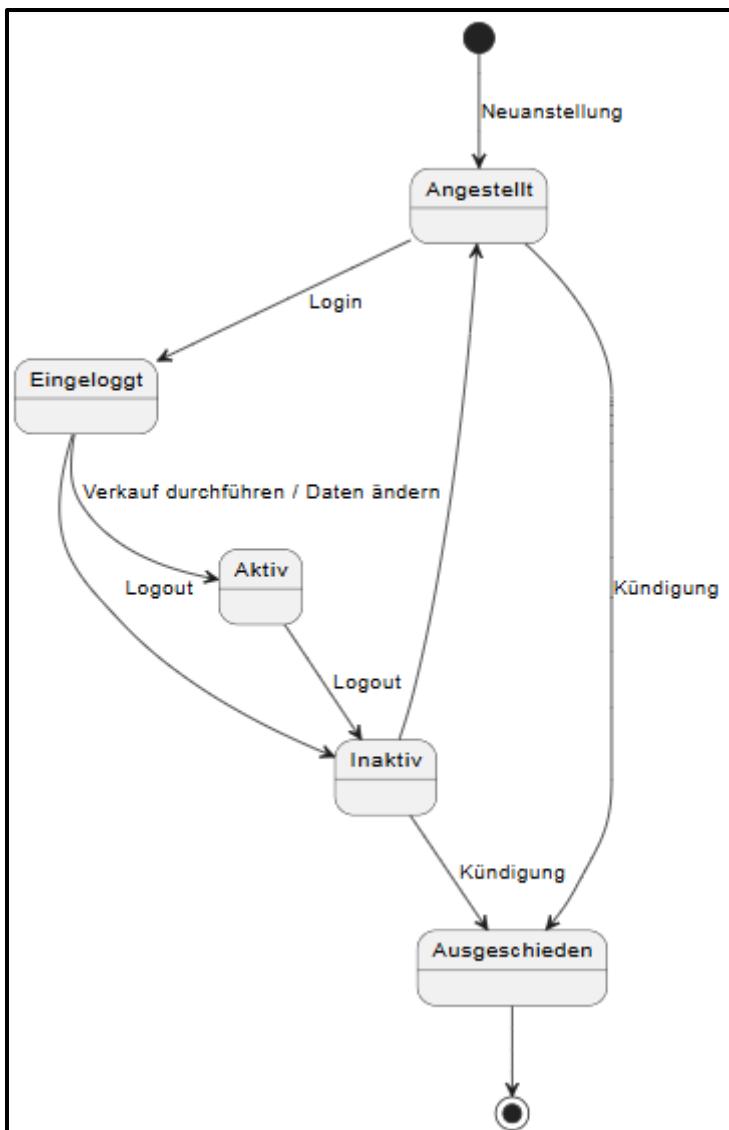


Abbildung 12: Zustandsdiagramm - Mitarbeiter (Arbeitsprozess)

### 3. Technische Umsetzung in SAP RAP

#### 3.1. Packages in SAP RAP

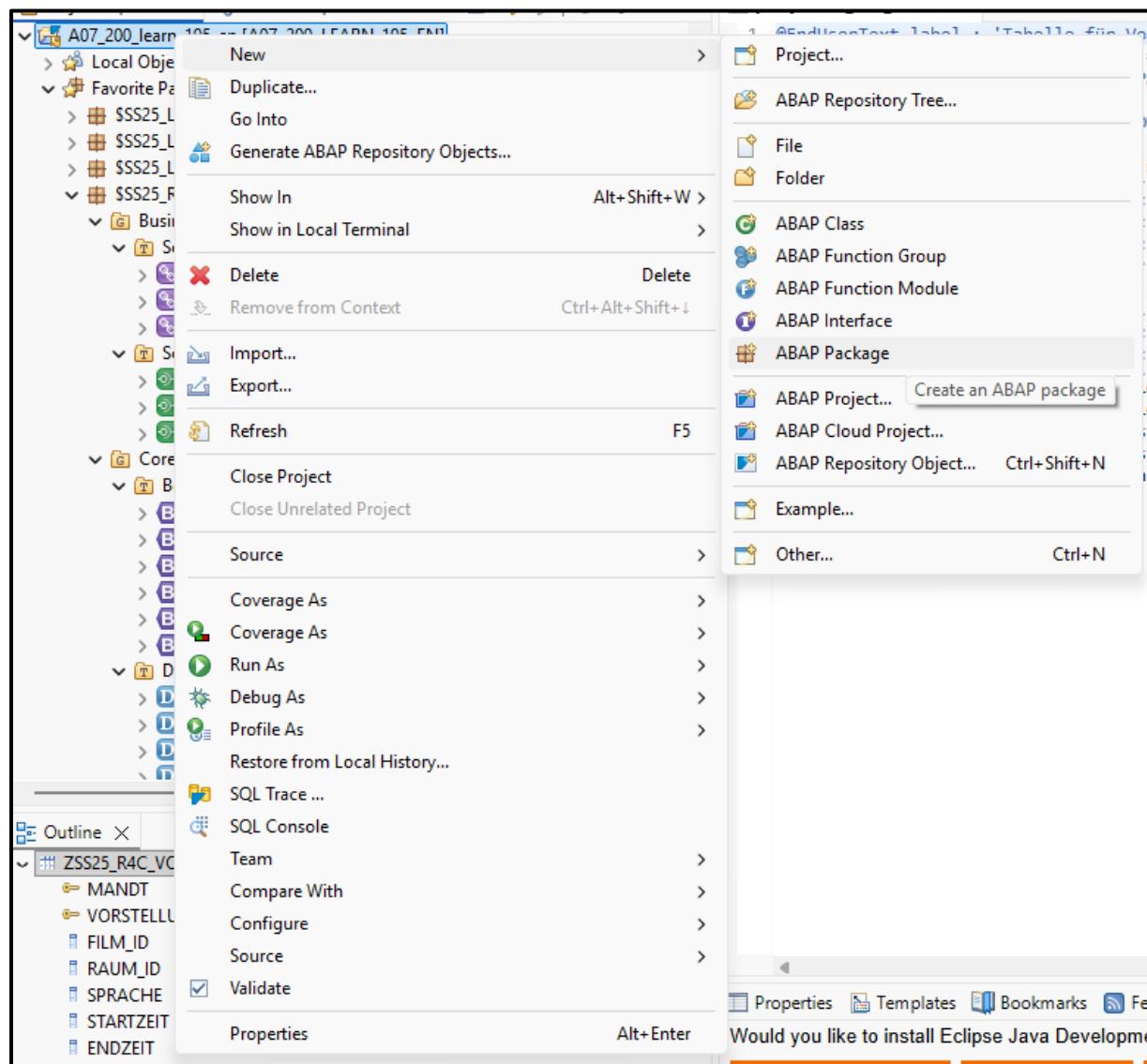
Ein Package im SAP-System ist ein logischer Container für Entwicklungsobjekte und spielt im SAP RAP (RESTful ABAP Programming Model) eine zentrale Rolle. Seine Hauptaufgaben sind:

- **Strukturierung:** Bündelt alle RAP-Objekte (z. B. Tabellen, Views, Behavior Definitions, Service Bindings) in einer klaren Projektstruktur für bessere Übersicht und Wartbarkeit.
- **Transport:** Dient als Transportcontainer, um alle enthaltenen Objekte gemeinsam in andere Systeme (z. B. Test- oder Produktivsysteme) zu übertragen.

- **Autorisierung:** Steuert Zugriffsrechte auf Entwicklungsobjekte über die Package-Ebene.
- **Namensraum & Wiederverwendbarkeit:** Definiert einen eindeutigen Namensraum (z. B. \$SS25\_RAP4CINEMA) und ermöglicht die Wiederverwendung von Objekten in mehreren Projekten.
- **Übersicht in Eclipse/ADT:** Stellt alle Objekte hierarchisch dar und erleichtert so die Navigation und Projektübersicht.

### Schritt-für-Schritt-Anleitung:

Rechtsklick auf den Knoten des ABAP-Systems → New → ABAP Package → Name und Beschreibung eingeben, z. B. Name: \$SS25\_RAP4CINEMA || Beschreibung: RAP-Projekt RAP4Cinema → Next → Next → Finish



New ABAP Package

**ABAP Package**

Create an ABAP package.

Project: \* A07\_200\_learn-195\_en

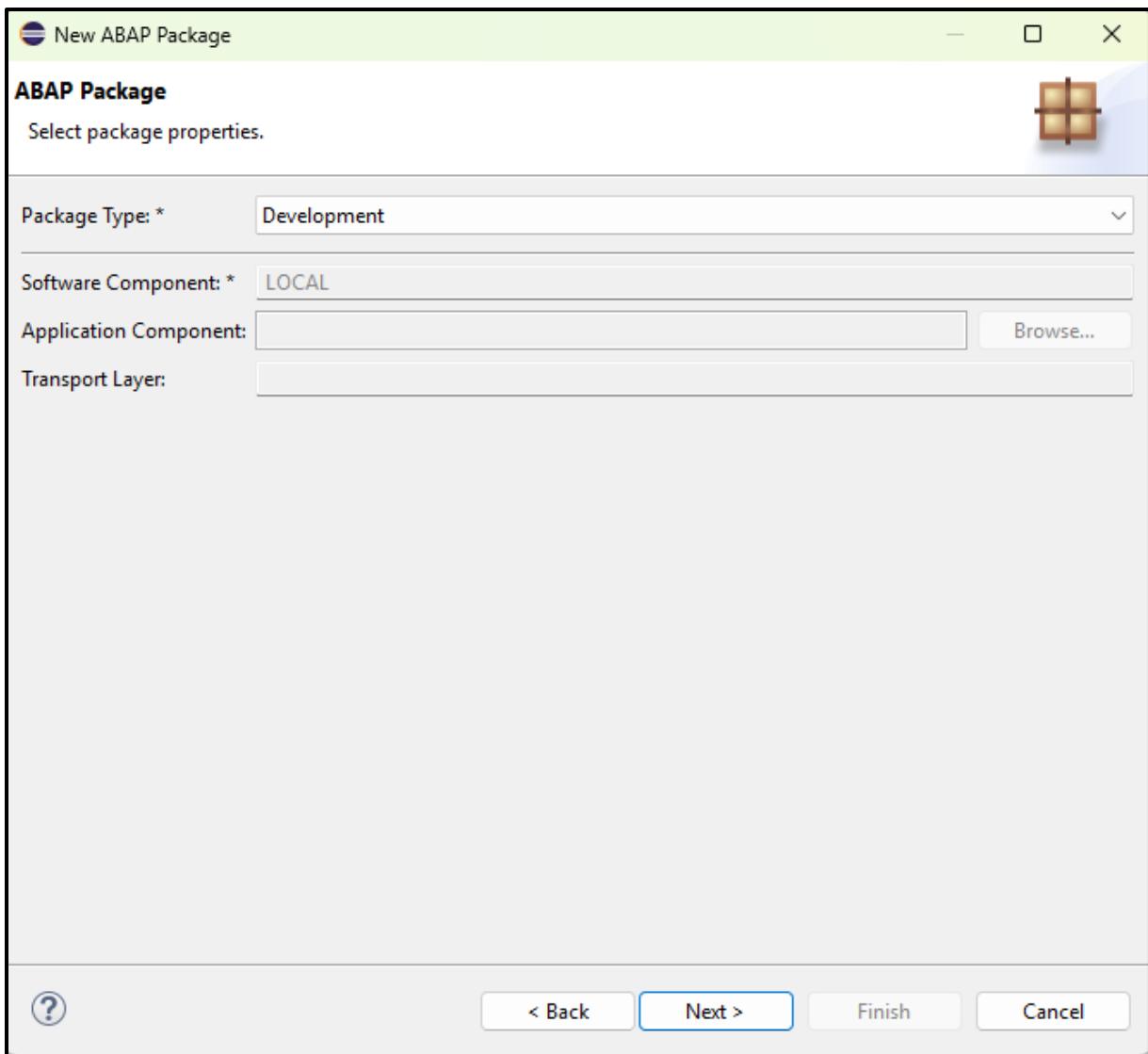
Name: \* SSS25\_RAP4CINEMA

Description: \* RAP-Projekt RAP4Cinema

Original Language: EN

Add to favorite packages

Superpackage:



New ABAP Package

### Select Transport Request

(i) No change recording enabled for package

Choose from requests in which I am involved

Create a new request

Request Description: \*

CTS Project:  Browse...

Enter a request number

Request Number:  Browse...

? < Back Next > Finish Cancel

## 3.2. Database Tables

Die physische Datenhaltung erfolgt in der ABAP-Datenbank, in der alle relevanten Informationen des Kino-Management-Systems dauerhaft gespeichert werden - etwa zu Kinos, Räumen, Filmen, Vorstellungen, Besuchern, Tickets, Mitarbeitern und Verkäufen. Jede dieser Tabellen bildet die technische Grundlage der entsprechenden Business-Entität.

Im SAP RAP (RESTful ABAP Programming Model) dienen Datenbanktabellen als persistente Datenhaltungsschicht. Sie bilden das Fundament für alle weiteren Schichten des RAP-Modells, insbesondere für:

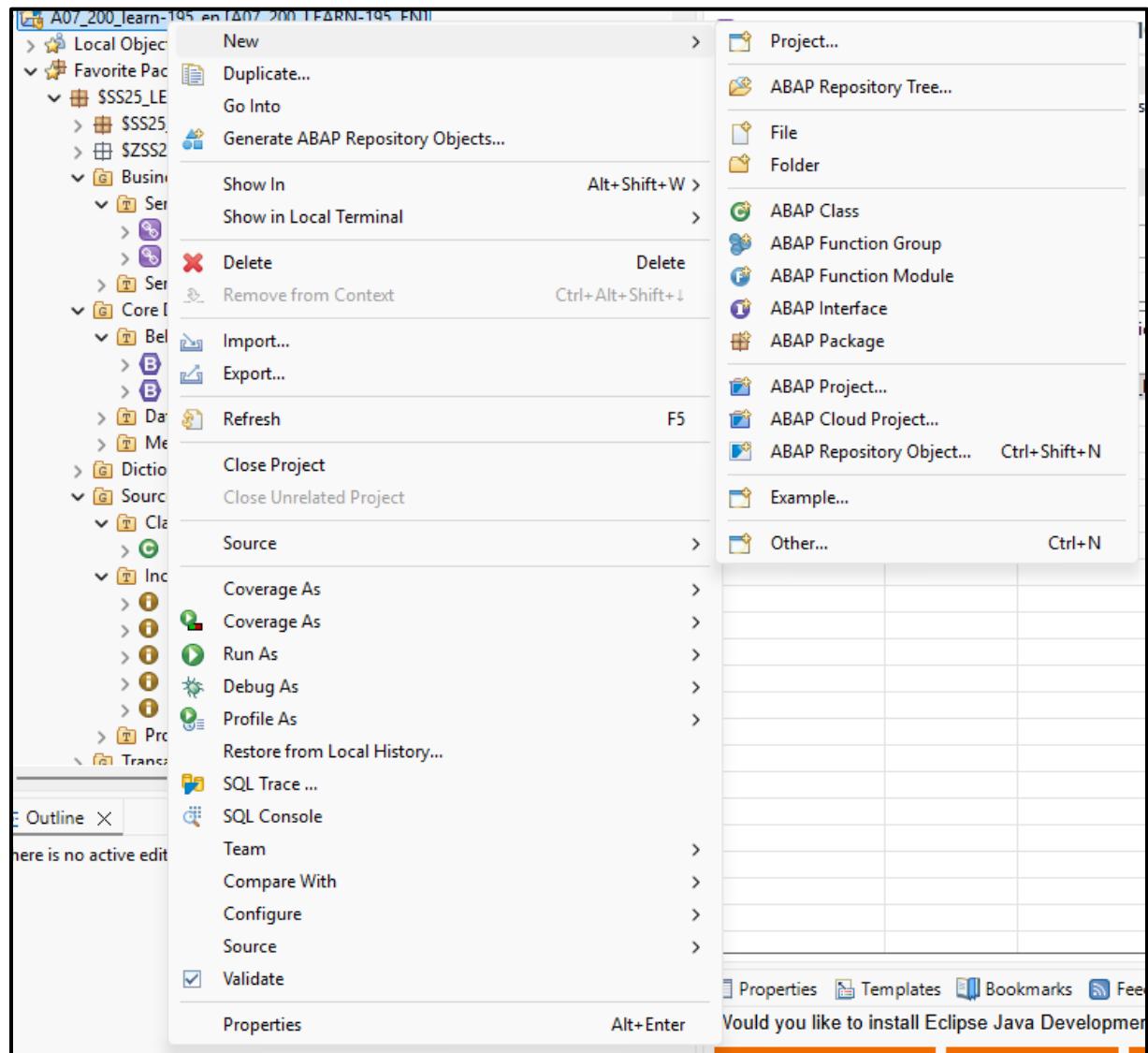
- **Business Objects (BOs):** Modellieren die logischen Geschäftsobjekte auf Basis der zugrunde liegenden Tabellen.

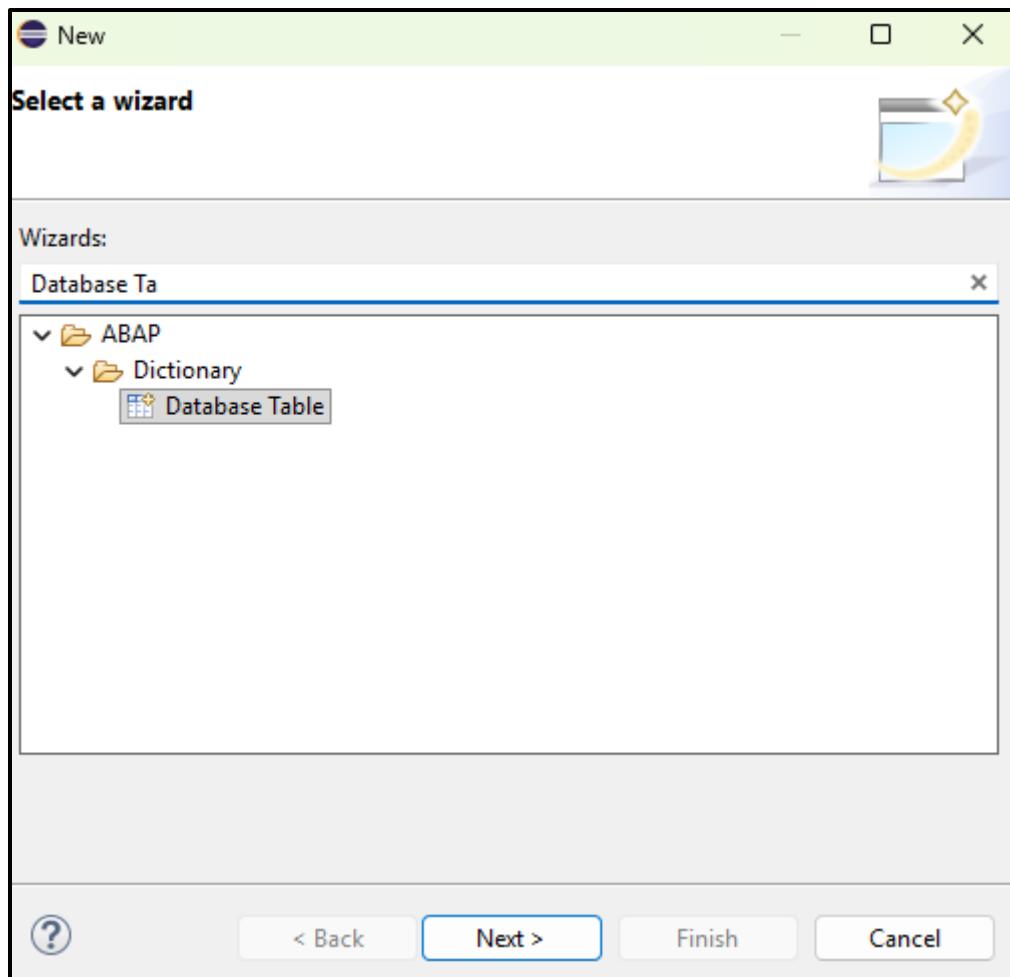
- **Behaviors:** Definieren zulässige Operationen wie Erstellen, Ändern oder Löschen von Datensätzen sowie geschäftsspezifische Aktionen.
- **Service Views:** Stellen Daten modular nach außen bereit, beispielsweise über OData V4 Services für Fiori/UI5-Anwendungen.

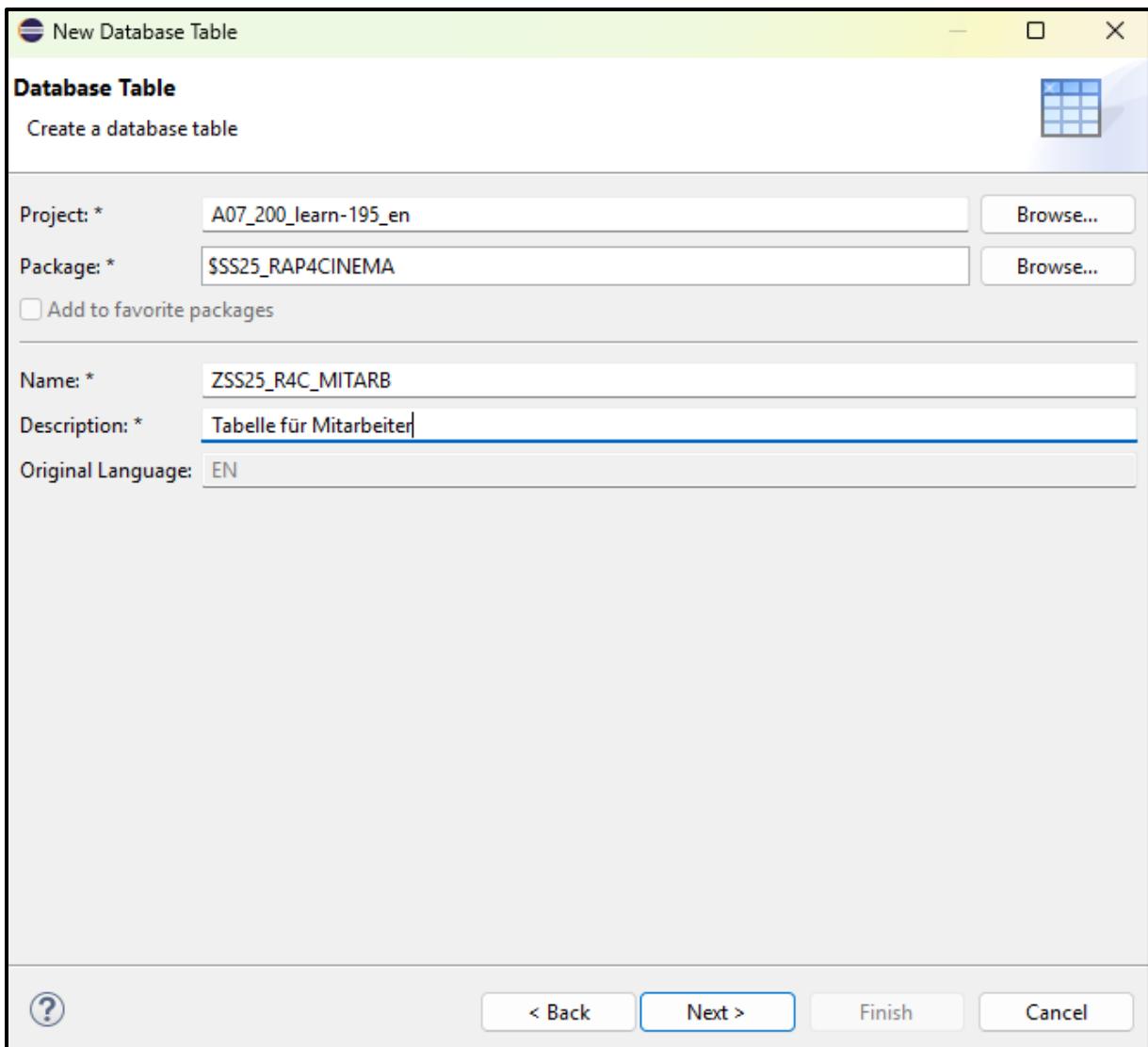
Durch diese klare Trennung zwischen Datenhaltung, Geschäftslogik und Servicebereitstellung wird eine saubere, erweiterbare Architektur gewährleistet - im Sinne des Clean Architecture-Ansatzes des Projekts RAP4Cinema.

### Schritt-für-Schritt-Anleitung:

Im Project Explorer → Rechter Klick auf dein Package → New → Other... → Database Table auswählen → Name und Beschreibung eingeben, z. B. Name: ZSS25\_R4C\_MITARB || Beschreibung: Tabelle für Mitarbeiter → Next → Finish







New Database Table

Select Transport Request

(i) No change recording enabled for package \$SS25\_RAP4CINEMA

Choose from requests in which I am involved

Create a new request

Request Description: \*

CTS Project:

Enter a request number

Request Number:

- Für **Mitarbeiter** (ZSS25\_R4C\_MITARB) stellt sich der Inhalt der Tabelle wie folgt dar

### [A07] ZSS25\_R4C\_MITARB

```
1 @EndUserText.label : 'Tabelle für Mitarbeiter'
2 @AbapCatalog.enhancement.category : #NOT_EXTENSIBLE
3 @AbapCatalog.tableCategory : #TRANSPARENT
4 @AbapCatalog.deliveryClass : #A
5 @AbapCatalog.dataMaintenance : #ALLOWED
6 define table zss25_r4c_mitarb {
7   key mandt          : abap.clnt not null;
8   key mitarbeiter_id : abap.numc(10) not null;
9   name                : abap.char(100);
10  geburtsdatum       : abap.dats;
11  rolle               : abap.char(50);
12  abteilung           : abap.char(50);
13  einstellungsdatum  : abap.dats;
14  gehalt              : abap.dec(10,2);
15  kontaktinfo         : abap.char(200);
16  created_by          : abp_creation_user;
17  created_at          : abp_creation_tstmp;
18  last_changed_by    : abp_locinst_lastchange_user;
19  last_changed_at    : abp_locinst_lastchange_tstmp;
20  local_last_changed_at : abp_lastchange_tstmp;
21 }
22 }
```

- Für **Besucher** (ZSS25\_R4C\_BESUCH) stellt sich der Inhalt der Tabelle wie folgt dar:

### [A07] ZSS25\_R4C\_BESUCH

```
1 @EndUserText.label : 'Tabelle für Besucher'
2 @AbapCatalog.enhancement.category : #EXTENSIBLE_ANY
3 @AbapCatalog.tableCategory : #TRANSPARENT
4 @AbapCatalog.deliveryClass : #A
5 @AbapCatalog.dataMaintenance : #RESTRICTED
6 define table zss25_r4c_besuch {
7   key mandt          : abap.clnt not null;
8   key besucher_id   : abap.numc(10) not null;
9   name                : abap.char(100);
10  geburtsdatum       : abap.dats;
11  email               : abap.char(100);
12  telefon             : abap.char(30);
13  adresse             : abap.char(200);
14  registrierung       : abap.dats;
15  created_by          : abp_creation_user;
16  created_at          : abp_creation_tstmp;
17  last_changed_by    : abp_locinst_lastchange_user;
18  last_changed_at    : abp_locinst_lastchange_tstmp;
19  local_last_changed_at : abp_lastchange_tstmp;
20 }
21 }
```

- Für **Film** (ZSS25\_R4C\_FILM) stellt sich der Inhalt der Tabelle wie folgt dar:

```

[ A07] ZSS25_R4C_FILM X
1 @EndUserText.label : 'Tabelle für Film'
2 @AbapCatalog.enhancement.category : #NOT_EXTENSIBLE
3 @AbapCatalog.tableCategory : #TRANSPARENT
4 @AbapCatalog.deliveryClass : #A
5 @AbapCatalog.dataMaintenance : #ALLOWED
6 define table zss25_r4c_film {
7   key mandt           : abap.clnt not null;
8   key film_id         : abap.numc(10) not null;
9   titel               : abap.char(200);
10  genre               : abap.char(50);
11  dauer_minuten      : abap.int4;
12  altersfreigabe     : abap.int2;
13  produktionsjahr    : abap.numc(4);
14  regisseur          : abap.char(100);
15  created_by          : abp_creation_user;
16  created_at          : abp_creation_tstmp;
17  last_changed_by     : abp_locinst_lastchange_user;
18  last_changed_at     : abp_locinst_lastchange_tstmp;
19  local_last_changed_at : abp_lastchange_tstmp;
20
21 }
```

- Für **Kino** (ZSS25\_R4C\_KINO) stellt sich der Inhalt der Tabelle wie folgt dar:

```

[ A07] ZSS25_R4C_KINO X
1 @EndUserText.label : 'Tabelle für Kino'
2 @AbapCatalog.enhancement.category : #NOT_EXTENSIBLE
3 @AbapCatalog.tableCategory : #TRANSPARENT
4 @AbapCatalog.deliveryClass : #A
5 @AbapCatalog.dataMaintenance : #ALLOWED
6 define table zss25_r4c_kino {
7   key mandt           : abap.clnt not null;
8   key kino_id         : abap.numc(10) not null;
9   name                : abap.char(100);
10  adresse             : abap.char(200);
11  stadt               : abap.char(100);
12  plz                 : abap.char(10);
13  anzahl_raeume       : abap.int4;
14  eroeffnungsdatum    : abap.dats;
15  kontaktinfo         : abap.char(200);
16  created_by          : abp_creation_user;
17  created_at          : abp_creation_tstmp;
18  last_changed_by     : abp_locinst_lastchange_user;
19  last_changed_at     : abp_locinst_lastchange_tstmp;
20  local_last_changed_at : abp_lastchange_tstmp;
21
22 }
```

- Für **Raum** (ZSS25\_R4C\_RAUM) stellt sich der Inhalt der Tabelle wie folgt dar:

```

[ A07] ZSS25_R4C_RAUM ×
1 @EndUserText.label : 'Tabelle für Raum'
2 @AbapCatalog.enhancement.category : #NOT_EXTENSIBLE
3 @AbapCatalog.tableCategory : #TRANSPARENT
4 @AbapCatalog.deliveryClass : #A
5 @AbapCatalog.dataMaintenance : #ALLOWED
6 define table zss25_r4c_raum {
7   key mandt           : abap.clnt not null;
8   key raum_id         : abap.numc(10) not null;
9   kino_id            : abap.numc(10);
10  name               : abap.char(50);
11  stuehl_anzahl_pro_reihe : abap.int4;
12  sitzplaetze       : abap.int4;
13  ausstattung        : abap.char(200);
14  created_by          : abp_creation_user;
15  created_at          : abp_creation_tstmp;
16  last_changed_by    : abp_locinst_lastchange_user;
17  last_changed_at    : abp_locinst_lastchange_tstmp;
18  local_last_changed_at : abp_lastchange_tstmp;
19
20 }
```

- Für **Ticket** (ZSS25\_R4C\_TICKET) stellt sich der Inhalt der Tabelle wie folgt dar:

```

[ A07] ZSS25_R4C_TICKET ×
1 @EndUserText.label : 'Tabelle für Ticket'
2 @AbapCatalog.enhancement.category : #NOT_EXTENSIBLE
3 @AbapCatalog.tableCategory : #TRANSPARENT
4 @AbapCatalog.deliveryClass : #A
5 @AbapCatalog.dataMaintenance : #ALLOWED
6 define table zss25_r4c_ticket {
7   key mandt           : abap.clnt not null;
8   key ticket_id       : abap.numc(10) not null;
9   vorstellung_id     : abap.numc(10) not null;
10  besucher_id        : abap.numc(10) not null;
11  sitzplatznummer   : abap.int4;
12  preis              : abap.dec(10,2);
13  ticket_typ         : abap.char(20);
14  qr_code             : abap.char(100);
15  status              : abap.char(20);
16  created_by          : abp_creation_user;
17  created_at          : abp_creation_tstmp;
18  last_changed_by    : abp_locinst_lastchange_user;
19  last_changed_at    : abp_locinst_lastchange_tstmp;
20  local_last_changed_at : abp_lastchange_tstmp;
21
22 }
```

- Für **Verkauf** (ZSS25\_R4C\_VERKF) stellt sich der Inhalt der Tabelle wie folgt dar:

```

[ A07] ZSS25_R4C_VERKF X
1 @EndUserText.label : 'Tabelle für Verkauf (Faktentabelle)'
2 @AbapCatalog.enhancement.category : #NOT_EXTENSIBLE
3 @AbapCatalog.tableCategory : #TRANSPARENT
4 @AbapCatalog.deliveryClass : #A
5 @AbapCatalog.dataMaintenance : #ALLOWED
6 define table zss25_r4c_verkf {
7   key mandt          : abap.clnt not null;
8   key verkauf_id    : abap.numc(10) not null;
9   ticket_id         : abap.numc(10);
10  mitarbeiter_id   : abap.numc(10);
11  verkaufsdatum    : abap.dats;
12  zahlungsart      : abap.char(30);
13  rabatt           : abap.dec(5,2);
14  created_by        : abp_creation_user;
15  created_at        : abp_creation_tstmp;
16  last_changed_by   : abp_locinst_lastchange_user;
17  last_changed_at   : abp_locinst_lastchange_tstmp;
18  local_last_changed_at : abp_lastchange_tstmp;
19
20 }

```

- Für **Vorstellung** (ZSS25\_R4C\_VORST) stellt sich der Inhalt der Tabelle wie folgt dar:

```

[ A07] ZSS25_R4C_VORST X
1 @EndUserText.label : 'Tabelle für Vorstellung'
2 @AbapCatalog.enhancement.category : #NOT_EXTENSIBLE
3 @AbapCatalog.tableCategory : #TRANSPARENT
4 @AbapCatalog.deliveryClass : #A
5 @AbapCatalog.dataMaintenance : #ALLOWED
6 define table zss25_r4c_vorst {
7   key mandt          : abap.clnt not null;
8   key vorstellung_id : abap.numc(10) not null;
9   film_id           : abap.numc(10);
10  raum_id           : abap.numc(10);
11  sprache           : abap.char(50);
12  datum             : abap.dats;
13  startzeit         : abap.tims;
14  endzeit           : abap.tims;
15  untertitel        : abap.char(10);
16  vorstellung_format : abap.char(20);
17  status             : abap.char(20);
18  created_by         : abp_creation_user;
19  created_at         : abp_creation_tstmp;
20  last_changed_by   : abp_locinst_lastchange_user;
21  last_changed_at   : abp_locinst_lastchange_tstmp;
22  local_last_changed_at : abp_lastchange_tstmp;
23
24 }

```

### Notiz:

- Nachdem die Tabellenstruktur erstellt wurde, muss sie mit Strg + S gespeichert und anschließend über das Aktivierungssymbol in SAP aktiviert werden.

- Nach erfolgreicher Aktivierung sind die Database Tables im Projektbaum unter dem Pfad Dictionary → Database Tables zu finden.

### 3.3. Dateneingabe in SAP Logon mit SE16N

Nachdem die Database Tables angelegt und aktiviert wurden, sind sie zunächst leer und müssen über SAP Logon mit Datensätzen befüllt werden. Dazu wird die Transaktion SE16N verwendet, die in SAP als Allgemeiner Tabellen-Display bezeichnet wird. Sie dient primär der Anzeige von Tabelleninhalten, kann jedoch unter bestimmten Voraussetzungen auch zur Pflege von Daten genutzt werden.

#### **Schritt-für-Schritt-Anleitung:**

SAP Logon öffnen → SE16N eingeben → Tabellennamen eingeben (z. B. ZSS25\_R4C\_MITARB) → Enter → F8 drücken → +-Symbol auswählen → Daten eingeben → Sichern

SAP

SE16N

Mehr ▾

▼ Favoriten

- ★ Data Browser
- ★ ABAP Dictionary Pflege
- ★ Object Navigator
- ★ ABAP Class Builder
- ★ SAP Fiori Launchpad

▼ SAP Menü

- > Konnektor für SAP Multi-Bank Connectivity
- > Global Bike Transaktionen
- > Büro
- > Anwendungsübergreifende Komponenten
- > Logistik
- > Rechnungswesen
- > Personal
- > Infosysteme
- > Service
- > Werkzeuge
- > WebClient-UI-Framework

Hintergrund Anzahl Einträge

Tabelle:

Texttabelle:

Anzeigevariante:

Maximale Trefferzahl:

Spalte einfügen:  Zeile einfügen Detail

MITARBEITER_ID	NAME	GEBURTS... DATUM	ROLLE	ABTEILU...	EINSTELLUNGSDATUM	GEHALT	KONTAKTINFO
5001	Kamal Badawi	01/15/1993	Kassierer	VKF1	10/01/2023	2,575.00	Seifgrundstr. 4, 61348 Bad Homburg
5002	Clara Schäfer	09/05/1998	Schichtleiter/in	VKF1	01/03/2016	3,375.00	Louisenstraße 32, 61348 Bad Homburg
5003	Sophie Braun	09/05/1995	Kassierer/in	VKF1	01/18/2016	2,800.00	Kaiser-Friedrich-Promenade 85, 61348 Bad Homburg
5004	Lukas Schneider	05/06/2000	Kassierer/in	VKF2	01/01/2019	1,950.00	Saalburgstraße 12, 61350 Bad Homburg
5005	Maximilian Hoffmann	05/06/2001	Schichtleiter/in	VKF2	01/01/2019	3,150.00	Friedrichstraße 7, 61348 Bad Homburg
5006	Anna Müller	05/06/2005	Kassierer/in	VKF2	01/01/2020	2,500.00	Gartenfeldstraße 22, 61348 Bad Homburg
5007	Lisa Müller	10/30/1997	Kassierer/in	VKF2	01/01/2022	1,980.00	Seifgrundstr. 1, 61348 Bad Homburg

MITARBEITER_ID	NAME	GEBURTS... DATUM	ROLLE	ABTEILU...	EINSTELLUNGSDATUM	GEHALT	KONTAKTINFO
5008	LARISSA DORF	03/04/2022	KASSIERER/IN	VKF1	05/19/2022	50.00	LOUISENSTRÄE 12, 61348 BAD HOMBURG
5001	Kamal Badawi	01/15/1993	KASSIERER/IN	VKF1	10/01/2023	2,575.00	Seifgrundstr. 4, 61348 Bad Homburg
5002	Clara Schäfer	09/05/1998	Schichtleiter/in	VKF1	01/03/2016	3,375.00	Louisenstraße 32, 61348 Bad Homburg
5003	Sophie Braun	09/05/1995	Kassierer/in	VKF1	01/18/2016	2,800.00	Kaiser-Friedrich-Promenade 85, 61348 Bad Homburg
5004	Lukas Schneider	05/06/2000	Kassierer/in	VKF2	01/01/2019	1,950.00	Saalburgstraße 12, 61350 Bad Homburg
5005	Maximilian Hoffmann	05/06/2001	Schichtleiter/in	VKF2	01/01/2019	3,150.00	Friedrichstraße 7, 61348 Bad Homburg
5006	Anna Müller	05/06/2005	Kassierer/in	VKF2	01/01/2020	2,500.00	Gartenfeldstraße 22, 61348 Bad Homburg
5007	Lisa Müller	10/30/1997	Kassierer/in	VKF2	01/01/2022	1,980.00	Seifgrundstr. 1, 61348 Bad Homburg

- Für **Mitarbeiter** (ZSS25\_R4C\_MITARB) stellt sich der Inhalt der Tabelle wie folgt dar:

MITARBEITER_ID	NAME	GEBURTS... DATUM	ROLLE	ABTEILUNG	EINSTELLUNGSDATUM	GEHALT	KONTAKTINFO
0000005006	Anna Müller	2005-05-06	Kassierer/in	VKF2	2020-01-01	2500.00	Gartenfeldstraße 22, ...
0000005002	Clara Schäfer	1998-09-05	Schichtleiter...	VKF1	2016-01-03	3375.00	Louisenstraße 32, 613...
0000005005	Maximilian ...	2001-05-06	Schichtleiter...	VKF2	2019-01-01	3150.00	Friedrichstraße 7, 613...
0000005003	Sophie Braun	1995-09-05	Kassierer/in	VKF1	2016-01-18	2800.00	Kaiser-Friedrich-Pro...
0000005007	Lisa Müller	1997-10-30	Kassierer/in	VKF2	2022-01-01	1980.00	Seifgrundstr. 1, 61348...
0000005004	Lukas Schne...	2000-05-06	Kassierer/in	VKF2	2019-01-01	1950.00	Saalburgstraße 12, 61...
0000005001	Kamal Badawi	1993-01-15	Kassierer	VKF1	2023-10-01	2575.00	Seifgrundstr. 4, 61348...

- Für **Besucher** (ZSS25\_R4C\_BESUCH) stellt sich der Inhalt der Tabelle wie folgt dar:

BESUCHER_ID	NAME	GEBURTS... DATUM	EMAIL	TELEFON	ADRESSE	REGISTRIERUNG
0000001001	ANNA MÜL...	1996-03-14	ANNA.MUE...	+49 151 834561	HAUPTSTR. 12...	2023-09-01
0000001002	TOBIAS SCH...	2010-07-22	T.SCHMIDT...	+49 176 998877	LINDENWEG 5,...	2012-07-25
0000001003	LINH NGUY...	2006-11-03	LINH.NGUY...	+49 172 556672	BERLINER ALL...	2019-02-07
0000001004	JANA BECKER	2014-01-28	JANA.BECKE...	+49 163 628894	GOETHESTR. 2...	2023-01-15
0000001005	OMAR ALI	1997-11-17	OMAR.ALI@...	+49 152 543255	RHEinstrasse ...	2023-09-01
0000001006	SOPHIE WA...	2002-12-17	SOPHIE.WA...	+49 151 712138	SCHILLERPLAT...	2023-10-02
0000001007	MARCO RO	1998-05-09	MARCO.RO	+49 162 334445	MOZARTSTR...	2012-09-10

- Für **Film** (ZSS25\_R4C\_FILM) stellt sich der Inhalt der Tabelle wie folgt dar:

FILM_ID	TITEL	GENRE	DAUER_MINUTEN	ALTERSFREIGABE	PRODUKTIONSJAHR	REGISSEUR
0000003006	AVATAR	SCI-FI	162	12	2009	JAMES CAMERON
0000003001	INCEPTION	SCI-FI	180	12	2016	RAFAEL MARTIN...
0000003003	THE DARK ...	ACTION	152	18	2008	LORENZO MILLA
0000003004	PARASITE	THRILLER	150	16	2025	BONG JOON-HO
0000003002	DIE EISKÖN...	ANIMATION	95	18	2024	JENNIFER LEE
0000003005	LA LA LAND	MUSICAL	128	0	2025	DAMIEN CHAZ...
0000003007	COCO	ANIMATION	105	18	2017	LEE UNKRICH
0000003008	INTERSTAL...	SCI-FI	65	6	2014	ULRICH KOCH

- Für **Kino** (ZSS25\_R4C\_KINO) stellt sich der Inhalt der Tabelle wie folgt dar:

RB	KINO_ID	RB	NAME	RB	ADRESSE	RB	STADT	RB	PLZ	12	ANZAHL_RAEUME	RB	EROEFFNUNGSDATUM	RB	KONTAKTINFO
0000004002	UCI DX	AM WEHRHA...	FRIEDBERG	40211						4	2003-03-30	01710 8846622			
0000004003	MATHÄSER ...	BAYERSTRÄÙE 3	MARBURG	80335						6	2004-12-08	01725 8636633			
0000004004	CINEMINNA...	DAMMTORST...	LIMBURG	20354						5	2006-09-15	01600 4446644			
0000004005	FILM ONE	HOHENZOLLE...	FRANKFURT	50672						3	2010-05-24	01500 2223355			
0000004001	RAP4YOU	POTSDAMER P...	BAD HOMB...	10785						5	2001-12-01	01514 1223334			

- Für **Raum** (ZSS25\_R4C\_RAUM) stellt sich der Inhalt der Tabelle wie folgt dar:

RB	RAUM_ID	RB	KINO_ID	RB	NAME	12	STUEHL_ANZAHL_PRO_REIHE	12	SITZPLAETZE	RB	AUSSTATTUNG
0000006001	0000004001	SAAL 1					10		120	3D, DOLBY SURROUND	
0000006002	0000004001	SAAL 2					8		80	2D, DOLBY SURROUND	
0000006003	0000004002	HAUPTSAAL					13		150	3D, DOLBY ATMOS	
0000006005	0000004003	SAAL B					6		60	2D, STANDARD	
0000006006	0000004004	SAAL 1					12		120	3D, DOBY SURROUND	
0000006007	0000004004	SAAL 2					9		90	2D, STANDARD	
0000006008	0000004005	GROÙER SAAL					15		200	3D, DOLBY ATMOS, I...	
0000006009	0000004005	KLEINER SAAL					7		70	2D, STANDARD	
0000006004	0000004003	SAAL A					10		100	2D, STANDARD	

- Für **Ticket** (ZSS25\_R4C\_TICKET) stellt sich der Inhalt der Tabelle wie folgt dar:

RB	TICKET_ID	RB	VORSTELLUNG_ID	RB	BESUCHER_ID	12	SITZPLATZNUMMER	12	PREIS	RB	TICKET_TYP	RB	QR_CODE	RB	STATUS
0000008012	0000007003	0000001004				40		8.00	KIND	HTTPS://GITHU...	BEZAHLT				
0000008017	0000007003	0000001001				4		40.00	VIP	HTTPS://GITHU...	BEZAHLT				
0000008005	0000007004	0000001005				2		35.00	VIP	HTTPS://GITHU...	BEZAHLT				
0000008015	0000007005	0000001006				2		25.00	VIP	HTTPS://GITHU...	BEZAHLT				
0000008001	0000007001	0000001001				13		15.00	KIND	HTTPS://GITHU...	BEZAHLT				
0000008013	0000007001	0000001001				20		20.00	STANDARD	HTTPS://GITHU...	BEZAHLT				
0000008011	0000007002	0000001003				22		20.00	STANDARD	HTTPS://GITHU...	BEZAHLT				
0000008008	0000007003	0000001001				1		45.00	VIP	HTTPS://GITHU...	BEZAHLT				
0000008007	0000007006	0000001007				5		15.00	KIND	HTTPS://GITHU...	BEZAHLT				
0000008016	0000007006	0000001007				3		35.00	VIP	HTTPS://GITHU...	BEZAHLT				
0000008022	0000007007	0000001006				2		25.00	VIP	HTTPS://GITHU...	BEZAHLT				
0000008021	0000007008	0000001007				5		35.00	VIP	HTTPS://GITHU...	BEZAHLT				
0000008006	0000007005	0000001006				1		35.00	VIP	HTTPS://GITHU...	BEZAHLT				
0000008018	0000007003	0000001001				5		40.00	VIP	HTTPS://GITHU...	BEZAHLT				
0000008009	0000007003	0000001001				3		40.00	VIP	HTTPS://GITHU...	STORNERT				
0000008014	0000007004	0000001005				1		45.00	VIP	HTTPS://GITHU...	STORNERT				
0000008020	0000007008	0000001006				1		25.00	VIP	HTTPS://GITHU...	STORNERT				
0000008023	0000007007	0000001007				3		25.00	VIP	HTTPS://GITHU...	STORNERT				
0000008002	0000007001	0000001002				32		20.00	STANDARD	HTTPS://GITHU...	BEZAHLT				
0000008003	0000007002	0000001003				34		18.00	STANDARD	HTTPS://GITHU...	BEZAHLT				
0000008004	0000007003	0000001004				2		22.00	STANDARD	HTTPS://GITHU...	BEZAHLT				
0000008010	0000007001	0000001002				35		15.00	STANDARD	HTTPS://GITHU...	BEZAHLT				
0000008019	0000007002	0000001002				5		20.00	STANDARD	HTTPS://GITHU...	BEZAHLT				

- Für **Verkauf** (ZSS25\_R4C\_VERKF) stellt sich der Inhalt der Tabelle wie folgt dar:

VERKAUF_ID	TICKET_ID	MITARBEITER_ID	VERKAUFSDATUM	ZAHLUNGSArt	RABATT
0000009001	0000008001	0000005003	2025-10-11	BAR	0.00
0000009002	0000008002	0000005005	2025-10-15	BAR	25.00
0000009010	0000008010	0000005007	2025-10-02	KARTE	0.00
0000009013	0000008013	0000005001	2025-10-01	KARTE	15.00
0000009003	0000008003	0000005001	2025-10-12	KARTE	10.00
0000009011	0000008011	0000005001	2025-10-22	BAR	0.00
0000009019	0000008019	0000005006	2025-10-23	KARTE	20.00
0000009008	0000008008	0000005002	2025-10-15	BAR	10.00
0000009012	0000008012	0000005004	2025-10-30	BAR	0.00
0000009004	0000008004	0000005002	2025-10-01	KARTE	0.00
0000009017	0000008017	0000005004	2025-10-25	BAR	5.00
0000009018	0000008018	0000005005	2025-09-27	BAR	0.00
0000009009	0000008009	0000005006	2025-10-30	KARTE	5.00
0000009005	0000008005	0000005001	2025-10-14	BAR	0.00
0000009014	0000008014	0000005003	2025-10-13	KARTE	15.00
0000009006	0000008006	0000005003	2025-10-12	BAR	5.00
0000009015	0000008015	0000005001	2025-10-17	BAR	0.00
0000009022	0000008022	0000005007	2025-10-23	BAR	0.00
0000009023	0000008023	0000005006	2025-10-25	KARTE	0.00
0000009020	0000008020	0000005001	2025-10-15	BAR	0.00
0000009021	0000008021	0000005005	2025-10-12	BAR	0.00
0000009007	0000008007	0000005004	2025-10-27	BAR	0.00
0000009016	0000008016	0000005007	2025-10-28	BAR	0.00

- Für **Vorstellung** (ZSS25\_R4C\_VORST) stellt sich der Inhalt der Tabelle wie folgt dar:

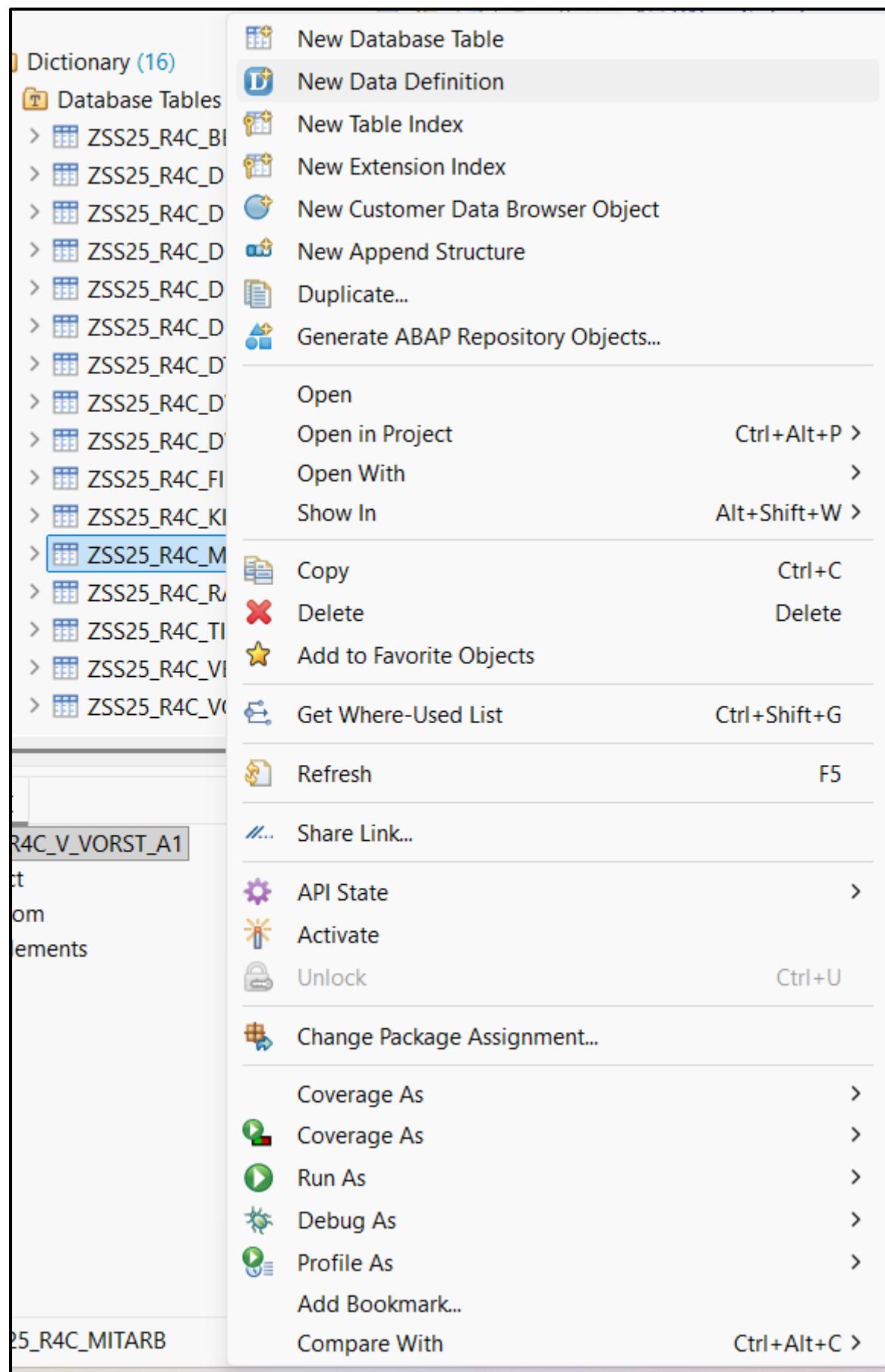
VORSTELLUNG_ID	FILM_ID	RAUM_ID	SPRACHE	DATUM	STARTZEIT	ENDZEIT	UNTERTITEL	VORSTELLUNG_FORMAT	STATUS
0000007007	0000003006	0000006006	EN	2025-11-07	05:00:00 PM	09:30:00 PM	VORHANDEN	2D	AKTIV
0000007008	0000003007	0000006008	DE	2025-10-25	04:00:00 PM	05:30:00 PM	VORHANDEN	3D	AKTIV
0000007001	0000003001	0000006001	DE	2025-11-09	02:00:00 PM	06:00:00 PM	VORHANDEN	3D	ABGESAGT
0000007002	0000003008	0000006001	DE	2025-10-23	05:00:00 PM	06:30:00 PM	VORHANDEN	IMAX	AKTIV
0000007003	0000003002	0000006002	DE	2025-11-02	03:00:00 PM	05:00:38 PM	N. V.	3D	AKTIV
0000007004	0000003003	0000006003	DE	2025-10-15	06:00:00 PM	09:00:00 PM	N. V.	IMAX	AKTIV
0000007005	0000003004	0000006004	EN	2025-10-12	06:00:00 PM	09:00:00 PM	VORHANDEN	IMAX	AKTIV
0000007006	0000003005	0000006005	DE	2025-10-18	06:00:00 PM	08:30:00 PM	VORHANDEN	2D	AKTIV

### 3.4. Basis View

Die Basis-View stellt die erste Abstraktionsschicht über der zugrunde liegenden Datenbanktabelle dar. Sie dient dazu, die in der physischen Tabelle gespeicherten Daten strukturiert und lesbar bereitzustellen. In dieser View werden die relevanten Felder der Tabelle ausgewählt, gegebenenfalls umbenannt und technisch vorbereitet, um sie in nachfolgenden Schichten weiterverwenden zu können. Die Basis-View bildet somit die Grundlage für die Entwicklung komplexerer CDS-Views wie Interface Views (Reusable Views) oder Projection Views. Sie enthält keine Geschäftslogik oder UI-spezifischen Elementen, sondern konzentriert sich ausschließlich auf die fachlich korrekte und performante Aufbereitung der Rohdaten aus der Datenbank. Durch diese klare Trennung wird eine saubere, wartbare Architektur geschaffen, in der jede Schicht eine definierte Aufgabe innerhalb des RAP-Modells erfüllt.

### **Schritt-für-Schritt-Anleitung:**

Rechtsklick auf eine Database Table → New Data Definition → Referenced Object, Name und Beschreibung eingeben, z. B.: Referenced Object: ZSS25\_R4C/MITARB || Name: ZSS25\_R4C\_V/MITARB\_A1 || Beschreibung: View für Mitarbeiter → Next → Next → defineViewEntity auswählen → Finish



 New Data Definition

**Data Definition**

Create a data definition



Project: \* A07\_200\_learn-195\_en

Package: \* \$SS25\_RAP4CINEMA

Add to favorite packages

Name: \* ZSS25\_R4C\_V\_MITARB\_A1

Description: \* View für Mitarbeiter

Original Language: EN

Referenced Object: ZSS25\_R4C\_MITARB

 < Back Next > Finish Cancel

 New Data Definition

### Templates

 Selected template uses features which are not yet supported by the backend system.

Use the selected template

Name	Description
View (creation)	
 defineViewEntity	Define View Entity
 defineRootViewEntity	Define Root View Entity
 defineViewEntityWithToParentAssociation	Define View Entity with To-Parent Association
 defineView	Define View (obsolete as of AS ABAP 7.57)

```

@AbapCatalog.viewEnhancementCategory: [#NONE]
@AccessControl.authorizationCheck: #NOT_REQUIRED
@EndUserText.label: '${ddl_source_description}'
@Metadata.ignorePropagatedAnnotations: true
@ObjectModel.usageType:{
    serviceQuality: #X,
    sizeCategory: #S,
    dataClass: #MIXED
}
define view entity ${ddl_source_name} as select from ${data_source_name}
{
    ${data_source_elements}${cursor}
}

```

- Für **Mitarbeiter** (ZSS25\_R4C\_V\_MITARB\_A1) stellt sich der Inhalt der Basis View wie folgt dar:

```

[D] [A07] ZSS25_R4C_V_MITARB_A1 X
ntabelle) AbapCatalog.viewEnhancementCategory: [#NONE]
  2 @AccessControl.authorizationCheck: #NOT_REQUIRED
  3 @EndUserText.label: 'View für Mitarbeiter'
  4 @Metadata.ignorePropagatedAnnotations: true
  5 @ObjectModel.usageType: {
  6   serviceQuality: #X,
  7   sizeCategory: #S,
  8   dataClass: #MIXED
  9 }
10 define view entity ZSS25_R4C_V_MITARB_A1 as select from zss25_r4c_mitarb
11 {
12   key mitarbeiter_id as MitarbeiterId,
13   name as Name,
14   geburtsdatum as Geburtsdatum,
15   rolle as Rolle,
16   abteilung as Abteilung,
17   einstellungsdatum as Einstellungsdatum,
18   gehalt as Gehalt,
19   kontaktinfo as Kontaktinfo,
20   @Semantics.user.createdBy: true
21   created_by as CreatedBy,
22   @Semantics.systemDateTime.createdAt: true
23   created_at as CreatedAt,
24   @Semantics.user.lastChangedBy: true
25   last_changed_by as LastChangedBy,
26   @Semantics.systemDateTime.lastChangedAt: true
27   last_changed_at as LastChangedAt,
28   @Semantics.systemDateTime.localInstanceLastChangedAt: true
29   local_last_changed_at as LocalLastChangedAt
30 }
31

```

- Für **Besucher** (ZSS25\_R4C\_V\_BESUCH\_A1) stellt sich der Inhalt der Basis View wie folgt dar:

```

1 @AbapCatalog.viewEnhancementCategory: [#NONE]
2 @AccessControl.authorizationCheck: #NOT_REQUIRED
3 @EndUserText.label: 'View für Besucher'
4 @Metadata.ignorePropagatedAnnotations: true
5 @ObjectModel.usageType:{
6     serviceQuality: #X,
7     sizeCategory: #S,
8     dataClass: #MIXED
9 }
10 define root view entity ZSS25_R4C_V_BESUCH_A1 as select from zss25_r4c_besuch
11
12 {
13     key besucher_id as BesucherId,
14     name as Name,
15     geburtsdatum as Geburtsdatum,
16     email as Email,
17     telefon as Telefon,
18     adresse as Adresse,
19     registrierung as Registrierung,
20     created_by as CreatedBy,
21     created_at as CreatedAt,
22     last_changed_by as LastChangedBy,
23     last_changed_at as LastChangedAt,
24     local_last_changed_at as LocalLastChangedAt,
25
26
27

```

Das Alter der Besucher (BesucherAlter) wurde auf Basis des Geburtsdatums (geburtsdatum) wie folgt berechnet:

```

25
26
27
28 // ****
29 // ****
30 // BERECHNUNGEN
31 // ****
32 // ****
33
34 // BesucherAlter
35 @EndUserText.label: 'BesucherAlter'
36 cast(
37     ( cast( substring( $session.system_date, 1, 4 ) as abap.int2 )
38     - cast( substring( geburtsdatum, 1, 4 ) as abap.int2 ) )
39     - case
40         when cast( substring( $session.system_date, 5, 4 ) as abap.int2 )
41             < cast( substring( geburtsdatum, 5, 4 ) as abap.int2 )
42             then 1
43             else 0
44         end
45     as abap.int2
46 ) as BesucherAlter
47
48
49
50

```

- Für **Film** (ZSS25\_R4C\_V\_FILM\_A1) stellt sich der Inhalt der Basis View wie folgt dar:

```
D [A07] ZSS25_R4C_V_FILM_A1 X
1@AbapCatalog.viewEnhancementCategory: [#NONE]
2@AccessControl.authorizationCheck: #NOT_REQUIRED
3@EndUserText.label: 'View für Film'
4@Metadata.ignorePropagatedAnnotations: true
5@ObjectModel.usageType:{
6    serviceQuality: #X,
7    sizeCategory: #S,
8    dataClass: #MIXED
9}
10define view entity ZSS25_R4C_V_FILM_A1 as select from zss25_r4c_film
11{
12    key film_id as FilmId,
13    titel as Titel,
14    genre as Genre,
15    dauer_minuten as DauerMinuten,
16    altersfreigabe as Altersfreigabe,
17    produktionsjahr as Produktionsjahr,
18    regisseur as Regisseur,
19    @Semantics.user.createdBy: true
20    created_by as CreatedBy,
21    @Semantics.systemDateTime.createdAt: true
22    created_at as CreatedAt,
23    @Semantics.user.lastChangedBy: true
24    last_changed_by as LastChangedBy,
25    @Semantics.systemDateTime.lastChangedAt: true
26    last_changed_at as LastChangedAt,
27    @Semantics.systemDateTime.localInstanceLastChangedAt: true
28    local_last_changed_at as LocalLastChangedAt
29
30}
31
```

- Für **Kino** (ZSS25\_R4C\_V\_KINO\_A1) stellt sich der Inhalt der Basis View wie folgt dar:

```

D [A07] ZSS25_R4C_V_KINO_A1 X
1 @AbapCatalog.viewEnhancementCategory: [#NONE]
2 @AccessControl.authorizationCheck: #NOT_REQUIRED
3 @EndUserText.label: 'View für Kino'
4 @Metadata.ignorePropagatedAnnotations: true
5 @ObjectModel.usageType:{
6   serviceQuality: #X,
7   sizeCategory: #S,
8   dataClass: #MIXED
9 }
10 define view entity ZSS25_R4C_V_KINO_A1 as select from zss25_r4c_kino
11 {
12   key kino_id as KinoId,
13   name as Name,
14   adresse as Adresse,
15   stadt as Stadt,
16   plz as Plz,
17   anzahl_raeume as AnzahlRaeume,
18   eroeffnungsdatum as Eroeffnungsdatum,
19   kontaktinfo as Kontaktinfo,
20   @Semantics.user.createdBy: true
21   created_by as CreatedBy,
22   @Semantics.systemDateTime.createdAt: true
23   created_at as CreatedAt,
24   @Semantics.user.lastChangedBy: true
25   last_changed_by as LastChangedBy,
26   @Semantics.systemDateTime.lastChangedAt: true
27   last_changed_at as LastChangedAt,
28   @Semantics.systemDateTime.localInstanceLastChangedAt: true
29   local_last_changed_at as LocalLastChangedAt
30 }
31

```

- Für **Raum** (ZSS25\_R4C\_V\_RAUM\_A1) stellt sich der Inhalt der Basis View wie folgt dar:

```

1① @AbapCatalog.viewEnhancementCategory: [#NONE]
2  @AccessControl.authorizationCheck: #NOT_REQUIRED
3  @EndUserText.label: 'View für Raum'
4  @Metadata.ignorePropagatedAnnotations: true
5  @ObjectModel.usageType:{
6    serviceQuality: #X,
7    sizeCategory: #S,
8    dataClass: #MIXED
9  }
10 define view entity ZSS25_R4C_V_RAUM_A1 as select from zss25_r4c_raum
11 {
12   key raum_id as RaumId,
13   kino_id as KinoId,
14   name as Name,
15   stuehl_anzahl_pro_reihe as StuehlAnzahlProReihe,
16   sitzplaetze as Sitzplaetze,
17   ausstattung as Ausstattung,
18   @Semantics.user.createdBy: true
19   created_by as CreatedBy,
20   @Semantics.systemDateTime.createdAt: true
21   created_at as CreatedAt,
22   @Semantics.user.lastChangedBy: true
23   last_changed_by as LastChangedBy,
24   @Semantics.systemDateTime.lastChangedAt: true
25   last_changed_at as LastChangedAt,
26   @Semantics.systemDateTime.localInstanceLastChangedAt: true
27   local_last_changed_at as LocalLastChangedAt
28
29 }
```

- Für **Ticket** (ZSS25\_R4C\_V\_TICKET\_A1) stellt sich der Inhalt der Basis View wie folgt dar:

```

1@AbapCatalog.viewEnhancementCategory: [#NONE]
2@AccessControl.authorizationCheck: #NOT_REQUIRED
3@EndUserText.label: 'View für Ticket'
4@Metadata.ignorePropagatedAnnotations: true
5@ObjectModel.usageType:{
6    serviceQuality: #X,
7    sizeCategory: #S,
8    dataClass: #MIXED
9}
10define view entity ZSS25_R4C_V_TICKET_A1 as select from zss25_r4c_ticket as tkt
11inner join zss25_r4c_vorst as vor on vor.vorstellung_id = tkt.vorstellung_id
12inner join zss25_r4c_raum as rm on rm.raum_id = vor.raum_id
13
14
15
16{
17    key tkt.ticket_id as TicketId,
18    tkt.vorstellung_id as VorstellungId,
19    tkt.besucher_id as BesucherId,
20    rm.raum_id as RaumId,
21    rm.kino_id as KinoId,
22    tkt.sitzplatznummer as Sitzplatznummer,
23    tkt.preis as Preis,
24    tkt.ticket_typ as TicketTyp,
25    tkt.qr_code as QrCode,
26    tkt.status as Status,
27    tkt.created_by as CreatedBy,
28    tkt.created_at as CreatedAt,
29    tkt.last_changed_by as LastChangedBy,
30    tkt.last_changed_at as LastChangedAt,
31    tkt.local_last_changed_at as LocalLastChangedAt
32
33
34}
35

```

- Für **Verkauf** (ZSS25\_R4C\_V\_VERKF\_A1) stellt sich der Inhalt der Basis View wie folgt dar:

D [A07] ZSS25\_R4C\_V\_VERKF\_A1 X

```
4 @EndUserText.label: 'View für Verkauf (Faktentabelle)'  
5 @Metadata.ignorePropagatedAnnotations: true  
6 @ObjectModel.usageType:{  
7   serviceQuality: #X,  
8   sizeCategory: #S,  
9   dataClass: #MIXED  
10 }  
11 define view entity ZSS25_R4C_V_VERKF_A1 as select from zss25_r4c_verkf as vkf  
12 left outer join ZSS25_R4C_V_MITARB_A1 as ma on ma.MitarbeiterId = vkf.mitarbeiter_id  
13 left outer join ZSS25_R4C_V_TICKET_A1 as tkt on tkt.TicketId = vkf.ticket_id  
14 left outer join ZSS25_R4C_V_BESUCH_A1 as bes on bes.BesucherId = tkt.BesucherId  
15 left outer join ZSS25_R4C_V_VORST_A1 as vorst on vorst.VorstellungId = tkt.VorstellungId  
16 left outer join ZSS25_R4C_V_FILM_A1 as film on film.FilmId = vorst.FilmId  
17 left outer join ZSS25_R4C_V_RAUM_A1 as rm on rm.RaumId = vorst.RaumId  
18 left outer join ZSS25_R4C_V_KINO_A1 as kn on kn.KinoId = rm.KinoId  
19 {  
20   key vkf.verkauf_id as VerkaufId,  
21   @EndUserText.label: 'Ticket-Nr.'  
22   vkf.ticket_id as TicketId,  
23   vkf.mitarbeiter_id as MitarbeiterId,  
24   vkf.verkaufsdatum as Verkaufsdatum,  
25   vkf.zahlungsart as Zahlungsart,  
26   vkf.rabatt as Rabatt,  
27   vkf.created_by as CreatedBy,  
28   vkf.created_at as CreatedAt,  
29   vkf.last_changed_by as LastChangedBy,  
30   vkf.last_changed_at as LastChangedAt,  
31   vkf.local_last_changed_at as LocalLastChangedAt,  
32  
33  
34
```

```

35@    // ****
36    // MITARBEITER
37    // ****
38    ma.Name as MitarbeiterName,
39    ma.Geburtsdatum as MitarbeiterGeburtsdatum,
40    ma.Rolle as MitarbeiterRolle,
41    ma.Abteilung as MitarbeiterAbteilung,
42    ma.Einstellungsdatum as MitarbeiterEinstellungsdatum,
43    ma.Gehalt as MitarbeiterGehalt,
44    ma.Kontaktinfo as MitarbeiterKontaktinfo,
45
46
47@    // ****
48    // TICKET
49    // ****
50    tkt.Sitzplatznummer as TicketSitzplatznummer,
51    tkt.Preis as TicketPreis,
52    tkt.TicketTyp as TicketTicketTyp,
53    tkt.QrCode as TicketQrCode,
54    tkt.Status as TicketStatus,
55
56
57
58@    // ****
59    // BESUCHER
60    // ****
61    bes.Name as BesucherName,
62    bes.Geburtsdatum as BesucherGeburtsdatum,
63    bes.BesucherAlter as BesucherAlter,
64    bes.Email as BesucherEmail,
65    bes.Telefon as BesucherTelefon,
66    bes.Adresse as BesucherAdresse,
67    bes.Registrierung as BesucherRegistrierung,
68
69

```

```

70@ // ****
71 // VORSTELLUNG
72 // ****
73 vorst.Sprache as VorstellungSprache,
74 vorst.Datum as Datum,
75 vorst.Startzeit as VorstellungStartzeit,
76 vorst.Endzeit as VorstellungEndzeit,
77 vorst.Untertitel as VorstellungUntertitel,
78 vorst.VorstellungFormat as VorstellungVorstellungFormat,
79 vorst.Status as VorstellungStatus,
80
81
82@ // ****
83 // FILM
84 // ****
85 flm.Titel as FilmTitel,
86 flm.Genre as FilmGenre,
87 flm.DauerMinuten as FilmDauerMinuten,
88 flm.Altersfreigabe as FilmAltersfreigabe,
89 flm.Produktionsjahr as FilmProduktionsjahr,
90 flm.Regisseur as FilmRegisseur,
91
92
93@ // ****
94 // RAUM
95 // ****
96 rm.Name as RaumName,
97 rm.Sitzplaetze as RaumSitzplaetze,
98 rm.Ausstattung as RaumAusstattung,
99
100

```

```

100
101
102@ // ****
103 // KINO
104 // ****
105 kn.Name as KinoName,
106 kn.Adresse as KinoAdresse,
107 kn.Stadt as KinoStadt,
108 kn.Plz as KinoPlz,
109 kn.AnzahlRaeume as KinoAnzahlRaeume,
110 kn.Eroeffnungsdatum as KinoEroeffnungsdatum,
111 kn.Kontaktinfo as KinoKontaktinfo,
112
113
114

```

Im folgenden Abschnitt werden verschiedene Berechnungen definiert, die zusätzliche Informationen zu Besuchern, Filmen und Räumen liefern. Zunächst wird aus dem Alter des Besuchers (BesucherAlter) die entsprechende Altersgruppe (BesucherAltersGruppe) abgeleitet: Besucher unter 15 Jahren werden als Kind, zwischen 15 und 64 Jahren als Erwachsenen und ab 65 Jahren als Senior eingestuft. Anschließend wird geprüft, ob ein Besucher den Film aufgrund der Altersfreigabe (Altersfreigabe) sehen darf (BesucherIstErlaubt). Ist das Alter des Besuchers geringer als die im Film hinterlegte Altersfreigabe, wird der Status Verboten gesetzt, andernfalls Erlaubt.

Darüber hinaus wird die Anzahl der Sitzreihen (RaumAnzahlReihen) im jeweiligen Raum berechnet, indem die Gesamtzahl der Sitzplätze (Sitzplaetze) durch die Anzahl der Stühle pro Reihe (StuehlAnzahlProReihe) geteilt und auf die nächste ganze Zahl

aufgerundet wird. Schließlich wird auf Basis der Sitzplatznummer (Sitzplatznummer) und der Anzahl der Stühle pro Reihe (StuehlAnzahlProReihe) die Sitzreihe des Besuchers (BesucherSitzreihe) ermittelt, ebenfalls durch Aufrunden auf die nächste ganze Zahl. Der zu zahlende Betrag (Betrag) wurde anhand des Ticketpreises (tkt.Preis) und des gewährten Rabatts (vkf.rabatt) berechnet.

```

115@    // BesucherAltersgruppe
116    @EndUserText.label: 'BesucherAltersgruppe'
117    case
118        when bes.BesucherAlter < 15 then 'Kinds'
119        when bes.BesucherAlter between 18 and 64 then 'Erwachsen'
120        else 'Senior'
121    end as BesucherAltersgruppe,
122
123
124@    // BesucherIstErlaubt
125    @EndUserText.label: 'BesucherIstErlaubt'
126    case
127        when bes.BesucherAlter < flm.Altersfreigabe then 'Verboten'
128
129        else 'Erlaubt'
130    end as BesucherIstErlaubt,
131

132
133@    // BesucherSitzReihe
134    @EndUserText.label: 'BesucherSitzReihe'
135    cast( ceil( tkt.Sitzplatznummer / rm.StuehlAnzahlProReihe ) as abap.int4 ) as BesucherSitzReihe,
136
137
138@    // Prüft, ob das Ticket gültig ist
139    @EndUserText.label: 'Ticket ist gültig'
140    case
141        when bes.BesucherAlter > 14 and tkt.TicketTyp = 'KIND' then 'Ungültig'
142        else 'Gültig'
143    end as TicketIstGueltig,
144
145
146
147    // Zuzahlender Betrag
148    cast( tkt.Preis * (1 - (vkf.rabatt / 100)) as abap.dec(15,2) ) as Betrag,
149
150
151
152@    // RaumAnzahlReihen
153    @EndUserText.label: 'RaumAnzahlReihen'
154    cast( ceil( rm.Sitzplaetze / rm.StuehlAnzahlProReihe ) as abap.int4 ) as RaumAnzahlReihen,
155
156
157

```

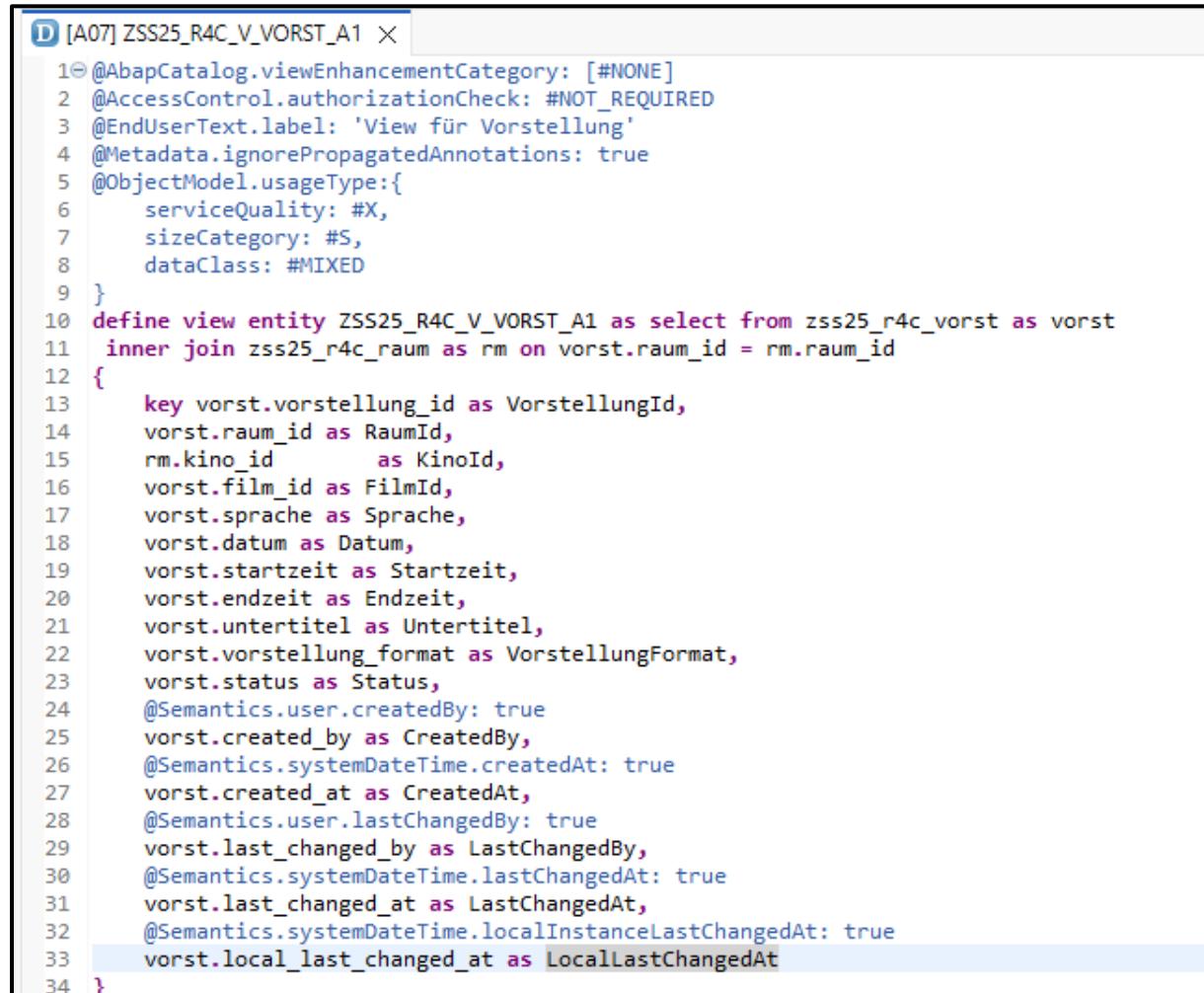
In der folgenden Abbildung wurde die zeitliche Einordnung (Zeitkategorie) der Vorstellungen (Vergangenheit, Heute, Zukunft) anhand der Spalte (vorst.datum) berechnet.

```

156
157
158     //Zeitkategorie
159     case
160         when vorst.Datum < $session.system_date
161             then 'Vergangenheit'
162         when vorst.Datum = $session.system_date
163             then 'Heute'
164         else 'Zukunft'
165     end as Zeitkategorie
166
167 }
```

Die Verkaufstabelle dient als Faktentabelle, in der alle anderen Tabellen miteinander verknüpft werden (Joins). Die Verknüpfung erfolgt über Primärschlüssel und Fremdschlüssel.

- Für **Vorstellung** (ZSS25\_R4C\_V\_VORST\_A1) stellt sich der Inhalt der Basis View wie folgt dar:



The screenshot shows the ABAP code for the view ZSS25\_R4C\_V\_VORST\_A1. The code defines the view as a select from the zss25\_r4c\_vorst table, joining it with the zss25\_r4c\_raum table. It lists various fields such as VorstellungId, RaumId, Kinoid, FilmId, Sprache, Datum, Startzeit, Endzeit, Untertitel, VorstellungFormat, Status, CreatedBy, CreatedAt, LastChangedBy, LastChangedAt, and LocalLastChangedAt. Annotations like @ObjectModel.usageType and @Semantics.user.\* are used throughout the code.

```

D [A07] ZSS25_R4C_V_VORST_A1 X
1 @AbapCatalog.viewEnhancementCategory: [#NONE]
2 @AccessControl.authorizationCheck: #NOT_REQUIRED
3 @EndUserText.label: 'View für Vorstellung'
4 @Metadata.ignorePropagatedAnnotations: true
5 @ObjectModel.usageType:{ ...
6     serviceQuality: #X,
7     sizeCategory: #S,
8     dataClass: #MIXED
9 }
10 define view entity ZSS25_R4C_V_VORST_A1 as select from zss25_r4c_vorst as vorst
11   inner join zss25_r4c_raum as rm on vorst.raum_id = rm.raum_id
12 {
13   key vorst.vorstellung_id as VorstellungId,
14   vorst.raum_id as RaumId,
15   rm.kino_id      as Kinoid,
16   vorst.film_id as FilmId,
17   vorst.sprache as Sprache,
18   vorst.datum as Datum,
19   vorst.startzeit as Startzeit,
20   vorst.endzeit as Endzeit,
21   vorst.untertitel as Untertitel,
22   vorst.vorstellung_format as VorstellungFormat,
23   vorst.status as Status,
24   @Semantics.user.createdBy: true
25   vorst.created_by as CreatedBy,
26   @Semantics.systemDateTime.createdAt: true
27   vorst.created_at as CreatedAt,
28   @Semantics.user.lastChangedBy: true
29   vorst.last_changed_by as LastChangedBy,
30   @Semantics.systemDateTime.lastChangedAt: true
31   vorst.last_changed_at as LastChangedAt,
32   @Semantics.systemDateTime.localInstanceLastChangedAt: true
33   vorst.local_last_changed_at as LocalLastChangedAt
34 }
```

**Notiz:**

- Nachdem die Basis View erstellt wurde, muss sie mit Strg + S gespeichert und anschließend über das Aktivierungssymbol in SAP aktiviert werden.
- Nach erfolgreicher Aktivierung ist die Basis View im Projektbaum unter dem Pfad Core Data Services → Data Definitions zu finden.

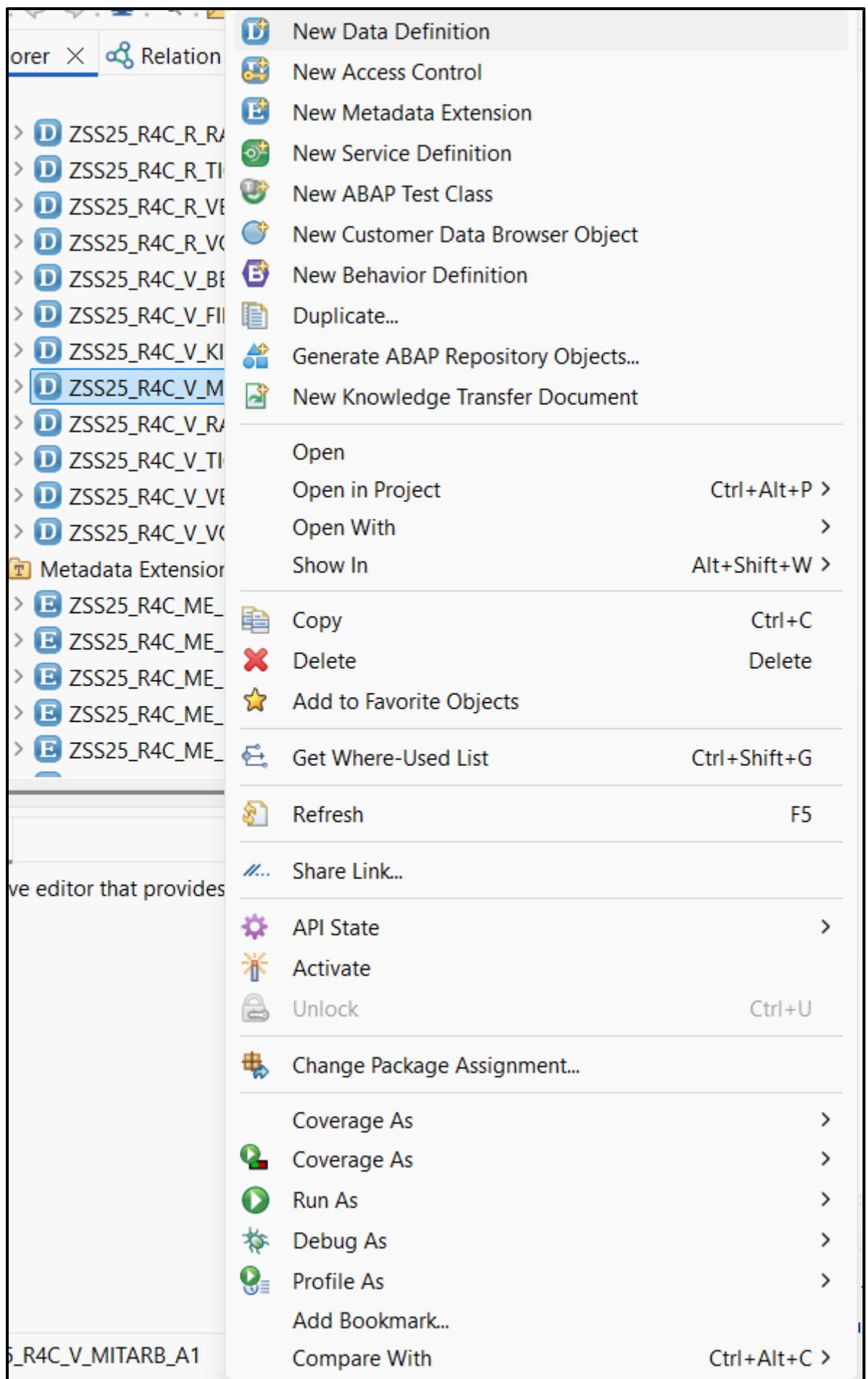
### 3.5. Interface View (Reusable View)

Die Interface View definiert die fachliche Struktur einer Entität, einschließlich ihrer relevanten Felder und Assoziationen. Sie bildet die zentrale Schnittstelle für weiterführende CDS-Views und Anwendungen und stellt damit eine klare Abstraktion der zugrunde liegenden Datenbanktabelle dar.

Als wiederverwendbare View ist sie frei von UI-spezifischen Elementen und dient ausschließlich der konsistenten Bereitstellung von Daten für nachgelagerte Schichten - etwa für Composition Views, Consumption Views oder Behavior Definitions. Dadurch wird eine saubere Trennung zwischen Datenmodell, Logik und Präsentationsebene sichergestellt.

#### **Schritt-für-Schritt-Anleitung:**

Rechtsklick auf eine Basis View → New Data Definition → Referenced Object, Name und Beschreibung eingeben, z. B.: Referenced Object: ZSS25\_R4C\_V\_MITARB\_A1 || Name: ZSS25\_R4C\_R\_MITARB\_A2 || Beschreibung: Reusable View für Mitarbeiter → Next → Next → defineRootViewEntity auswählen → Finish



New Data Definition

### Data Definition

Create a data definition

**D**

Project: \* A07\_200\_learn-195\_en

Package: \* \$SS25\_RAP4CINEMA

Add to favorite packages

Name: \* ZSS25\_R4C\_R\_MITARB\_A2

Description: \* Reusable View für Mitarbeiter

Original Language: EN

Referenced Object: ZSS25\_R4C\_V\_MITARB\_A1

 New Data Definition

**Select Transport Request**

 No change recording enabled for package \$SS25\_RAP4CINEMA



Choose from requests in which I am involved  Configure Columns

Transport Request	Owner	Target	Description	CTS Proj

Create a new request

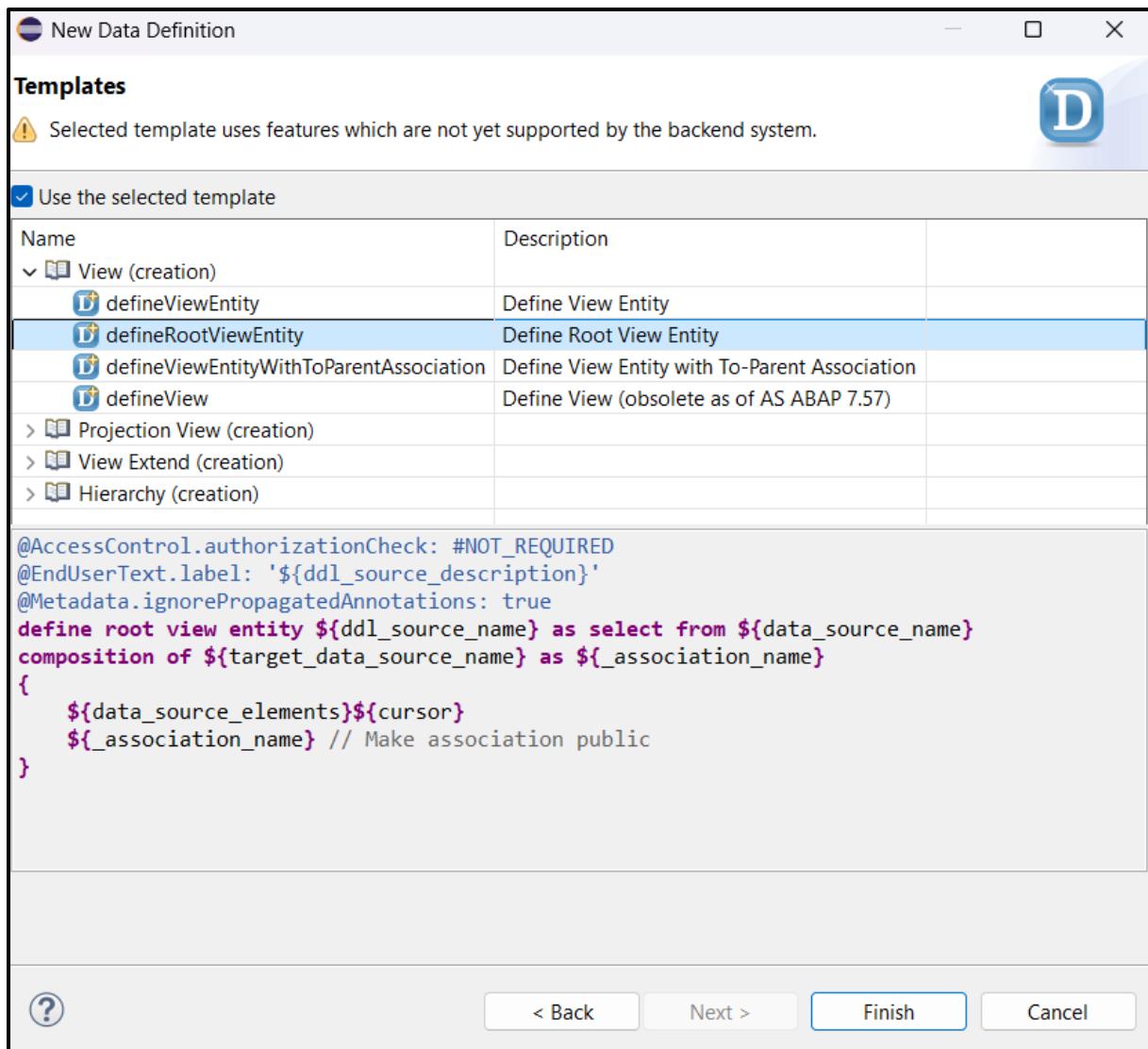
Request Description: \*

CTS Project:  Browse...

Enter a request number

Request Number:  Browse...





- Für **Mitarbeiter** (ZSS25\_R4C\_R\_MITARB\_A2) stellt sich der Inhalt der Reusable View wie folgt dar:

```

D [A07] ZSS25_R4C_R_MITARB_A2 ×
1 @AccessControl.authorizationCheck: #NOT_REQUIRED
2 @EndUserText.label: 'Reusable View für Mitarbeiter'
3 @Metadata.ignorePropagatedAnnotations: true
4 define root view entity ZSS25_R4C_R_MITARB_A2 as select from ZSS25_R4C_V_MITARB_A1
5 composition [1..*] of ZSS25_R4C_R_VERKF_A2 as _Verkauf
6 {
7     key MitarbeiterId,
8     Name,
9     Geburtsdatum,
10    Rolle,
11    Abteilung,
12    Einstellungsdatum,
13    Gehalt,
14    Kontaktinfo,
15    CreatedBy,
16    CreatedAt,
17    LastChangedBy,
18    LastChangedAt,
19    LocalLastChangedAt,
20
21    _Verkauf
22 }
```

- Für **Besucher** (ZSS25\_R4C\_R\_BESUCH\_A2) stellt sich der Inhalt der Reusable View wie folgt dar:

```

D [A07] ZSS25_R4C_R_BESUCH_A2 ×
1 @AccessControl.authorizationCheck: #NOT_REQUIRED
2 @EndUserText.label: 'Reusable View für Besucher'
3 @ObjectModel: {
4     modelCategory: #BUSINESS_OBJECT,
5     compositionRoot: true,
6     semanticKey: ['BesucherId']
7 }
8 define root view entity ZSS25_R4C_R_BESUCH_A2
9     as select from ZSS25_R4C_V_BESUCH_A1
10
11
12 {
13     key BesucherId,
14     Name,
15     Geburtsdatum,
16     Email,
17     Telefon,
18     Adresse,
19     Registrierung,
20     CreatedBy,
21     CreatedAt,
22     LastChangedBy,
23     LastChangedAt,
24     LocalLastChangedAt
25 }
```

- Für **Film** (ZSS25\_R4C\_R\_FILM\_A2) stellt sich der Inhalt der Reusable View wie folgt dar:

```

D [A07] ZSS25_R4C_R_FILM_A2 X
1@AbapCatalog.viewEnhancementCategory: [#NONE]
2@AccessControl.authorizationCheck: #NOT_REQUIRED
3@EndUserText.label: 'Reusable View für Film'
4@Metadata.ignorePropagatedAnnotations: true
5@ObjectModel.usageType:{
6    serviceQuality: #X,
7    sizeCategory: #S,
8    dataClass: #MIXED
9}
10define root view entity ZSS25_R4C_R_FILM_A2 as select from ZSS25_R4C_V_FILM_A1
11
12{
13    key FilmId,
14    Titel,
15    Genre,
16    DauerMinuten,
17    Altersfreigabe,
18    Produktionsjahr,
19    Regisseur,
20    CreatedBy,
21    CreatedAt,
22    LastChangedBy,
23    LastChangedAt,
24    LocalLastChangedAt
25}

```

- Für **Kino** (ZSS25\_R4C\_R\_KINO\_A2) stellt sich der Inhalt der Reusable View wie folgt dar:

```

D [A07] ZSS25_R4C_R_KINO_A2 X
1@AccessControl.authorizationCheck: #NOT_REQUIRED
2@EndUserText.label: 'Reusable View für Kino'
3@Metadata.ignorePropagatedAnnotations: true
4define root view entity ZSS25_R4C_R_KINO_A2 as select from ZSS25_R4C_V_KINO_A1
5composition [1..*] of ZSS25_R4C_R_RAUM_A2 as _Raum
6{
7    key KinoId,
8    Name,
9    Adresse,
10   Stadt,
11   Plz,
12   AnzahlRaeume,
13   Eroeffnungsdatum,
14   Kontaktinfo,
15   CreatedBy,
16   CreatedAt,
17   LastChangedBy,
18   LastChangedAt,
19   LocalLastChangedAt,
20
21   _Raum
22}

```

- Für **Raum** (ZSS25\_R4C\_R\_RAUM\_A2) stellt sich der Inhalt der Reusable View wie folgt dar:

```

D [A07] ZSS25_R4C_R_RAUM_A2 ×
1 @AbapCatalog.viewEnhancementCategory: [#NONE]
2 @AccessControl.authorizationCheck: #NOT_REQUIRED
3 @EndUserText.label: 'Reusable View für Raum'
4 @Metadata.ignorePropagatedAnnotations: true
5 @ObjectModel.usageType:{ ...
6   serviceQuality: #X,
7   sizeCategory: #S,
8   dataClass: #MIXED
9 }
10 define view entity ZSS25_R4C_R_RAUM_A2 as select from ZSS25_R4C_V_RAUM_A1
11 association to parent ZSS25_R4C_R_KINO_A2 as _Kino
12   on _Kino.KinoId = $projection.KinoId
13 composition [1..*] of ZSS25_R4C_R_VORST_A2 as _Vorstellung
14 {
15   key RaumId,
16   KinoId,
17   Name,
18   StuehlAnzahlProReihe,
19   Sitzplaetze,
20   Ausstattung,
21   CreatedBy,
22   CreatedAt,
23   LastChangedBy,
24   LastChangedAt,
25   LocalLastChangedAt,
26
27   _Kino,
28   _Vorstellung
29 }
```

- Für **Ticket** (ZSS25\_R4C\_R\_TICKET\_A2) stellt sich der Inhalt der Reusable View wie folgt dar:

D [A07] ZSS25\_R4C\_R\_TICKET\_A2 X

```
1 @AbapCatalog.viewEnhancementCategory: [#NONE]
2 @AccessControl.authorizationCheck: #NOT_REQUIRED
3 @EndUserText.label: 'Reusable View für Ticket'
4 @Metadata.ignorePropagatedAnnotations: true
5 @ObjectModel.usageType:{
6     serviceQuality: #X,
7     sizeCategory: #S,
8     dataClass: #MIXED
9 }
10 define view entity ZSS25_R4C_R_TICKET_A2
11     as select from ZSS25_R4C_V_TICKET_A1
12
13
14 association [1..1] to ZSS25_R4C_R_BESUCH_A2 as _Besucher
15     on $projection.BesucherId = _Besucher.BesucherId
16
17 association to parent ZSS25_R4C_R_VORST_A2 as _Vorstellung
18     on $projection.VorstellungId = _Vorstellung.VorstellungId
19
20 association to ZSS25_R4C_R_KINO_A2 as _Kino
21     on $projection.KinoId = _Kino.KinoId
22
23 association to ZSS25_R4C_R_RAUM_A2 as _Raum
24     on $projection.RaumId = _Raum.RaumId
25
```

```
26 {
27     key TicketId,
28     VorstellungId,
29     BesucherId,
30     RaumId,
31     KinoId,
32     Sitzplatznummer,
33     Preis,
34     TicketTyp,
35     QrCode,
36     Status,
37     CreatedBy,
38     CreatedAt,
39     LastChangedBy,
40     LastChangedAt,
41     LocalLastChangedAt,
42
43     _Besucher,
44     _Vorstellung,
45     _Kino,
46     _Raum
47
48 }
```

- Für **Verkauf** (ZSS25\_R4C\_R\_VERKF\_A2) stellt sich der Inhalt der Reusable View wie folgt dar:

D [A07] ZSS25\_R4C\_R\_VERKF\_A2 X

```
1@AccessControl.authorizationCheck: #NOT_REQUIRED
2@EndUserText.label: 'Reusable View für Verkauf'
3@Metadata.ignorePropagatedAnnotations: true
4define view entity ZSS25_R4C_R_VERKF_A2 as select from ZSS25_R4C_V_VERKF_A1
5association to parent ZSS25_R4C_R_MITARB_A2 as _Mitarbeiter
6    on _Mitarbeiter.MitarbeiterId = $projection.MitarbeiterId
7association [1..1] to ZSS25_R4C_R_TICKET_A2 as _Ticket
8    on _Ticket.TicketId = $projection.TicketId
9{
10    key VerkaufId,
11    TicketId,
12    MitarbeiterId,
13    MitarbeiterName,
14    BesucherName,
15    Verkaufsdatum,
16    Zahlungsart,
17    TicketPreis,
18    Rabatt,
19    Betrag,
20    BesucherAltersgruppe,
21    BesucherIstErlaubt,
22    TicketSitzplatznummer,
23    BesucherSitzReihe,
24    TicketIstGueltig,
25    Zeitkategorie,
26    TicketStatus,
27    CreatedBy,
28    CreatedAt,
29    LastChangedBy,
30    LastChangedAt,
31    LocalLastChangedAt,
32    _Mitarbeiter,
33    _Ticket
34}
35
```

- Für **Vorstellung** (ZSS25\_R4C\_R\_VORST\_A2) stellt sich der Inhalt der Reusable View wie folgt dar:

```

D [A07] ZSS25_R4C_R_VORST_A2 ×
1 @AbapCatalog.viewEnhancementCategory: [#NONE]
2 @AccessControl.authorizationCheck: #NOT_REQUIRED
3 @EndUserText.label: 'Reusable View für Vorstellung'
4 @Metadata.ignorePropagatedAnnotations: true
5 @ObjectModel.usageType:{
6   serviceQuality: #X,
7   sizeCategory: #S,
8   dataClass: #MIXED}
9 define view entity ZSS25_R4C_R_VORST_A2 as select from ZSS25_R4C_V_VORST_A1
10 association to parent ZSS25_R4C_R_RAUM_A2 as _Raum
11   on _Raum.RaumId = $projection.RaumId
12 association to ZSS25_R4C_R_KINO_A2 as _Kino
13   on $projection.KinoId = _Kino.KinoId
14 association [1..1] to ZSS25_R4C_R_FILM_A2 as _Film
15   on $projection.FilmId = _Film.FilmId
16
17 composition [1..*] of ZSS25_R4C_R_TICKET_A2 as _Ticket
18 {
19   key VorstellungId,
20   RaumId,
21   KinoId,
22   FilmId,
23   Sprache,
24   Datum,
25   Startzeit,
26   Endzeit,
27   Untertitel,
28   VorstellungFormat,
29   Status,
30   CreatedBy,
31   CreatedAt,
32   LastChangedBy,
33   LastChangedAt,
34   LocalLastChangedAt,
35
36   _Raum,
37   _Kino,
38   _Film,
39   _Ticket
40 }

```

### Notiz:

- Nachdem die Reusable View erstellt wurde, muss sie mit Strg + S gespeichert und anschließend über das Aktivierungssymbol in SAP aktiviert werden.
- Nach erfolgreicher Aktivierung ist die Reusable View im Projektbaum unter dem Pfad Core Data Services → Data Definitions zu finden.

## 3.6. Projection View (Consumption View)

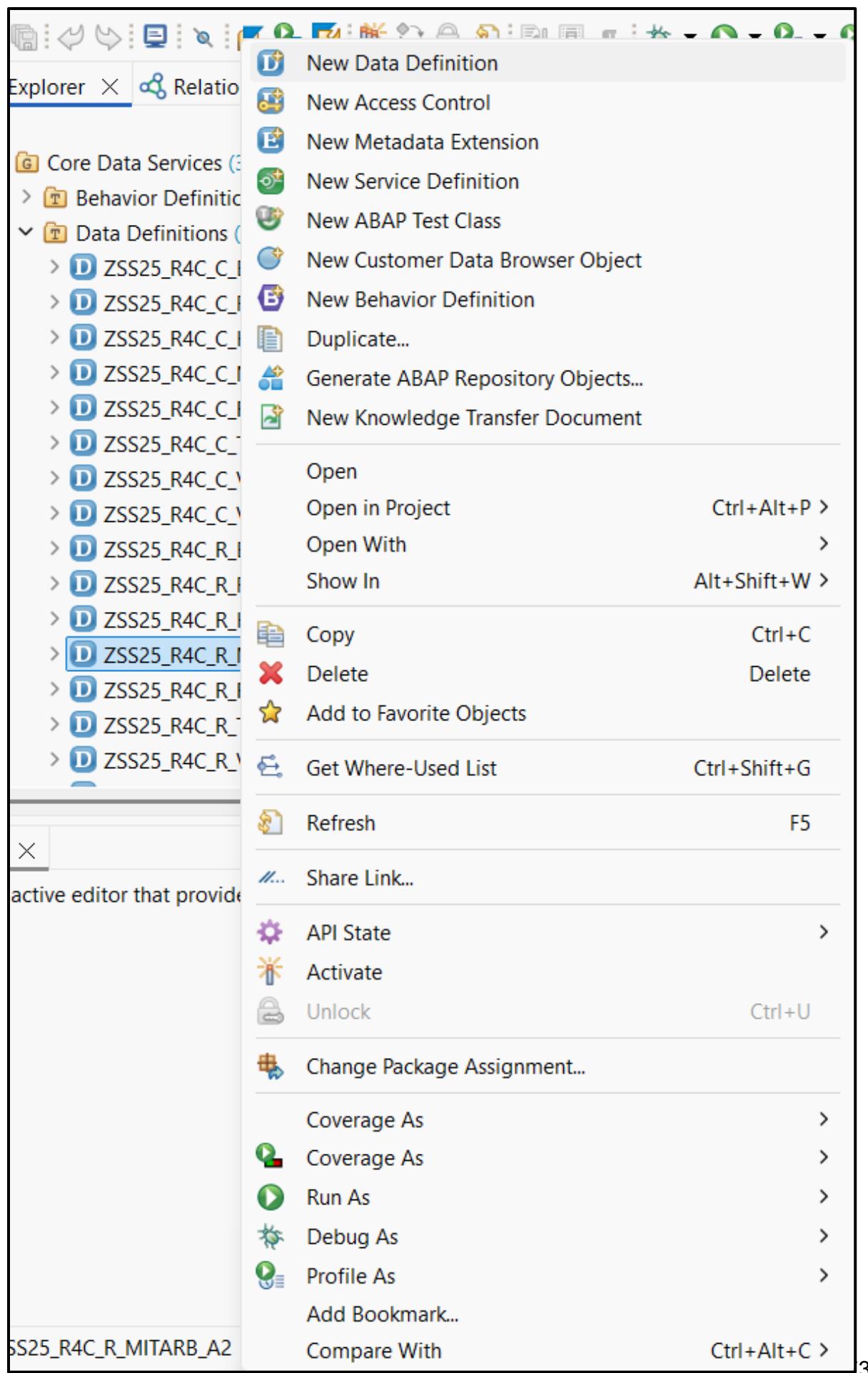
Die Projection View stellt die jeweilige Entität für die Nutzung in der Benutzeroberfläche oder über OData-Services bereit. Sie basiert auf der zugrunde liegenden Interface View und bildet die Konsumschicht im RAP-Modell.

In dieser View können Daten anwendungs- oder benutzerspezifisch gefiltert, eingeschränkt oder angereichert werden - etwa durch die Auswahl bestimmter Felder, die Definition zusätzlicher Bedingungen oder die Integration von Annotationen für Fiori/UI5-Anwendungen.

Damit dient die Projection View als Bindeglied zwischen dem fachlichen Datenmodell und der Präsentationsebene und ermöglicht eine gezielte, performante Bereitstellung der relevanten Informationen für Endanwender oder externe Services.

### **Schritt-für-Schritt-Anleitung:**

Rechtsklick auf eine Reusable View → New Data Definition → Referenced Object, Name und Beschreibung eingeben, z. B.: Referenced Object: ZSS25\_R4C\_R\_MITARB\_A2 || Name: ZSS25\_R4C\_C\_MITARB\_A3 || Beschreibung: Consumption View für Mitarbeiter → Next → Next → defineProjectionView auswählen → Finish



New Data Definition

### Data Definition

Create a data definition

Project: \* A07\_200\_learn-195\_en

Package: \* \$SS25\_RAP4CINEMA

Add to favorite packages

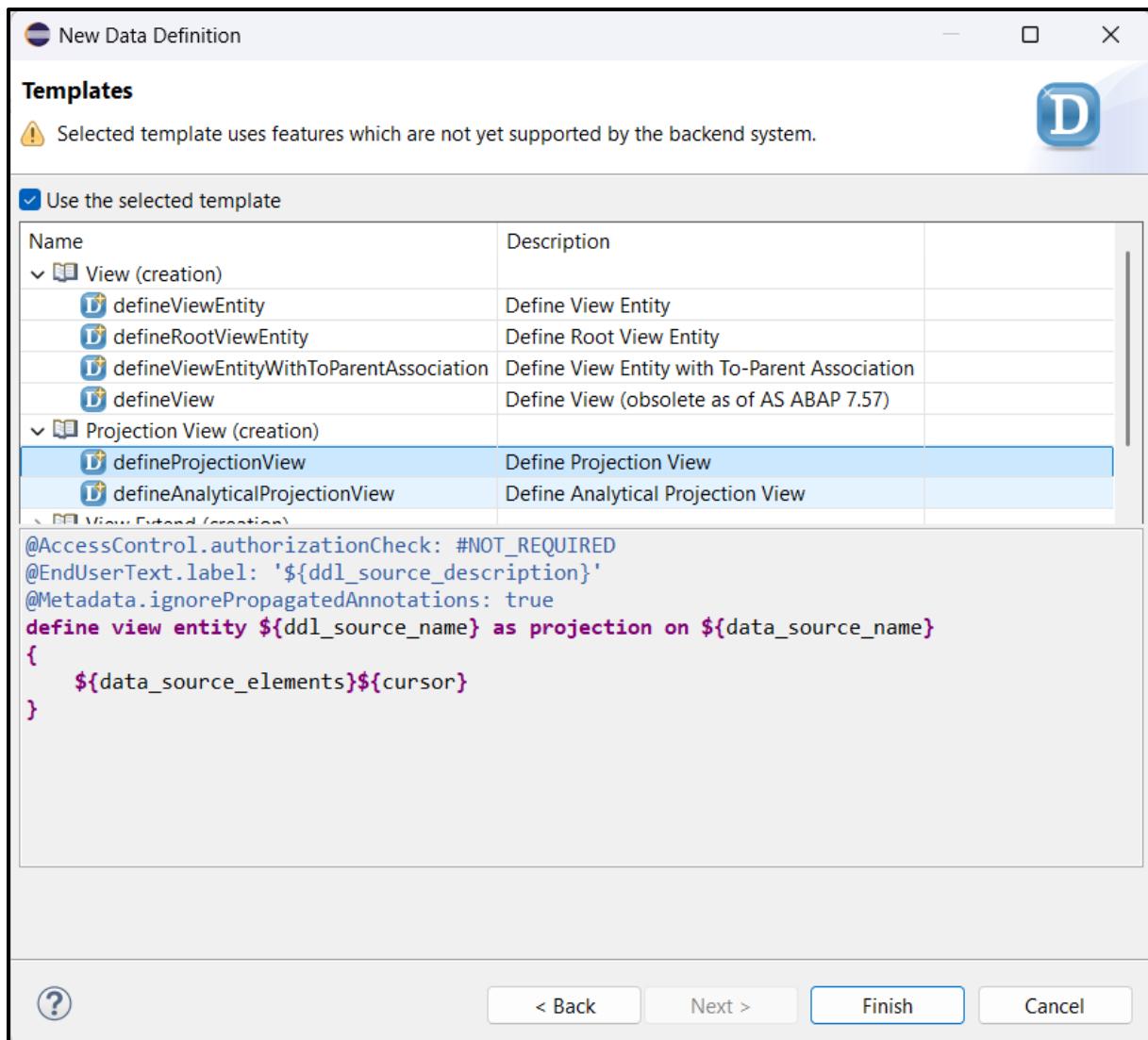
Name: \* ZSS25\_R4C\_C\_MITARB\_A3

Description: \* Consumption View für Mitarbeiter

Original Language: EN

Referenced Object: ZSS25\_R4C\_R\_MITARB\_A2

< Back



- Für **Mitarbeiter** (ZSS25\_R4C\_C\_MITARB\_A3) stellt sich der Inhalt der Consumption View wie folgt dar:

```

D [A07] ZSS25_R4C_C_MITARB_A3 X
1 @AccessControl.authorizationCheck: #NOT_REQUIRED
2 @EndUserText.label: 'Consumption View für Mitarbeiter'
3 @Metadata.ignorePropagatedAnnotations: true
4 @Metadata.allowExtensions: true
5 @Search.searchable: true
6 define root view entity ZSS25_R4C_C_MITARB_A3 provider contract transactional_query
7   as projection on ZSS25_R4C_R_MITARB_A2
8 {
9   @EndUserText.label: 'Mitarbeiternummer'
10  @Search.defaultSearchElement: true
11  key MitarbeiterId,
12  Name,
13  Geburtsdatum,
14  Rolle,
15  Abteilung,
16  Einstellungsdatum,
17  Gehalt,
18  Kontaktinfo,
19  CreatedBy,
20  CreatedAt,
21  LastChangedBy,
22  LastChangedAt,
23  LocalLastChangedAt,
24
25  /* Associations */
26  _Verkauf : redirected to composition child ZSS25_R4C_C_VERKF_A3
27 }
28

```

- Für **Besucher** (ZSS25\_R4C\_C\_BESUCH\_A3) stellt sich der Inhalt der Consumption View wie folgt dar:

```

D [A07] ZSS25_R4C_C_BESUCH_A3 X
1 @AccessControl.authorizationCheck: #NOT_REQUIRED
2 @EndUserText.label: 'Consumption View für Besucher'
3 @Metadata.ignorePropagatedAnnotations: true
4 @Metadata.allowExtensions: true
5 @Search.searchable: true
6
7 define root view entity ZSS25_R4C_C_BESUCH_A3
8 provider contract transactional_query
9 as projection on ZSS25_R4C_R_BESUCH_A2
10 {
11     @EndUserText.label: 'Besuchernummer'
12     @Search.defaultSearchElement: true
13     key BesucherId,
14     Name,
15     Geburtsdatum,
16     Email,
17     Telefon,
18     Adresse,
19     Registrierung,
20     CreatedBy,
21     CreatedAt,
22     LastChangedBy,
23     LastChangedAt,
24     LocalLastChangedAt
25
26
27 }

```

- Für **Film** (ZSS25\_R4C\_C\_FILM\_A3) stellt sich der Inhalt der Consumption View wie folgt dar:

```

D [A07] ZSS25_R4C_C_FILM_A3 X
1 @AccessControl.authorizationCheck: #NOT_REQUIRED
2 @EndUserText.label: 'Consumption View für Film'
3 @Metadata.ignorePropagatedAnnotations: true
4 @Metadata.allowExtensions: true
5 @Search.searchable: true
6 define root view entity ZSS25_R4C_C_FILM_A3 provider contract transactional_query
7 as projection on ZSS25_R4C_R_FILM_A2
8 {
9     @EndUserText.label: 'Filmmnummer'
10    @Search.defaultSearchElement: true
11    key FilmId,
12    Titel,
13    Genre,
14    DauerMinuten,
15    Altersfreigabe,
16    Produktionsjahr,
17    Regisseur,
18    CreatedBy,
19    CreatedAt,
20    LastChangedBy,
21    LastChangedAt,
22    LocalLastChangedAt
23
24 }

```

- Für **Kino** (ZSS25\_R4C\_C\_KINO\_A3) stellt sich der Inhalt der Consumption View wie folgt dar:

```
D [A07] ZSS25_R4C_C_KINO_A3 X
1 @AccessControl.authorizationCheck: #NOT_REQUIRED
2 @EndUserText.label: 'Consumption View für Kino'
3 @Metadata.ignorePropagatedAnnotations: true
4 @Metadata.allowExtensions: true
5 @Search.searchable: true
6 define root view entity ZSS25_R4C_C_KINO_A3 provider contract transactional_query
7     as projection on ZSS25_R4C_R_KINO_A2
8 {
9     @EndUserText.label: 'Kinonummer'
10    @Search.defaultSearchElement: true
11    key KinoId,
12    Name,
13    Adresse,
14    Stadt,
15    Plz,
16    AnzahlRaeume,
17    Eroeffnungsdatum,
18    Kontaktinfo,
19    CreatedBy,
20    CreatedAt,
21    LastChangedBy,
22    LastChangedAt,
23    LocalLastChangedAt,
24
25    /* Associations */
26    _Raum: redirected to composition child ZSS25_R4C_C_RAUM_A3
27 }
28
```

- Für **Raum** (ZSS25\_R4C\_C\_RAUM\_A3) stellt sich der Inhalt der Consumption View wie folgt dar:

```

D [A07] ZSS25_R4C_C_RAUM_A3 ×
1⊕ @AccessControl.authorizationCheck: #NOT_REQUIRED
2 @EndUserText.label: 'Consumption View für Raum'
3 @Metadata.ignorePropagatedAnnotations: true
4 @Metadata.allowExtensions: true
5 @Search.searchable: true
6 define view entity ZSS25_R4C_C_RAUM_A3 as projection on ZSS25_R4C_R_RAUM_A2
7 {
8⊕   @EndUserText.label: 'Raumnummer'
9     @Search.defaultSearchElement: true
10    key RaumId,
11    KinoId,
12    Name,
13    StuehlAnzahlProReihe,
14    Sitzplaetze,
15    Ausstattung,
16    CreatedBy,
17    CreatedAt,
18    LastChangedBy,
19    LastChangedAt,
20    LocalLastChangedAt,
21
22  /* Associations */
23  _Kino : redirected to parent ZSS25_R4C_C_KINO_A3,
24  _Vorstellung: redirected to composition child ZSS25_R4C_C_VORST_A3
25
26

```

- Für **Ticket** (ZSS25\_R4C\_C\_TICKET\_A3) stellt sich der Inhalt der Consumption View wie folgt dar:

D [A07] ZSS25\_R4C\_C\_TICKET\_A3 X

```
1 @AccessControl.authorizationCheck: #NOT_REQUIRED
2 @EndUserText.label: 'Consumption View für Ticket'
3 @Metadata.ignorePropagatedAnnotations: true
4 @Metadata.allowExtensions: true
5 @Search.searchable: true
6 define view entity ZSS25_R4C_C_TICKET_A3
7   as projection on ZSS25_R4C_R_TICKET_A2
8 {
9   @EndUserText.label: 'Ticketnummer'
10  @Search.defaultSearchElement: true
11  key TicketId,
12  VorstellungId,
13  BesucherId,
14  RaumId,
15  KinoId,
16  Sitzplatznummer,
17  Preis,
18  TicketTyp,
19  QrCode,
20  Status,
21  CreatedBy,
22  CreatedAt,
23  LastChangedBy,
24  LastChangedAt,
25  LocalLastChangedAt,
26
27 /* Associations */
28 _Besucher : redirected to ZSS25_R4C_C_BESUCH_A3,
29 _Vorstellung : redirected to parent ZSS25_R4C_C_VORST_A3 ,
30 _Kino: redirected to ZSS25_R4C_C_KINO_A3,
31 _Raum: redirected to ZSS25_R4C_C_RAUM_A3
32 }
33 }
```

- Für **Verkauf** (ZSS25\_R4C\_C\_VERKF\_A3) stellt sich der Inhalt der Consumption View wie folgt dar:

```

D [A07] ZSS25_R4C_C_VERKF_A3 X
1 @AccessControl.authorizationCheck: #NOT_REQUIRED
2 @EndUserText.label: 'Consumption View für Verkauf'
3 @Metadata.ignorePropagatedAnnotations: true
4 @Metadata.allowExtensions: true
5 @Search.searchable: true
6 define view entity ZSS25_R4C_C_VERKF_A3 as projection on ZSS25_R4C_R_VERKF_A2
7 {
8     @EndUserText.label: 'Rechnungsnummer'
9     @Search.defaultSearchElement: true
10    key VerkaufId,
11    TicketId,
12    MitarbeiterId,
13    MitarbeiterName,
14    BesucherName,
15    Verkaufsdatum,
16    Zahlungsart,
17    TicketPreis,
18    Rabatt,
19    Betrag,
20    BesucherAltersgruppe,
21    BesucherIstErlaubt,
22    TicketSitzplatznummer,
23    BesucherSitzReihe,
24    TicketIstGueltig,
25    Zeitkategorie,
26    TicketStatus,
27    CreatedBy,
28    CreatedAt,
29    LastChangedBy,
30    LastChangedAt,
31    LocalLastChangedAt,
32
33    /* Associations */
34    _Mitarbeiter : redirected to parent ZSS25_R4C_C_MITARB_A3,
35    _Ticket: redirected to ZSS25_R4C_C_TICKET_A3
36
37 }

```

- Für **Vorstellung** (ZSS25\_R4C\_C\_VORST\_A3) stellt sich der Inhalt der Consumption View wie folgt dar:

```

D [A07] ZSS25_R4C_C_VORST_A3 X
1 @AccessControl.authorizationCheck: #NOT_REQUIRED
2 @EndUserText.label: 'Consumption View für Vorstellung'
3 @Metadata.ignorePropagatedAnnotations: true
4 @Metadata.allowExtensions: true
5 @Search.searchable: true
6 define view entity ZSS25_R4C_C_VORST_A3 as projection on ZSS25_R4C_R_VORST_A2
7 {
8
9 @EndUserText.label: 'Vorstellungnummer'
10 @Search.defaultSearchElement: true
11 key VorstellungId,
12 RaumId,
13 KinoId,
14 FilmId,
15 Sprache,
16 Datum,
17 Startzeit,
18 Endzeit,
19 Untertitel,
20 VorstellungFormat,
21 Status,
22 CreatedBy,
23 CreatedAt,
24 LastChangedBy,
25 LastChangedAt,
26 LocalLastChangedAt,
27
28 /* Associations */
29 _Raum : redirected to parent ZSS25_R4C_C_RAUM_A3,
30   _Film: redirected to ZSS25_R4C_C_FILM_A3,
31   _Kino: redirected to ZSS25_R4C_C_KINO_A3,
32   _Ticket: redirected to composition child ZSS25_R4C_C_TICKET_A3
33 }

```

### Notiz:

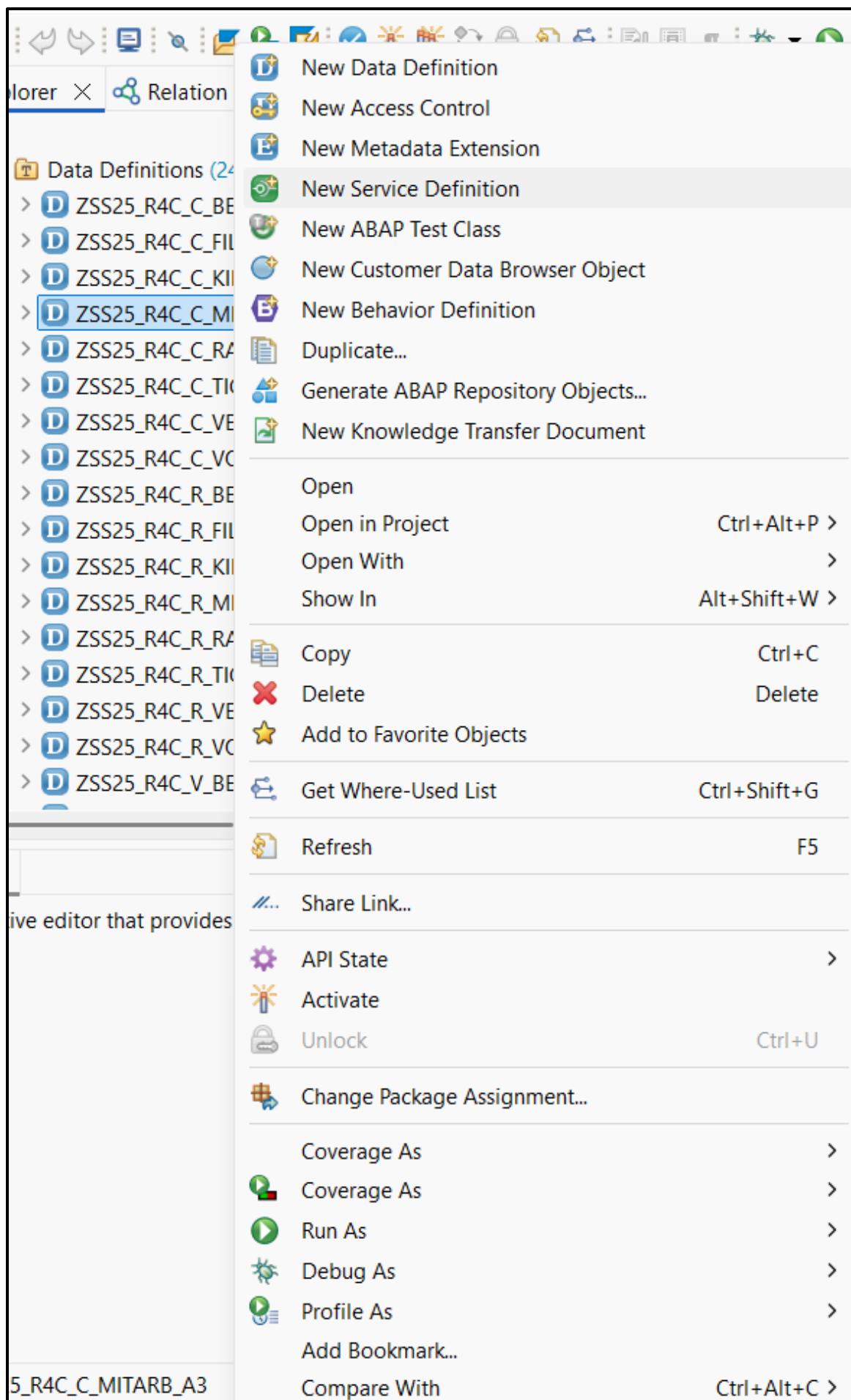
- Nachdem die Consumption View erstellt wurde, muss sie mit Strg + S gespeichert und anschließend über das Aktivierungssymbol in SAP aktiviert werden.
- Nach erfolgreicher Aktivierung ist die Consumption View im Projektbaum unter dem Pfad Core Data Services → Data Definitions zu finden.

## 3.7. Service Definition

Die Service Definition legt fest, welche Entitäten, Views und Aktionen eines Business-Objekts nach außen bereitgestellt werden. Sie bildet die konzeptionelle Schnittstelle zwischen dem Datenmodell und der Service-Exposition. In ihr wird definiert, welche Daten konsumierbar sind - beispielsweise für Fiori/UI5-Anwendungen oder externe Systeme. Durch diese klare Abgrenzung wird sichergestellt, dass nur relevante und geprüfte Informationen über die Service-Schnittstelle zugänglich sind. Damit schafft die Service Definition die Grundlage für eine strukturierte, sichere und modulare Bereitstellung von Geschäftsobjekten innerhalb des RAP-Modells.

### **Schritt-für-Schritt-Anleitung:**

Rechtsklick auf eine Consumption View → New Service Definition → Referenced Object, Source Type, Name und Beschreibung eingeben, z. B.: Referenced Object: ZSS25\_R4C\_C\_KINO\_A3 || Source Type: Defintion || Name: ZSS25\_R4C\_SD\_KINO\_A4 || Beschreibung: Service Definition für Kino → Next → Next → defineService auswählen → Finish



New Service Definition

### Service Definition

Create a service definition



Project: \* A07\_200\_learn-195\_en

Package: \* SSS25\_RAP4CINEMA

Add to favorite packages

Name: \* ZSS25\_R4C\_SD\_KINO\_A4

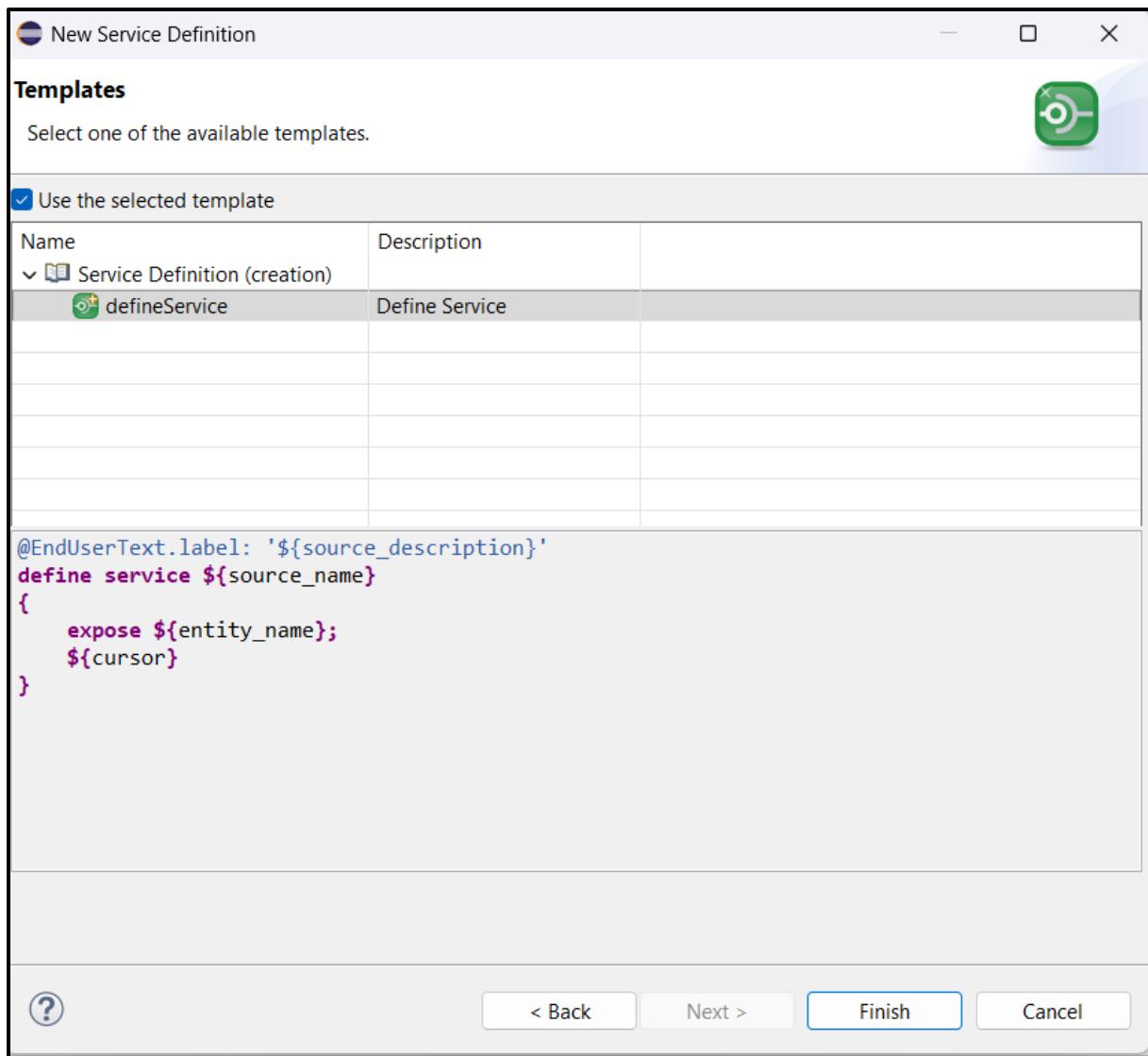
Description: \* Service Definition für Kino

Original Language: EN

Source Type: Definition

Referenced Object: ZSS25\_R4C\_C\_KINO\_A3

 < Back Next > Finish Cancel



- Für **Kino** (ZSS25\_R4C\_SD\_KINO\_A4) stellt sich der Inhalt der Service-Definition wie folgt dar:

```

[A07] ZSS25_R4C_SD_KINO_A4 X
1 @EndUserText.label: 'Service Definition für Kino'
2 define service ZSS25_R4C_SD_KINO_A4 {
3     expose ZSS25_R4C_C_VERKF_A3;
4     expose ZSS25_R4C_C_KINO_A3;
5     expose ZSS25_R4C_C_RAUM_A3;
6     expose ZSS25_R4C_C_VORST_A3;
7     expose ZSS25_R4C_C_FILM_A3;
8     expose ZSS25_R4C_C_BESUCH_A3;
9     expose ZSS25_R4C_C_TICKET_A3;
10    expose ZSS25_R4C_C_MITARB_A3;
11 }

```

### Notiz:

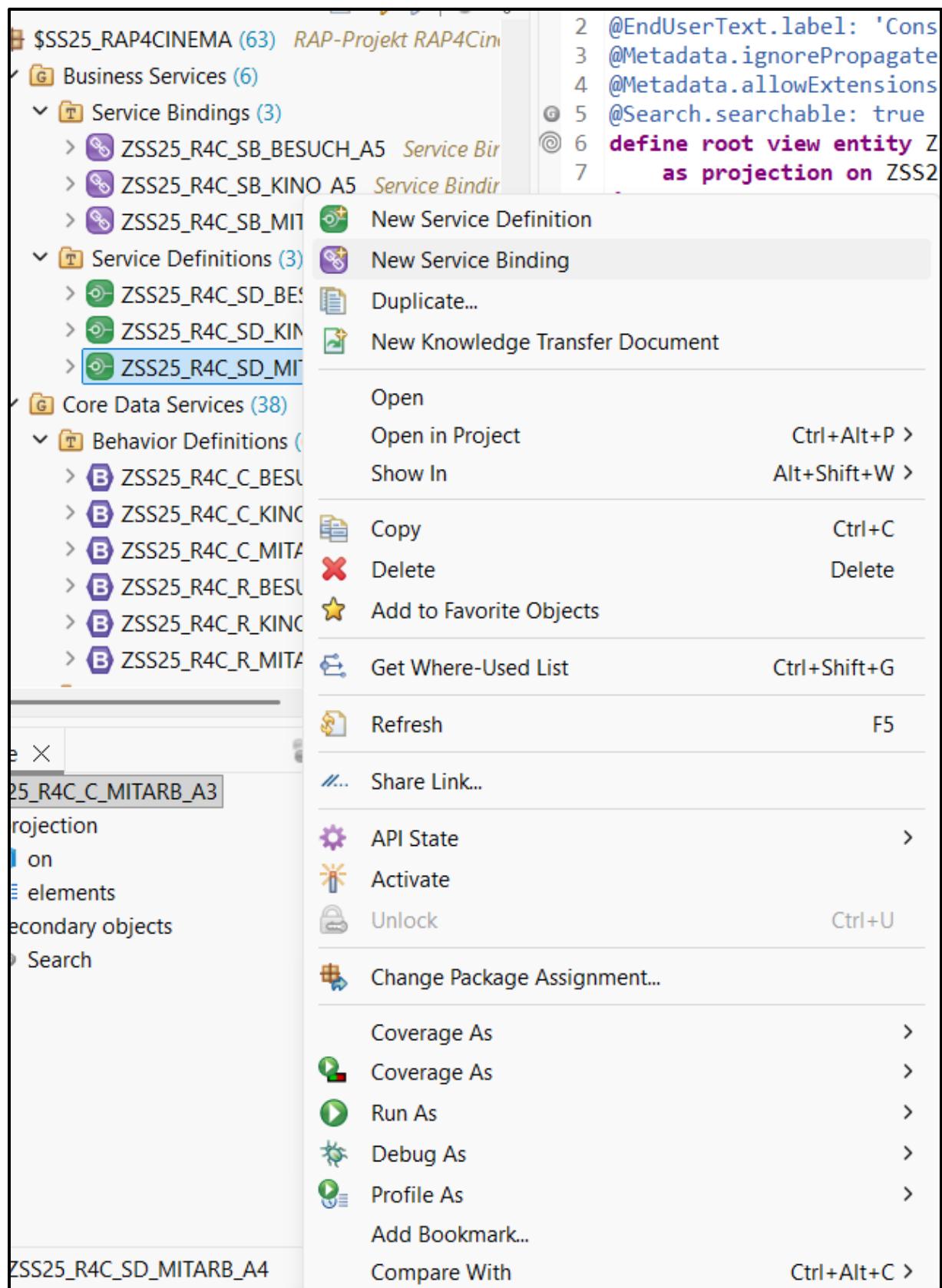
- Nachdem die Service Definition erstellt wurde, muss sie mit Strg + S gespeichert und anschließend über das Aktivierungssymbol in SAP aktiviert werden.
- Nach erfolgreicher Aktivierung ist die Service Definition im Projektbaum unter dem Pfad Business Services → Service Definitions zu finden.

## 3.8. Service Binding

Das Service Binding definiert die technische Bereitstellung der in der Service Definition beschriebenen Inhalte. Es legt fest, über welches Protokoll (z. B. OData V4) und in welcher Form die Services verfügbar gemacht werden. Durch das Aktivieren eines Service Bindings wird der definierte Service lauffähig und kann direkt konsumiert werden - etwa von Fiori/UI5-Anwendungen oder anderen externen Clients. Das Service Binding bildet somit die technische Expositionsschicht des RAP-Modells und stellt sicher, dass Daten und Geschäftslogik standardisiert, sicher und performant nach außen bereitgestellt werden.

### **Schritt-für-Schritt-Anleitung:**

Rechtsklick auf eine Service Definition → New Service Binding → Referenced Object, Binding Type, Name und Beschreibung eingeben, z. B.: Referenced Object: ZSS25\_R4C\_SD\_KINO\_A4 || Binding Type: OData V4 - UI || Name: ZSS25\_R4C\_SB\_KINO\_A5 || Beschreibung: Service Binding für Kino → Next → Finish



 New Service Binding

**Service Binding**

Create a Service Binding.



Project: \* A07\_200\_learn-195\_en

Package: \* SSS25\_RAP4CINEMA

Add to favorite packages

Name: \* ZSS25\_R4C\_SB\_KINO\_A5

Description: \* Service Binding für Kino

Original Language: EN

Binding Type: \* OData V4 - UI

Service Definition: \* ZSS25\_R4C\_SD\_KINO\_A4



 New Service Binding

### Select Transport Request

 No change recording enabled for package \$SS25\_RAP4CINEMA



Choose from requests in which I am involved Configure Columns

Transport Request	Owner	Target	Description	CTS Proj

Create a new request

Request Description: \*

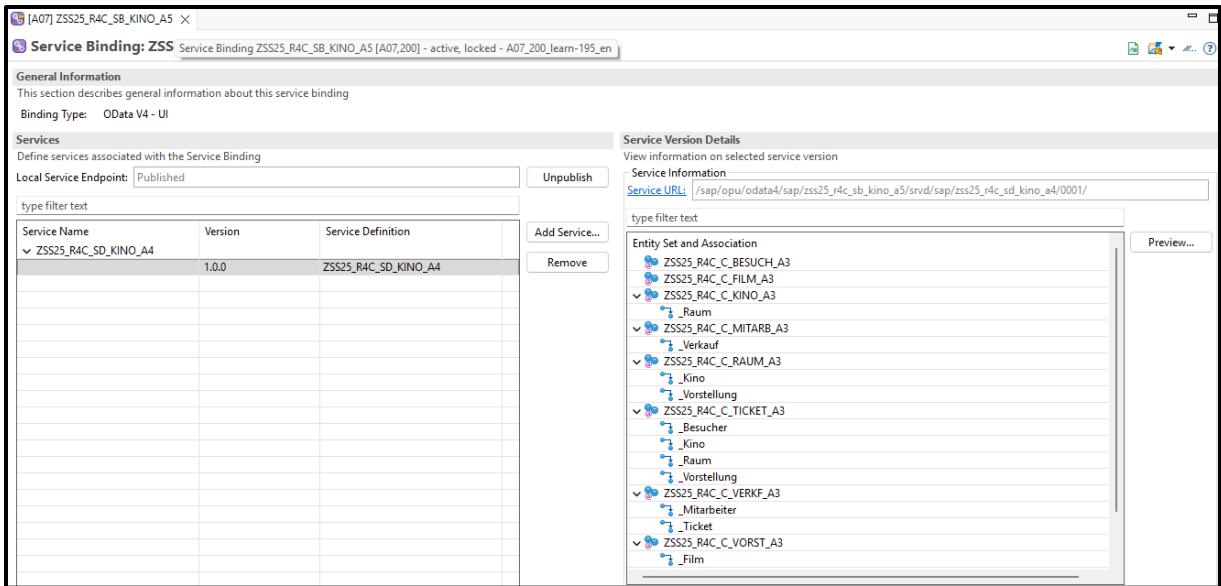
CTS Project:  Browse...

Enter a request number

Request Number:  Browse...

?   < Back   Next >   Finish   Cancel

- Für **Kino** (ZSS25\_R4C\_SB\_KINO\_A5) sieht das User Interface der Service-Binding wie folgt aus:



### Notiz:

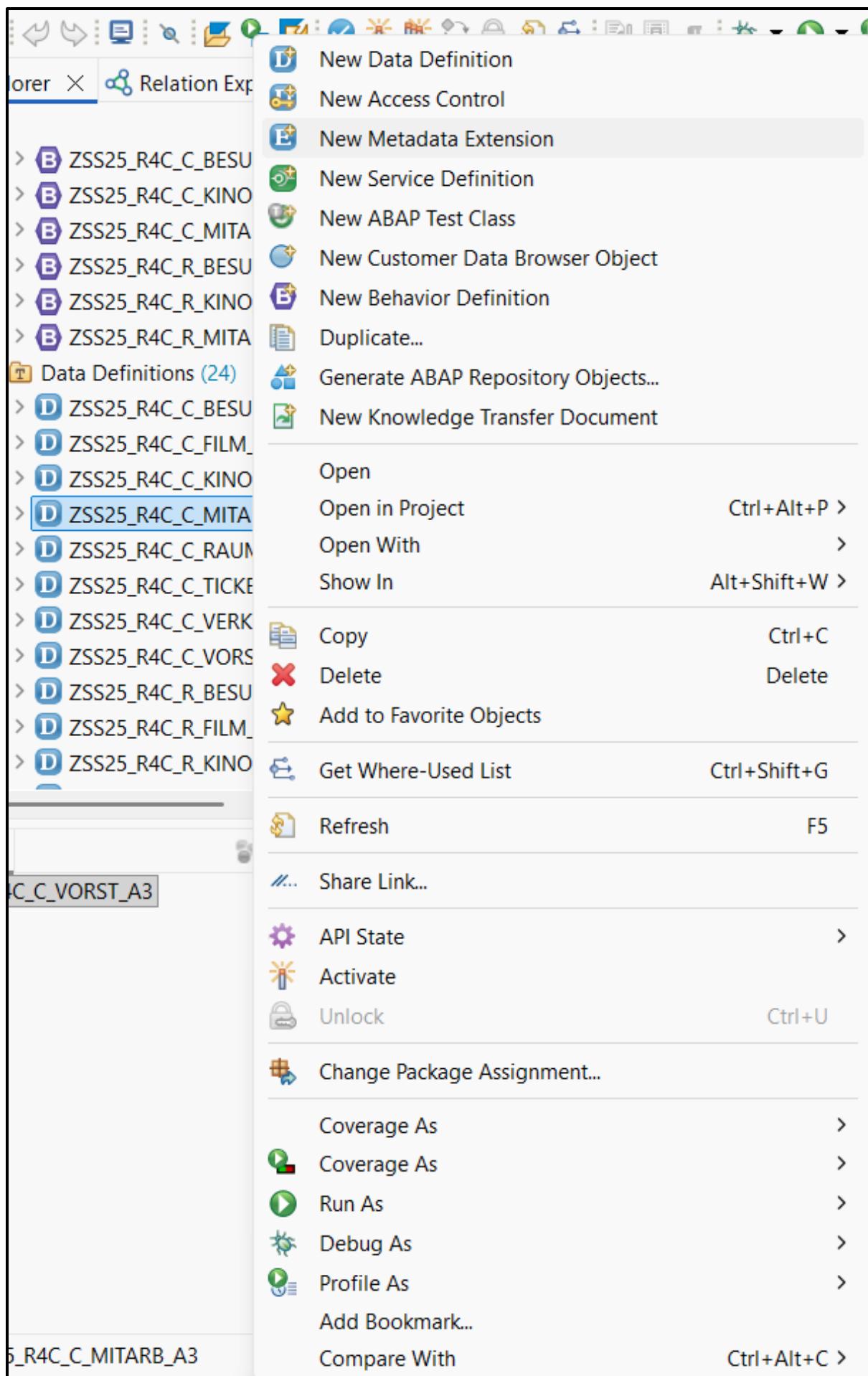
- Nachdem die Service Binding erstellt wurde, muss sie mit Strg + S gespeichert und anschließend über das Aktivierungssymbol in SAP aktiviert und publiziert werden. Danach den erzeugten Link im Service Binding kopieren.
- Nach erfolgreicher Aktivierung ist die Service Binding im Projektbaum unter dem Pfad Business Services → Service Bindings zu finden.
- Um die Service Binding im Browser zu nutzen, müssen die folgenden Schritte ausgeführt werden:
  - 1) Die kopierte URL aktivieren.
  - 2) Die IP-Adresse unter <https://fk-feedback-test.mni.thm.de/sap-proxy-unlock/> für 24 Stunden freischalten.
  - 3) Anschließend <https://fk-feedback-test.mni.thm.de:4343/> öffnen und dort die kopierte URL einfügen.

## 3.9. Metadata Extension (UI-Annotationen)

Die Metadata Extension erweitert CDS-Views um UI-relevante Annotationen, ohne dabei die zugrunde liegende Datenstruktur zu verändern. Sie dient der Trennung von Datenmodell und Darstellungsschicht und ermöglicht eine flexible Anpassung der Benutzeroberfläche, ohne in die fachliche Logik einzugreifen. Über UI-Annotationen werden beispielsweise Feldbeschriftungen, Tabellenstrukturen, Gruppierungen, Sortierungen oder Sichtbarkeiten definiert. Dadurch kann das Erscheinungsbild und Verhalten einer Fiori/UI5-Anwendung gezielt gesteuert werden. Im Projekt RAP4Cinema trägt die Metadata Extension wesentlich zur UI-Optimierung und Wiederverwendbarkeit bei, indem sie eine klare, wartbare Verbindung zwischen den CDS-Views und der Benutzeroberfläche schafft - im Sinne des Clean Architecture-Ansatzes.

**Schritt-für-Schritt-Anleitung:**

Rechtsklick auf eine Consumption View → New Metadata Extension → Extended Entity, Name und Beschreibung eingeben, z. B.: Name: ZSS25\_R4C\_ME\_MITARB\_A6 || Extended Entity: ZSS25\_R4C\_SD\_MITARB\_A4 || Beschreibung: Metadata Extension für Mitarbeiter → Next → Next → annotateEntity auswählen → Finish



New Metadata Extension

### Metadata Extension

Create a metadata extension



Project: \* A07\_200\_learn-195\_en

Package: \* \$SS25\_RAP4CINEMA

Add to favorite packages

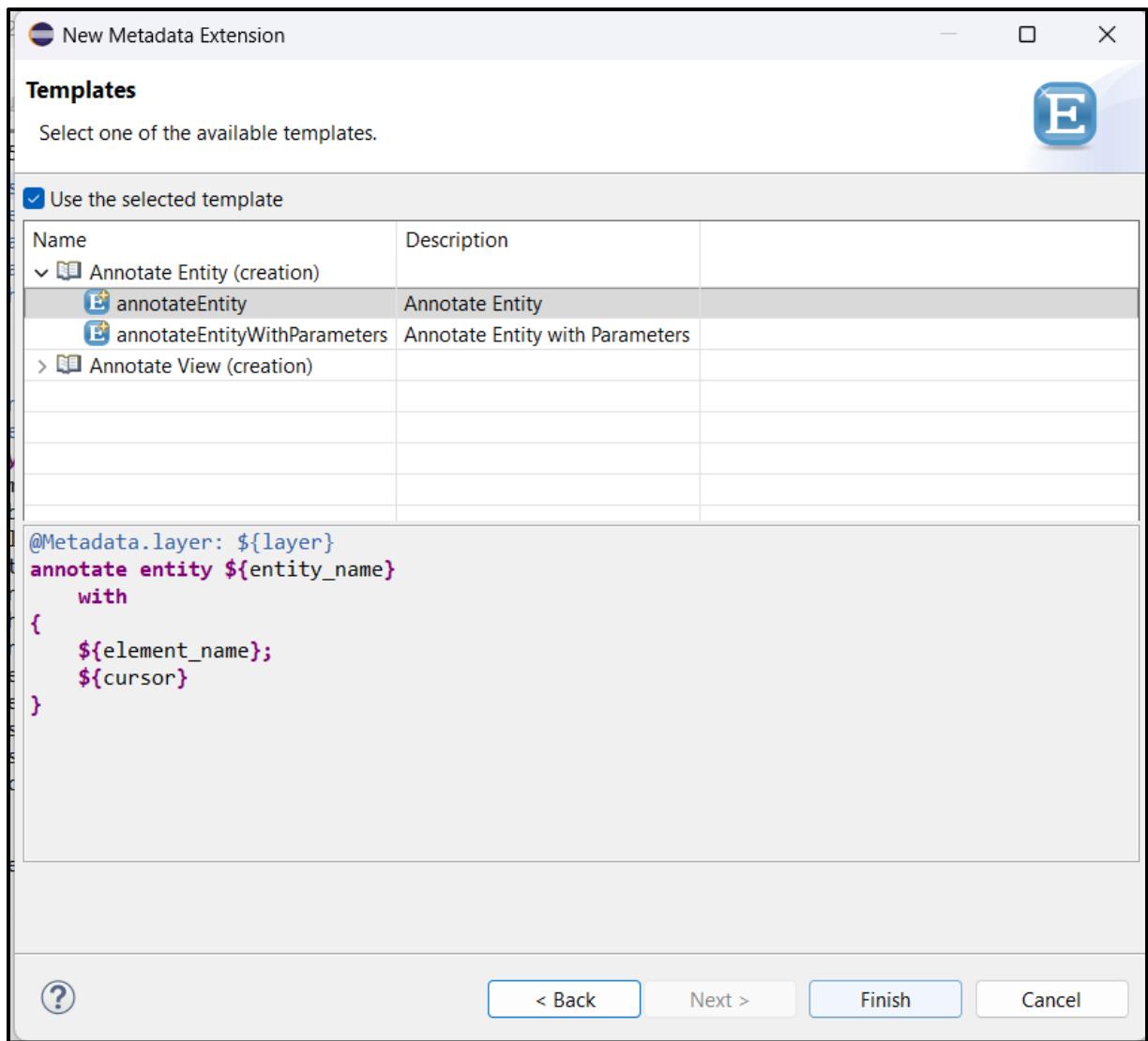
Name: \* ZSS25\_R4C\_ME\_MITARB\_A6

Description: \* Metadata Extension für Mitarbeiter

Original Language: EN

Extended Entity: ZSS25\_R4C\_C\_MITARB\_A3





- Für **Mitarbeiter** (ZSS25\_R4C\_ME\_MITARB\_A6) stellt sich der Inhalt der Metadata Extension wie folgt dar:

```

E [A07] ZSS25_R4C_ME_MITARB_A6 X
1 @Metadata.layer: #CUSTOMER                                // Diese Annotation legt fest, dass die Anpassung auf Kundenebene erfolgt.
2                                         // Dadurch bleibt die SAP-Standarddefinition (Layer #CORE) unverändert.
3
4 @UI: {
5   headerInfo: {
6     typeName: 'Mitarbeiter',
7     typeNamePlural: 'Mitarbeiter'
8   }
9 }
10
11 @Search.searchable: true                               // Aktiviert die globale Fiori-Suche für diese Entität.
12
13 annotate entity ZSS25_R4C_C_MITARB_A3 with
14 {
15@ // -----
16 // FACETS (Abschnitte auf der Object Page im Fiori-UI)
17 // -----
18 @UI.facet: [
19   { position: 10,                                     // Reihenfolge (kleinere Zahl = weiter oben)
20     label: 'Mitarbeiterinformationen',                // Titel des Abschnitts im UI
21     type: '#IDENTIFICATION_REFERENCE'               // Verknüpfung mit Identifikationsfeldern (z. B. Stammdaten)
22   }
23 ]
24
25 // -----
26 // FELDER MIT UI- UND SUCH-ANNOTATIONEN
27 // -----
28
29 // --- Mitarbeiter-ID ---
30 @EndUserText.label: 'Mitarbeiter-ID'                  // Beschriftung im UI
31 MitarbeiterId;
32
33@ // --- Name ---
34 @EndUserText.label: 'Name'                           // Feld erscheint im Such-/Filterbereich (List Report)
35 @UI.selectionField: [{ position: 10 }]              // Feld erscheint im oberen Informationsbereich (Object Page)
36 @UI.identification: [{ position: 10 }]              // Feld wird in der Volltextsuche berücksichtigt
37 @Search: { defaultSearchElement: true,             // Unscharfe Suche erlaubt (z. B. bei Tippfehlern)
38           fuzzinessThreshold: 0.7 }                  // Feld erscheint in der List Report Tabelle
39 @UI.lineItem: [{ position: 10 }]
40 Name;

```

```

41
42@ // --- Geburtsdatum ---
43 @EndUserText.label: 'Geburtsdatum'                   // Im Detailbereich sichtbar
44 @UI.identification: [{ position: 20 }]              // In der Tabelle angezeigt
45 @UI.lineItem: [{ position: 20 }]
46 Geburtsdatum;
47
48@ // --- Rolle ---
49 @EndUserText.label: 'Rolle'                         // Filterbar im Suchbereich
50 @UI.selectionField: [{ position: 20 }]              // Im Detailbereich sichtbar
51 @UI.identification: [{ position: 30 }]              // In der Tabelle sichtbar
52 @UI.lineItem: [{ position: 30 }]
53 Rolle;
54
55@ // --- Abteilung ---
56 @EndUserText.label: 'Abteilung'                    // Filterbar im List Report
57 @UI.selectionField: [{ position: 30 }]              // Im Identifikationsbereich sichtbar
58 @UI.identification: [{ position: 40 }]              // In der Tabelle angezeigt
59 @UI.lineitem: [{ position: 40 }]
60 Abteilung;
61
62@ // --- Einstellungsdatum ---
63 @EndUserText.label: 'Einstellungsdatum'            // In der Object Page sichtbar
64 @UI.identification: [{ position: 50 }]              // In der Tabellenansicht sichtbar
65 @UI.lineItem: [{ position: 50 }]
66 Einstellungsdatum;
67
68@ // --- Gehalt ---
69 @EndUserText.label: 'Gehalt'                       // Detailanzeige
70 @UI.identification: [{ position: 60 }]              // Tabellenanzeige (kann für Berechtigungen ggf. eingeschränkt werden)
71 @UI.lineItem: [{ position: 60 }]
72 Gehalt;
73
74@ // --- Kontaktinformationen ---
75 @EndUserText.label: 'Kontaktinformationen'          // In der Detailansicht sichtbar
76 @UI.identification: [{ position: 70 }]              // In der List Report Tabelle sichtbar
77 @UI.lineItem: [{ position: 70 }]
78 Kontaktinfo;
79

```

```

79
80@ // -----
81 // TECHNISCHE FELDER (werden im Fiori UI ausgeblendet)
82 // -----
83 @UI.hidden: true;                                // Benutzer, der den Datensatz angelegt hat (nicht im UI anzeigen)
84
85 @UI.hidden: true;                                // Zeitstempel der Erstellung (nicht anzeigen)
86 CreatedAt;
87
88 @UI.hidden: true;                                // Letzter Bearbeiter (nicht anzeigen)
89 LastChangedBy;
90
91 @UI.hidden: true;                                // Zeitpunkt der letzten Änderung (nicht anzeigen)
92 LastChangedAt;
93
94 @UI.hidden: true;                                // Letzte lokale Änderung (nicht anzeigen)
95 LocalLastChangedAt;
96
97 }

```

- Für **Besucher** (ZSS25\_R4C\_ME\_BESUCH\_A6) stellt sich der Inhalt der Metadata Extension wie folgt dar:

 [A07] ZSS25\_R4C\_ME\_BESUCH\_A6 X

```

1@Metadata.layer: #CUSTOMER
2 @UI: {
3   headerInfo: {
4     typeName: 'Besucher',           // Singularbezeichnung
5     typeNamePlural: 'Besucher'      // Pluralbezeichnung
6   }
7 }
8 @Search.searchable: true
9 annotate entity ZSS25_R4C_C_BESUCH_A3 with
10 {
11@ // -----
12 // FACETS (Abschnitte auf der Object Page)
13 // -----
14 @UI.facet: [
15   { position: 10,
16     label: 'Besucherinformationen',
17     type: #IDENTIFICATION_REFERENCE
18   }
19 ]
20
21 // -----
22 // FELDER MIT UI- UND SUCH-ANNOTATIONEN
23 // -----
24
25 // --- Besucher-ID ---
26 @EndUserText.label: 'Besucher-ID'
27 @UI.identification: [{ position: 10 }]
28 BesucherId;
29
30@ // --- Name ---
31 @EndUserText.label: 'Name'
32 @UI.selectionField: [{ position: 10 }]
33 @UI.identification: [{ position: 20 }]
34 @Search: { defaultSearchElement: true, fuzzinessThreshold: 0.7 }
35 @UI.lineItem: [{ position: 10 }]
36 Name;
37

```

```

38@ // --- Geburtsdatum ---
39 @EndUserText.label: 'Geburtsdatum'
40 @UI.identification: [{ position: 30 }]
41 @UI.lineItem: [{ position: 20 }]
42 Geburtsdatum;
43
44@ // --- Email ---
45 @EndUserText.label: 'E-Mail'
46 @UI.selectionField: [{ position: 20 }]
47 @UI.identification: [{ position: 40 }]
48 @UI.lineItem: [{ position: 30 }]
49 Email;
50
51@ // --- Telefon ---
52 @EndUserText.label: 'Telefon'
53 @UI.selectionField: [{ position: 30 }]
54 @UI.identification: [{ position: 50 }]
55 @UI.lineItem: [{ position: 40 }]
56 Telefon;
57
58@ // --- Adresse ---
59 @EndUserText.label: 'Adresse'
60 @UI.identification: [{ position: 60 }]
61 @UI.lineItem: [{ position: 50 }]
62 Adresse;
63
64@ // --- Registrierung ---
65 @EndUserText.label: 'Registrierung'
66 @UI.identification: [{ position: 70 }]
67 @UI.lineItem: [{ position: 60 }]
68 Registrierung;
69
70@ // -----
71 // TECHNISCHE FELDER (versteckt)
72 // -----
73 @UI.hidden: true
74 CreatedBy;
75

```

```

76 @UI.hidden: true
77 CreatedAt;
78
79 @UI.hidden: true
80 LastChangedBy;
81
82 @UI.hidden: true
83 LastChangedAt;
84
85 @UI.hidden: true
86 LocalLastChangedAt;
87 }

```

- Für **Film** (ZSS25\_R4C\_ME\_FILM\_A6) stellt sich der Inhalt der Metadata Extension wie folgt dar:

```

E [A07] ZSS25_R4C_ME_FILM_A6 X
1 @Metadata.layer: #CUSTOMER // Diese Annotation legt fest, dass diese Anpassung auf Kundenebene erfolgt.
2 // Dadurch bleibt die SAP-Standarddefinition (Layer #CORE) unverändert.
3
4 @UI: {
5   headerInfo: {
6     typeName: 'Film',
7     typeNamePlural: 'Filme' // Singularbezeichnung des Business-Objekts (z. B. im Fiori-Titel)
8   } // Pluralbezeichnung (z. B. für List Reports)
9 }
10
11 @Search.searchable: true // Aktiviert die globale Fiori-Suche für diese Entität.
12
13 annotate entity ZSS25_R4C_C_FILM_A3 with
14 {
15@ // -----
16 // FACETS (Abschnitte der Object Page in Fiori Elements)
17 // -----
18 @UI.facet: [
19   { position: 10, // Anzeige-Reihenfolge des Abschnitts
20     label: 'Filminformationen', // Titel des Abschnitts
21     type: '#IDENTIFICATION_REFERENCE' // Verknüpfung zu den Identifikationsfeldern (z. B. Stammdaten)
22   }
23 ]
24
25 // -----
26 // FELDER MIT UI- UND SUCH-ANNOTATIONEN
27 // -----
28
29 // --- Film-ID ---
30 @EndUserText.label: 'Film-ID' // Beschriftung des Feldes im UI
31 FilmId;
32
33
34

```

```

34
35@ // --- Titel ---
36 @EndUserText.label: 'Titel'
37 @UI.selectionField: [{ position: 10 }] // Erscheint im Such-/Filterbereich des List Reports
38 @UI.identification: [{ position: 20 }] // Wird im oberen Informationsbereich angezeigt
39 @Search: { defaultSearchElement: true, // Dieses Feld ist das Hauptsuchfeld
40   fuzzinessThreshold: 0.7 } // Unscharfe Suche erlaubt (z. B. bei Tippfehlern)
41 @UI.lineItem: [{ position: 20 }] // Wird in der List Report Tabelle angezeigt
42 Titel;
43
44@ // --- Genre ---
45 @EndUserText.label: 'Genre'
46 @UI.selectionField: [{ position: 20 }] // Filterbar im Suchbereich
47 @UI.identification: [{ position: 30 }] // Im Detailbereich sichtbar
48 @UI.lineItem: [{ position: 30 }] // In der Tabellenansicht sichtbar
49 Genre;
50
51@ // --- Dauer (Minuten) ---
52 @EndUserText.label: 'Dauer (Minuten)'
53 @UI.identification: [{ position: 40 }] // In der Detailansicht sichtbar
54 @UI.lineItem: [{ position: 40 }] // In der List Report Tabelle angezeigt
55 DauerMinuten;
56
57@ // --- Altersfreigabe ---
58 @EndUserText.label: 'Altersfreigabe'
59 @UI.selectionField: [{ position: 30 }] // Filterbar
60 @UI.identification: [{ position: 50 }] // Im Detailbereich sichtbar
61 @UI.lineItem: [{ position: 50 }] // In der Tabellenansicht sichtbar
62 Altersfreigabe;
63
64@ // --- Produktionsjahr ---
65 @EndUserText.label: 'Produktionsjahr'
66 @UI.identification: [{ position: 60 }] // Im Detailbereich sichtbar
67 @UI.lineItem: [{ position: 60 }] // In der Tabelle sichtbar
68 Produktionsjahr;
69

```

```

69 // --- Regisseur ---
70 @EndUserText.label: 'Regisseur'
71 @UI.identification: [{ position: 70 }]           // In der Object Page sichtbar
72 @UI.lineItem: [{ position: 70 }]                 // In der Tabellenansicht sichtbar
73 Regisseur;
74
75
76// -----
77// TECHNISCHE FELDER (nicht im UI anzeigen)
78// -----
79 @UI.hidden: true
80 CreatedBy;                                     // Benutzer, der den Datensatz angelegt hat
81
82 @UI.hidden: true
83 CreatedAt;                                    // Erstellungszeitpunkt
84
85 @UI.hidden: true
86 LastChangedBy;                                // Letzter Bearbeiter
87
88 @UI.hidden: true
89 LastChangedAt;                               // Letzter Änderungszeitpunkt
90
91 @UI.hidden: true
92 LocalLastChangedAt;                          // Letzte lokale Änderung
93 }

```

- Für **Kino** (ZSS25\_R4C\_ME\_KINO\_A6) stellt sich der Inhalt der Metadata Extension wie folgt dar:

```

E [A07] ZSS25_R4C_ME_KINO_A6 X
1 @Metadata.layer: #CUSTOMER                // Diese Annotation legt fest, dass die Anpassung auf Kundenebene erfolgt.
2                                         // Damit bleibt die SAP-Standarddefinition (Layer #CORE) unberührt.
3
4 @UI: {
5   headerInfo: {
6     typeName: 'Kino',                      // Singularbezeichnung für das Business-Objekt (z. B. im Fiori-Titel)
7     typeNamePlural: 'Kinos'                 // Pluralbezeichnung (z. B. in Listenüberschriften)
8   }
9 }
10 @Search.searchable: true                  // Aktiviert die globale Fiori-Suche für diese Entität.
11
12 annotate entity ZSS25_R4C_C_KINO_A3 with
13 {
14   // -----
15   // FACETS (Abschnitte der Object Page in Fiori)
16   // -----
17   @UI.facet: [
18     { position: 10,                         // Anzeige-Reihenfolge (kleinere Zahl = weiter oben)
19       label: 'Kinoinformationen',          // Überschrift des Abschnitts im Fiori UI
20       type: #IDENTIFICATION_REFERENCE    // Zeigt Stammdatenfelder im Bereich "Informationen" an
21     }
22   ]
23 }
24
25 // -----
26 // FELDER MIT UI- UND SUCH-ANNOTATIONEN
27 // -----
28
29 // --- Kino-ID ---
30 @EndUserText.label: 'Kino-ID'             // Beschriftung im UI (z. B. Spaltenname)
31 KinoId;
32
33@ // --- Name ---
34 @EndUserText.label: 'Name'                // Feldbezeichnung für den Benutzer
35 @UI.selectionField: [{ position: 10 }]     // Erscheint im Filterbereich (Suchleiste) des List Reports
36 @UI.identification: [{ position: 10 }]      // Wird im oberen Info-Bereich (Object Page) angezeigt
37 @Search: { defaultSearchElement: true,      // Feld wird in der globalen Suche verwendet
38            fuzzinessThreshold: 0.7 }          // Unscharfe Suche erlaubt (z. B. „Kinoo“ findet „Kino“)
39 @UI.lineItem: [{ position: 10 }]           // Feld erscheint in der List Report Tabelle
40 Name;

```

```

41 // --- Adresse ---
42@  @EndUserText.label: 'Adresse'
43  @UI.identification: [{ position: 20 }]
44  @UI.lineItem: [{ position: 20 }]          // In der Detailansicht (Object Page) sichtbar
45  @UI.lineItem: [{ position: 20 }]          // In der Tabelle angezeigt
46  Adresse;
47
48@ // --- Stadt ---
49  @EndUserText.label: 'Stadt'
50  @UI.selectionField: [{ position: 20 }]      // Filterbar im Suchbereich
51  @UI.identification: [{ position: 30 }]      // Im Detailbereich sichtbar
52  @UI.lineItem: [{ position: 30 }]          // In der Tabellenansicht sichtbar
53  Stadt;
54
55@ // --- PLZ ---
56  @EndUserText.label: 'PLZ'
57  @UI.identification: [{ position: 40 }]      // In der Object Page sichtbar
58  @UI.lineItem: [{ position: 40 }]          // In der Tabellenansicht sichtbar
59  Plz;
60
61@ // --- Anzahl Räume ---
62  @EndUserText.label: 'Anzahl Räume'
63  @UI.identification: [{ position: 50 }]      // Im Detailbereich sichtbar
64  @UI.lineItem: [{ position: 50 }]          // In der Tabellenansicht sichtbar
65  AnzahlRaeume;
66
67@ // --- Eröffnungsdatum ---
68  @EndUserText.label: 'Eröffnungsdatum'
69  @UI.identification: [{ position: 60 }]      // In der Object Page sichtbar
70  @UI.lineItem: [{ position: 60 }]          // In der Tabellenansicht sichtbar
71  Eroeffnungsdatum;
72
73@ // --- Kontaktinformationen ---
74  @EndUserText.label: 'Kontaktinformationen'
75  @UI.identification: [{ position: 70 }]      // In der Detailansicht sichtbar
76  @UI.lineItem: [{ position: 70 }]          // In der List Report Tabelle sichtbar
77  Kontaktinfo;
78

```

```

78
79@ // -----
80 // TECHNISCHE FELDER (im Fiori UI ausgeblendet)
81 // -----
82  @UI.hidden: true                         // Benutzer, der den Datensatz erstellt hat
83  CreatedBy;
84
85  @UI.hidden: true                         // Zeitstempel der Erstellung
86  CreatedAt;
87
88  @UI.hidden: true                         // Letzter Bearbeiter
89  LastChangedBy;
90
91  @UI.hidden: true                         // Zeitpunkt der letzten Änderung
92  LastChangedAt;
93
94  @UI.hidden: true                         // Letzte lokale Änderung (z. B. für Replikationen)
95  LocalLastChangedAt;
96 }

```

- Für **Raum** (ZSS25\_R4C\_ME\_RAUM\_A6) stellt sich der Inhalt der Metadata Extension wie folgt dar:

```

E [A07] ZSS25_R4C_ME_RAUM_A6 X
1 @Metadata.layer: #CUSTOMER // Definiert, dass diese Annotation auf der Kundenebene liegt.
2 // Dadurch bleibt die SAP-Standarddefinition (Layer #CORE) unberührt.
3
4 @UI: {
5   headerInfo: {
6     typeName: 'Raum', // Singular-Bezeichnung für das Objekt (z. B. "Raum anzeigen")
7     typeNamePlural: 'Räume' // Plural-Bezeichnung für Überschriften oder Listen ("Räume anzeigen")
8   }
9 }
10
11 @Search.searchable: true // Aktiviert die globale Fiori-Suche für diese Entität.
12
13 annotate entity ZSS25_R4C_C_RAUM_A3 with
14 {
15@ // -----
16 // FACETS (Abschnitte / Tabs auf der Fiori Object Page)
17 // -----
18 @UI.facet: [
19   { position: 10, // Reihenfolge auf der Seite (niedrige Zahl = weiter oben)
20     label: 'Rauminformationen', // Überschrift des Abschnitts
21     type: '#IDENTIFICATION_REFERENCE' // Verknüpft mit den Identifikationsfeldern (Detailinformationen)
22   }
23 ]
24
25 // -----
26 // FELDER MIT UI- UND SUCH-ANNOTATIONEN
27 // -----
28
29 // --- Raum-ID ---
30 @EndUserText.label: 'Raum-ID' // Benutzerfreundliche Beschriftung im UI
31 RaumId;
32
33@ // --- Kino-ID ---
34 @EndUserText.label: 'Kino-ID' // Fremdschlüssel zum zugehörigen Kino
35 @UI.selectionField: [{ position: 10 }] // Wird im Such-/Filterbereich angezeigt
36 @UI.identification: [{ position: 10 }] // Wird im oberen Info-Bereich (Object Page) angezeigt
37 @UI.lineItem: [{ position: 10 }] // Wird als Spalte im List Report angezeigt
38 KinoId;
39

```

```

40@ // --- Name ---
41 @EndUserText.label: 'Name' // Bezeichnung des Raums (z. B. "Saal 1", "VIP-Lounge")
42 @UI.selectionField: [{ position: 20 }] // Im Filterbereich sichtbar
43 @UI.identification: [{ position: 20 }] // In der Detailansicht (Object Page) sichtbar
44 @Search: { defaultSearchElement: true, // Standard-Suchfeld für die Fiori-Suche
45   fuzzinessThreshold: 0.7 } // Unscharfe Suche erlaubt (z. B. Tippfehler)
46 @UI.lineItem: [{ position: 20 }] // In der Tabellenansicht sichtbar
47 Name;
48
49@ // --- Stühle pro Reihe ---
50 @EndUserText.label: 'Stühle pro Reihe' // Anzahl Sitze je Reihe
51 @UI.identification: [{ position: 30 }] // In der Object Page sichtbar
52 @UI.lineItem: [{ position: 30 }] // In der Tabelle sichtbar
53 StuhelAnzahlProReihe;
54
55@ // --- Sitzplätze ---
56 @EndUserText.label: 'Sitzplätze' // Gesamtanzahl der Sitze im Raum
57 @UI.identification: [{ position: 40 }] // Im Detailbereich sichtbar
58 @UI.lineItem: [{ position: 40 }] // In der Tabelle sichtbar
59 Sitzplaetze;
60
61@ // --- Ausstattung ---
62 @EndUserText.label: 'Ausstattung' // Beschreibung technischer Ausstattung (z. B. "3D", "Dolby Atmos")
63 @UI.identification: [{ position: 50 }] // In der Detailansicht sichtbar
64 @UI.lineItem: [{ position: 50 }] // In der Tabellenansicht sichtbar
65 Ausstattung;
66

```

```

67@ // -----
68 // TECHNISCHE FELDER (nicht im UI anzeigen)
69 // -----
70 @UI.hidden: true
71 CreatedBy; // Benutzer, der den Datensatz erstellt hat
72
73 @UI.hidden: true
74 CreatedAt; // Zeitstempel der Erstellung
75
76 @UI.hidden: true
77 LastChangedBy; // Letzter Bearbeiter
78
79 @UI.hidden: true
80 LastChangedAt; // Zeitpunkt der letzten Änderung
81
82 @UI.hidden: true
83 LocalLastChangedAt; // Letzte lokale Änderung (für Replikation / Synchronisation)
84 }

```

- Für **Ticket** (ZSS25\_R4C\_ME\_TICKET\_A6) stellt sich der Inhalt der Metadata Extension wie folgt dar:

```

E [A07] ZSS25_R4C_ME_TICKET_A6 X
1 @Metadata.layer: #CUSTOMER // Legt fest, dass diese Annotationsdefinition auf der Kundenschicht liegt.
2 // So bleiben SAP-Standard-Annotationen (Layer #CORE) unverändert.
3
4 @UI: {
5   headerInfo: {
6     typeName: 'Ticket',
7     typeNamePlural: 'Tickets'
8   }
9 }
10 @Search.searchable: true
11 annotate entity ZSS25_R4C_C_TICKET_A3 with
12 {
13@ // -----
14 // FACETS: Abschnitte/Tabs auf der Object Page
15 // -----
16 @UI.facet: [
17   { position: 10,
18     label: 'Ticketinformationen',
19     type: '#IDENTIFICATION_REFERENCE'
20   }
21 ]
22
23 // -----
24 // FELDER UND IHRE ANNOTATIONEN
25 // -----
26
27 // --- Ticket-ID ---
28 @EndUserText.label: 'Ticket-ID' // Beschriftung im UI
29 @Search: { defaultSearchElement: true, // Standard-Suchfeld für die Fiori-Suche
30           fuzzinessThreshold: 0.7 } // Unscharfe Suche erlaubt (z. B. Tippfehler)
31
32 TicketId;
33
34@ // --- Vorstellung-ID ---
35 @EndUserText.label: 'Vorstellung-ID' // Referenz auf die Filmvorstellung
36 @UI.selectionField: [{ position: 10 }] // Wird im Filterbereich des List Reports angezeigt
37 @UI.identification: [{ position: 10 }] // Erscheint auf der Object Page
38 @UI.lineItem: [{ position: 10 }] // Spalte im List Report
39 VorstellungId;
40

```

```

40
41@ // --- Besucher-ID ---
42 @EndUserText.label: 'Besucher-ID' // Referenz auf den Besucher, der das Ticket gekauft hat
43 @UI.selectionField: [{ position: 20 }] // Filtermöglichkeit
44 @UI.identification: [{ position: 20 }] // Anzeige im Detailbereich
45 @UI.lineItem: [{ position: 20 }] // Spalte in der Ticketliste
46 BesucherId;
47
48@ // --- Sitzplatznummer ---
49 @EndUserText.label: 'Sitzplatznummer' // Sitznummer des Tickets (z. B. "Reihe 5, Platz 8")
50 @UI.selectionField: [{ position: 30 }] // Kann im Filterbereich gesucht werden
51 @UI.identification: [{ position: 30 }] // Anzeige auf der Detailseite
52 @UI.lineItem: [{ position: 30 }] // Anzeige im List Report
53 Sitzplatznummer;
54
55@ // --- Preis ---
56 @EndUserText.label: 'Preis' // Ticketpreis
57 @UI.identification: [{ position: 40 }] // Anzeige auf der Object Page
58 @UI.lineItem: [{ position: 40 }] // Spalte in der Tabelle
59 Preis;
60
61
62
63@ // --- Tickettyp ---
64 @EndUserText.label: 'Tickettyp' // Art des Tickets (z. B. "Standard", "VIP", "Kinder")
65 @UI.identification: [{ position: 50 }] // Detailanzeige
66 @UI.lineItem: [{ position: 50 }] // Spalte im List Report
67 TicketTyp;
68
69@ // --- QR-Code ---
70 @EndUserText.label: 'QR-Code' // Eindeutiger QR-Code für den Eintritt
71 @UI.identification: [{ position: 60 }] // Anzeige im Detail (z. B. für Kontrolle)
72 @UI.lineItem: [{ position: 60 }] // Optional im List Report
73 QrCode;
74

```

```

74 // --- Status ---
75 @EndUserText.label: 'Status'                                // Z. B. "bezahlt", "storniert", "verwendet"
76 @UI.selectionField: [{ position: 40 }]                      // Filtermöglichkeit im List Report
77 @UI.identification: [{ position: 70 }]                      // Anzeige auf der Object Page
78 @UI.lineItem: [{ position: 70 }]                            // Spalte im List Report
79
80 Status;
81
82 // -----
83 // TECHNISCHE FELDER – NICHT IM UI SICHTBAR
84 // -----
85 @UI.hidden: true                                         // Benutzer, der das Ticket erstellt hat
86 CreatedBy;
87
88 @UI.hidden: true                                         // Zeitpunkt der Erstellung
89 CreatedAt;
90
91 @UI.hidden: true                                         // Benutzer der letzten Änderung
92 LastChangedBy;
93
94 @UI.hidden: true                                         // Zeitpunkt der letzten Änderung
95 LastChangedAt;
96
97 @UI.hidden: true                                         // Letzte lokale Änderung (für Replikation / Offline-Sync)
98 LocalLastChangedAt;
99 }

```

- Für **Verkauf** (ZSS25\_R4C\_ME\_VERKF\_A6) stellt sich der Inhalt der Metadata Extension wie folgt dar:

```

[A07] ZSS25_R4C_ME_VERKF_A6 X
1@Metadata.layer: #CUSTOMERdata Extension ZSS25_R4C_ME_VV/Kunden-Layer überschreibt keine SAP-Standarddefinition
2
3 @UI: {
4   headerInfo: {
5     typeName: 'Verkauf',                                     // Singular (z. B. "Verkauf anzeigen")
6     typeNamePlural: 'Verkäufe'                                // Plural (z. B. "Verkäufe anzeigen")
7   }
8 }
9
10 @Search.searchable: true                                    // Globale Fiori-Suche aktivieren
11
12 annotate entity ZSS25_R4C_C_VERKF_A3 with
13 {
14 // -----
15 // FACETS (Tabs / Abschnitte auf der Fiori Object Page)
16 // -----
17 @UI.facet: [
18   { position: 10,
19     label: 'Verkaufsinformationen',
20     type: #IDENTIFICATION_REFERENCE
21   }
22 ]
23
24 // -----
25 // FELDER MIT UI- UND SUCH-ANNOTATIONEN
26 // -----
27
28 // --- Verkauf-ID ---
29 @EndUserText.label: 'Verkauf-ID'
30 VerkaufId;
31
32 // --- Ticket-ID ---
33 @EndUserText.label: 'Ticket-ID'
34 @UI.selectionField: [{ position: 10 }]
35 @UI.identification: [{ position: 10 }]
36 @UI.lineItem: [{ position: 10 }]
37 TicketId;
38

```

```

38
39@ // --- Mitarbeiter-ID ---
40 @EndUserText.label: 'Mitarbeiter-ID'
41 @UI.selectionField: [{ position: 20 }]
42 @UI.identification: [{ position: 20 }]
43 @UI.lineItem: [{ position: 20 }]
44 MitarbeiterId;
45
46@ // --- Mitarbeiter-Name ---
47 @EndUserText.label: 'Mitarbeitername'
48 @UI.identification: [{ position: 30 }]
49 @UI.lineItem: [{ position: 30 }]
50 @Search: { defaultSearchElement: true, fuzzinessThreshold: 0.7 }
51 MitarbeiterName;
52
53@ // --- Besuchername ---
54 @EndUserText.label: 'Besuchername'
55 @UI.identification: [{ position: 40 }]
56 @UI.lineItem: [{ position: 40 }]
57 BesucherName;
58
59@ // --- Verkaufsdatum ---
60 @EndUserText.label: 'Verkaufsdatum'
61 @UI.selectionField: [{ position: 30 }]
62 @UI.identification: [{ position: 50 }]
63 @UI.lineItem: [{ position: 50 }]
64 Verkaufsdatum;
65
66@ // --- Zahlungsart ---
67 @EndUserText.label: 'Zahlungsart'
68 @UI.selectionField: [{ position: 40 }]
69 @UI.identification: [{ position: 60 }]
70 @UI.lineItem: [{ position: 60 }]
71 Zahlungsart;
72

```

```
72
73@ // --- Ticketpreis ---
74 @EndUserText.label: 'Ticketpreis'
75 @UI.identification: [{ position: 70 }]
76 @UI.lineItem: [{ position: 70 }]
77 TicketPreis;
78
79@ // --- Rabatt ---
80 @EndUserText.label: 'Rabatt'
81 @UI.identification: [{ position: 80 }]
82 @UI.lineItem: [{ position: 80 }]
83 Rabatt;
84
85@ // --- Betrag ---
86 @EndUserText.label: 'Gesamtbetrag'
87 @UI.identification: [{ position: 90 }]
88 @UI.lineItem: [{ position: 90 }]
89 Betrag;
90
91@ // --- Besucher-Altersgruppe ---
92 @EndUserText.label: 'Besucher-Altersgruppe'
93 @UI.identification: [{ position: 100 }]
94 @UI.lineItem: [{ position: 100 }]
95 BesucherAltersgruppe;
96
97@ // --- Besucher ist erlaubt ---
98 @EndUserText.label: 'Besucher ist erlaubt'
99 @UI.identification: [{ position: 110 }]
100 @UI.lineItem: [{ position: 110 }]
101 BesucherIstErlaubt;
102
```

```

102
103@ // --- Sitzplatznummer ---
104 @EndUserText.label: 'Sitzplatznummer'
105 @UI.identification: [{ position: 120 }]
106 @UI.lineItem: [{ position: 120 }]
107 TicketSitzplatznummer;
108
109@ // --- Sitzreihe ---
110 @EndUserText.label: 'Sitzreihe'
111 @UI.identification: [{ position: 130 }]
112 @UI.lineItem: [{ position: 130 }]
113 BesucherSitzReihe;
114
115@ // --- Ticket gültig ---
116 @EndUserText.label: 'Ticket ist gültig'
117 @UI.identification: [{ position: 140 }]
118 @UI.lineItem: [{ position: 140 }]
119 TicketIstGueltig;
120
121@ // --- Zeitkategorie ---
122 @EndUserText.label: 'Zeitkategorie'
123 @UI.identification: [{ position: 150 }]
124 @UI.lineItem: [{ position: 150 }]
125 Zeitkategorie;
126
127@ // --- Ticketstatus ---
128 @EndUserText.label: 'Ticketstatus'
129 @UI.identification: [{ position: 160 }]
130 @UI.lineItem: [{ position: 160 }]
131 TicketStatus;
132

```

```

132
133@ // -----
134 // TECHNISCHE FELDER (nicht im UI anzeigen)
135 // -----
136 @UI.hidden: true
137 CreatedBy;
138
139 @UI.hidden: true
140 CreatedAt;
141
142 @UI.hidden: true
143 LastChangedBy;
144
145 @UI.hidden: true
146 LastChangedAt;
147
148 @UI.hidden: true
149 LocalLastChangedAt;
150
151@ // -----
152 // ASSOCIATIONS (nicht im UI anzeigen)
153 // -----
154 @UI.hidden: true
155 _Mitarbeiter;
156
157 @UI.hidden: true
158 _Ticket;
159

```

- Für **Vorstellung** (ZSS25\_R4C\_ME\_VORST\_A6) stellt sich der Inhalt der Metadata Extension wie folgt dar:

```

E [A07] ZSS25_R4C_ME_VORST_A6 ×
1 @Metadata.layer: #CUSTOMER // Definiert, dass diese Annotation auf der Kundenschicht liegt.
2 // Damit überschreibst du nichts aus dem SAP-Standard (Layer #CORE).
3
4 @UI: {
5   headerInfo: {
6     typeName: 'Vorstellung', // Singularname - wird z. B. im Titel der Detailansicht angezeigt
7     typeNamePlural: 'Vorstellungen' // Pluralname - wird z. B. in Tabellenüberschriften verwendet
8   }
9 }
10 @Search.searchable: true
11 annotate entity ZSS25_R4C_C_VORST_A3 with
12 {
13@ // -----
14 // FACETS: Gliederung der Object Page (Abschnitte)
15 // -----
16 @UI.facet: [
17   { position: 10, // Reihenfolge des Abschnitts auf der Seite
18     label: 'Vorstellungsinformationen', // Titel des Abschnitts
19     type: #IDENTIFICATION_REFERENCE // Zeigt Felder aus @UI.identification in diesem Abschnitt
20   }
21 ]
22
23 // -----
24 // FELDER MIT UI- UND SUCH-ANNOTATIONEN
25 // -----
26
27 // --- Vorstellung-ID ---
28 @EndUserText.label: 'Vorstellung-ID' // Eindeutige ID der Vorstellung
29 VorstellungId;
30
31@ // --- Raum-ID ---
32 @EndUserText.label: 'Raum-ID' // Verknüpfung zum Kinosaal, in dem der Film gezeigt wird
33 @UI.selectionField: [{ position: 10 }] // Wird im Filterbereich angezeigt
34 @UI.identification: [{ position: 10 }] // Auf der Object Page sichtbar
35 @UI.lineItem: [{ position: 10 }] // Spalte im List Report
36 @Search: { defaultSearchElement: true, // Standard-Suchfeld für die Fiori-Suche
37   fuzzinessThreshold: 0.7 } // Unscharfe Suche erlaubt (z. B. Tippfehler)
38 RaumId;
39
40@ // --- Film-ID ---
41 @EndUserText.label: 'Film-ID' // Verknüpfung zum gezeigten Film
42 @UI.selectionField: [{ position: 20 }] // Filterbar (z. B. nach bestimmten Filmen)
43 @UI.identification: [{ position: 20 }] // Anzeige im Detail
44 @UI.lineItem: [{ position: 20 }] // Spalte im List Report
45 FilmId;
46
47@ // --- Sprache ---
48 @EndUserText.label: 'Sprache' // Sprache der Vorstellung (z. B. "Deutsch", "Englisch")
49 @UI.selectionField: [{ position: 30 }] // Filterbar (z. B. "alle englischen Vorstellungen")
50 @UI.identification: [{ position: 30 }] // Anzeige im Detailbereich
51 @UI.lineItem: [{ position: 30 }] // Spalte im List Report
52 Sprache;
53
54@ // --- Startzeit ---
55 @EndUserText.label: 'Startzeit' // Beginn der Vorstellung
56 @UI.selectionField: [{ position: 40 }] // Filterbar nach Startzeit
57 @UI.identification: [{ position: 40 }] // Anzeige auf der Object Page
58 @UI.lineItem: [{ position: 40 }] // Anzeige im List Report
59 Startzeit;
60
61@ // --- Endzeit ---
62 @EndUserText.label: 'Endzeit' // Ende der Vorstellung
63 @UI.identification: [{ position: 50 }] // Detailanzeige
64 @UI.lineItem: [{ position: 50 }] // Spalte in der Liste
65 Endzeit;
66
67@ // --- Untertitel ---
68 @EndUserText.label: 'Untertitel' // Gibt an, ob Untertitel vorhanden sind
69 @UI.identification: [{ position: 60 }] // Anzeige im Detail
70 @UI.lineItem: [{ position: 60 }] // Spalte in der Tabelle
71 Untertitel;
72

```

```

72 // --- Vorstellungformat ---
73 @EndUserText.label: 'Vorstellungsformat'                                // Format der Vorstellung (z. B. 2D, 3D, IMAX)
74 @UI.identification: [{ position: 70 }]                                     // Anzeige im Detailbereich
75 @UI.lineItem: [{ position: 70 }]                                         // Spalte im List Report
76 VorstellungFormat;
77
78
79 // --- Status ---
80 @EndUserText.label: 'Status'                                              // Gibt an, ob die Vorstellung aktiv, abgesagt etc. ist
81 @UI.selectionField: [{ position: 50 }]                                     // Filterbar (z. B. "nur aktive Vorstellungen")
82 @UI.identification: [{ position: 80 }]                                     // Anzeige auf der Object Page
83 @UI.lineItem: [{ position: 80 }]                                         // Spalte in der Tabelle
84 Status;
85
86 // --- Datum ---
87 @EndUserText.label: 'Datum'                                                 // Datum der Vorstellung
88 @UI.selectionField: [{ position: 40 }]                                     // Filterbar nach Datum
89 @UI.identification: [{ position: 90 }]                                     // Anzeige auf der Object Page
90 @UI.lineItem: [{ position: 90 }]                                         // Anzeige im List Report
91 Datum;
92
93 // -----
94 // TECHNISCHE FELDER (nicht im UI anzeigen)
95 // -----
96 @UI.hidden: true                                                        // Benutzer, der den Datensatz erstellt hat
97 CreatedBy;                                                              
98
99 @UI.hidden: true                                                        // Zeitpunkt der Erstellung
100 CreatedAt;                                                              
101
102 @UI.hidden: true                                                        // Benutzer der letzten Änderung
103 LastChangedBy;                                                              
104
105 @UI.hidden: true                                                        // Zeitpunkt der letzten Änderung
106 LastChangedAt;                                                              
107
108 @UI.hidden: true                                                        // Zeitpunkt der letzten lokalen Änderung (z. B. bei Replikation)
109 LocalLastChangedAt;
110 }

```

## Notiz:

- Nachdem die Metadata Extension erstellt wurde, muss sie mit Strg + S gespeichert und anschließend über das Aktivierungssymbol in SAP aktiviert werden.
- Nach erfolgreicher Aktivierung ist die Metadata Extension im Projektbaum unter dem Pfad Core Data Services → Metadata Extensions zu finden.

## Ergebnisse:

- Das Service Binding **Mitarbeiter** liefert nach der Metadata Extension (ZSS25\_R4C\_ME\_MITARB\_A6) das folgende Ergebnis:

Mitarbeiter (7)							Create	Delete
	Name	Geburtsdatum	Rolle	Abteilung	Einstellungsdatum	Gehalt	Kontaktinformationen	
<input type="checkbox"/>	Kamal Badawi	Jan 15, 1993	Kassierer	VKF1	Oct 1, 2023	2,575.00	Seifgrundstr. 4, 61348 Bad Homburg	>
<input type="checkbox"/>	Clara Schäfer	Sep 5, 1998	SchichtleiterIn	VKF1	Jan 3, 2016	3,375.00	Louisenstraße 32, 61348 Bad Homburg	>
<input type="checkbox"/>	Sophie Braun	Sep 5, 1995	KassiererIn	VKF1	Jan 18, 2016	2,800.00	Kaiser-Friedrich-Promenade 85, 61348 Bad Homburg	>
<input type="checkbox"/>	Lukas Schneider	May 6, 2000	KassiererIn	VKF2	Jan 1, 2019	1,950.00	Saaburgstraße 12, 61350 Bad Homburg	>
<input type="checkbox"/>	Maximilian Hoffmann	May 6, 2001	SchichtleiterIn	VKF2	Jan 1, 2019	3,150.00	Friedrichstraße 7, 61348 Bad Homburg	>
<input type="checkbox"/>	Anna Müller	May 6, 2005	KassiererIn	VKF2	Jan 1, 2020	2,500.00	Gartenfeldstraße 22, 61348 Bad Homburg	>
<input type="checkbox"/>	Lisa Müller	Oct 30, 1997	KassiererIn	VKF2	Jan 1, 2022	1,980.00	Seifgrundstr. 1, 61348 Bad Homburg	>

- Das Service Binding **Besucher** liefert nach der Metadata Extension (ZSS25\_R4C\_ME\_BESUCH\_A6) das folgende Ergebnis:

Name:	E-Mail:	Telefon:				
<input type="text"/>	<input type="text"/>	<input type="text"/>				
<b>Besucher (7)</b>						
Name	Geburtsdatum	E-Mail	Telefon	Adresse	Registrierung	
ANNA MÜLLER	Mar 14, 1996	ANNA.MUELLER@DUMMYMAIL.DE	+49 151 834561	HAUPTSTR. 12, 70173 STUTTGART	Sep 1, 2023	>
TOBIAS SCHMIDT	Jul 22, 2010	T.SCHMIDT@DUMMYMAIL.DE	+49 176 998877	LINDENWEG 5, 50667 KÖLN	Jul 25, 2012	>
LINH NGUYEN	Nov 3, 2006	LINH.NGUYEN@DUMMYMAIL.DE	+49 172 556672	BERLINER ALLEE 45, 40212 DÜSSELDORF	Feb 7, 2019	>
JANA BECKER	Jan 28, 2014	JANA.BECKER@DUMMYMAIL.DE	+49 163 628894	GOETHESTR. 20, 04109 LEIPZIG	Jan 15, 2023	>
OMAR ALI	Nov 17, 1997	OMAR.ALI@DUMMYMAIL.DE	+49 152 543255	RHEINSTRÄÙ 33, 60329 FRANKFURT	Sep 1, 2023	>
SOPHIE WAGNER	Dec 17, 2002	SOPHIE.WAGNER@DUMMYMAIL.DE	+49 151 712138	SCHILLERPLATZ 7, 01067 DRESDEN	Oct 2, 2023	>
MARCO ROSSI	May 9, 1998	MARCO.ROSSI@DUMMYMAIL.DE	+49 162 334445	MOZARTSTR. 8, 80331 MÜNCHEN	Sep 10, 2012	>

- Das Service Binding **Film** liefert nach der Metadata Extension (ZSS25\_R4C\_ME\_FILM\_A6) das folgende Ergebnis:

Title:	Genre:	Altersfreigabe:				
<input type="text"/>	<input type="text"/>	<input type="text"/>				
<b>Filme (8)</b>						
Titel	Genre	Dauer (Minuten)	Altersfreigabe	Produktion...	Regisseur	
INCEPTION	SCI-FI	180	12	2016	RAFAEL MARTINEZ	>
DIE EISKÖNIGIN	ANIMATION	95	18	2024	JENNIFER LEE	>
THE DARK KNIGHT	ACTION	152	18	2008	LORENZO MILLA	>
PARASITE	THRILLER	150	16	2025	BONG JOON-HO	>
LA LA LAND	MUSICAL	128	0	2025	DAMIEN CHAZELLE	>
AVATAR	SCI-FI	162	12	2009	JAMES CAMERON	>
COCO	ANIMATION	105	18	2017	LEE UNKRICH	>
INTERSTELLER	SCI-FI	65	6	2014	ULRICH KOCH	>

- Das Service Binding **Kino** liefert nach der Metadata Extension (ZSS25\_R4C\_ME\_KINO\_A6) das folgende Ergebnis:

Name:	Stadt:	Editing Status:	Go	Add			
<input type="text"/> All	<input type="text"/>	<input type="button" value="All"/>		<input type="button" value="Enter"/>			
<b>Kinos (5)</b>			Create	Delete			
Name	Adresse	Stadt	PLZ	Anzahl Räume	Eroffnungsdatum	Kontaktinformationen	
<input type="checkbox"/> RAP4YOU	POTSDAMER PLATZ 1	BAD HOMBURG	10785	5	Dec 1, 2001	01514 1223334	>
<input type="checkbox"/> UCI DX	AM WEHRHAHN 45	FRIEDBERG	40211	4	Mar 30, 2003	01710 8846622	>
<input type="checkbox"/> MATHÄSER STAR	BAYERSTRÄÙ 3	MARBURG	80335	6	Dec 8, 2004	01725 8636633	>
<input type="checkbox"/> CINEMINNAXX	DAMMTORSTRÄÙ 7	LIMBURG	20354	5	Sep 15, 2006	01600 4446644	>
<input type="checkbox"/> FILM ONE	HOHENZOLLERNRING 35	FRANKFURT	50672	3	May 24, 2010	01500 2223355	>

- Das Service Binding **Raum** liefert nach der Metadata Extension (ZSS25\_R4C\_ME\_RAUM\_A6) das folgende Ergebnis:

Search		Editing Status:	Kino-ID:	Name:
		All		
<b>Räume (9)</b>				
	Kino-ID	Name	Stühle pro Reihe	Sitzplätze Ausstattung
<input type="checkbox"/>	4001	SAAL 1	10	120 3D, DOLBY SURROUND >
<input type="checkbox"/>	4001	SAAL 2	8	80 2D, DOLBY SURROUND >
<input type="checkbox"/>	4002	HAUPTSAAL	13	150 3D, DOLBY ATMOS >
<input type="checkbox"/>	4003	SAAL A	10	100 2D, STANDARD >
<input type="checkbox"/>	4003	SAAL B	6	60 2D, STANDARD >
<input type="checkbox"/>	4004	SAAL 1	12	120 3D, DOBY SURROUND >
<input type="checkbox"/>	4004	SAAL 2	9	90 2D, STANDARD >
<input type="checkbox"/>	4005	GROßER SAAL	15	200 3D, DOLBY ATMOS, IMAX >
<input type="checkbox"/>	4005	KLEINER SAAL	7	70 2D, STANDARD >

- Das Service Binding **Ticket** liefert nach der Metadata Extension (ZSS25\_R4C\_ME\_TICKET\_A6) das folgende Ergebnis:

Search		Editing Status:	Vorstellung-ID:	Besucher-ID:	Sitzplatznummer:	Status:
		All				
<b>Tickets (23)</b>						
	Vorstellung-ID	Besucher-ID	Sitzplatznummer	Preis	Tickettyp	QR-Code
<input type="checkbox"/>	7001	1001	13	15.00	KIND	HTTPS://GITHUB.COM/KAMAL-BADAWI?TAB=REPOSITORIES
<input type="checkbox"/>	7001	1002	32	20.00	STANDARD	HTTPS://GITHUB.COM/KAMAL-BADAWI?TAB=REPOSITORIES
<input type="checkbox"/>	7002	1003	34	18.00	STANDARD	HTTPS://GITHUB.COM/KAMAL-BADAWI?TAB=REPOSITORIES
<input type="checkbox"/>	7003	1004	2	22.00	STANDARD	HTTPS://GITHUB.COM/KAMAL-BADAWI?TAB=REPOSITORIES
<input type="checkbox"/>	7004	1005	2	35.00	VIP	HTTPS://GITHUB.COM/KAMAL-BADAWI?TAB=REPOSITORIES
<input type="checkbox"/>	7005	1006	1	35.00	VIP	HTTPS://GITHUB.COM/KAMAL-BADAWI?TAB=REPOSITORIES
<input type="checkbox"/>	7006	1007	5	15.00	KIND	HTTPS://GITHUB.COM/KAMAL-BADAWI?TAB=REPOSITORIES
<input type="checkbox"/>	7003	1001	1	45.00	VIP	HTTPS://GITHUB.COM/KAMAL-BADAWI?TAB=REPOSITORIES
<input type="checkbox"/>	7003	1001	3	40.00	VIP	HTTPS://GITHUB.COM/KAMAL-BADAWI?TAB=REPOSITORIES
<input type="checkbox"/>	7001	1002	35	15.00	STANDARD	HTTPS://GITHUB.COM/KAMAL-BADAWI?TAB=REPOSITORIES
<input type="checkbox"/>	7002	1003	22	20.00	STANDARD	HTTPS://GITHUB.COM/KAMAL-BADAWI?TAB=REPOSITORIES
<input type="checkbox"/>	7003	1004	40	8.00	KIND	HTTPS://GITHUB.COM/KAMAL-BADAWI?TAB=REPOSITORIES
<input type="checkbox"/>	7001	1001	20	20.00	STANDARD	HTTPS://GITHUB.COM/KAMAL-BADAWI?TAB=REPOSITORIES

- Das Service Binding **Verkauf** liefert nach der Metadata Extension (ZSS25\_R4C\_ME\_VERKF\_A6) das folgende Ergebnis:

Search		Editing Status:	All	Verkaufsdatum:	Zahlungsart:							
Verkäufe (23)												
	Verkaufsdatum	Zahlungsart	Ticketpreis	Rabatt	Gesamtbetrag	Besucher-Altersgr...	Besucher ist erl...	Sitzplatznummer	Sitzreihe	Ticket ist gültig	Zeitkategorie	Ticketstatus
<input type="checkbox"/>	Oct 11, 2025	BAR	15.00	0.00	15.00	Erwachsen	Erlaubt	13	2	Ungültig	Zukunft	BEZAHLT
<input type="checkbox"/>	Oct 15, 2025	BAR	20.00	25.00	15.00	Senior	Erlaubt	32	4	Gültig	Zukunft	BEZAHLT
<input type="checkbox"/>	Oct 12, 2025	KARTE	18.00	10.00	16.20	Erwachsen	Erlaubt	34	4	Gültig	Zukunft	BEZAHLT
<input type="checkbox"/>	Oct 1, 2025	KARTE	22.00	0.00	22.00	Kinds	Verboten	2	1	Gültig	Zukunft	BEZAHLT
<input type="checkbox"/>	Oct 14, 2025	BAR	35.00	0.00	35.00	Erwachsen	Erlaubt	2	1	Gültig	Zukunft	BEZAHLT
<input type="checkbox"/>	Oct 12, 2025	BAR	35.00	5.00	33.25	Erwachsen	Erlaubt	1	1	Gültig	Heute	BEZAHLT
<input type="checkbox"/>	Oct 27, 2025	BAR	15.00	0.00	15.00	Erwachsen	Erlaubt	5	1	Ungültig	Zukunft	BEZAHLT
<input type="checkbox"/>	Oct 15, 2025	BAR	45.00	10.00	40.50	Erwachsen	Erlaubt	1	1	Gültig	Zukunft	BEZAHLT
<input type="checkbox"/>	Oct 30, 2025	KARTE	40.00	5.00	38.00	Erwachsen	Erlaubt	3	1	Gültig	Zukunft	STORNIERT
<input type="checkbox"/>	Oct 2, 2025	KARTE	15.00	0.00	15.00	Senior	Erlaubt	35	4	Gültig	Zukunft	BEZAHLT
<input type="checkbox"/>	Oct 22, 2025	BAR	20.00	0.00	20.00	Erwachsen	Erlaubt	22	3	Gültig	Zukunft	BEZAHLT
<input type="checkbox"/>	Oct 30, 2025	BAR	8.00	0.00	8.00	Kinds	Verboten	40	5	Gültig	Zukunft	BEZAHLT
<input type="checkbox"/>	Oct 1, 2025	KARTE	20.00	15.00	17.00	Erwachsen	Erlaubt	20	2	Gültig	Zukunft	BEZAHLT

- Das Service Binding **Vorstellung** liefert nach der Metadata Extension (ZSS25\_R4C\_ME\_VORST\_A6) das folgende Ergebnis:

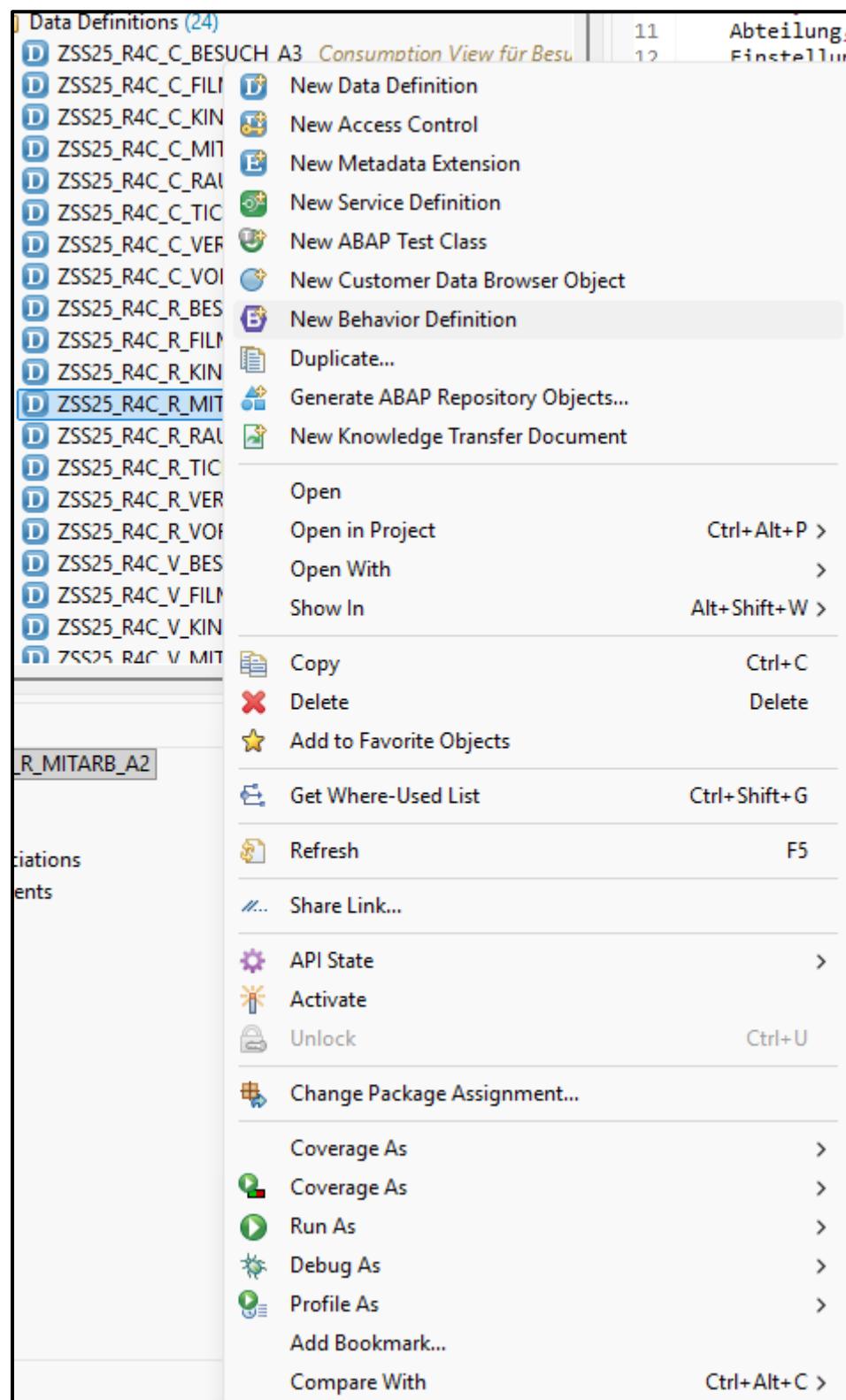
Search		Editing Status:	Raum-ID:	Film-ID:	Sprache:	Startzeit:	Datum:	Status:	Go Adapt Filters (1)		
Vorstellungen (8)											
	Raum-ID	Film-ID	Sprache	Startzeit	Endzeit	Untertitel	Vorstellungformat	Status	Datum		
<input type="checkbox"/>	6001	3001	DE	2:00:00 PM	6:00:00 PM	VORHANDEN	3D	ABGESAGT	Nov 9, 2025	>	
<input type="checkbox"/>	6001	3008	DE	5:00:00 PM	6:30:00 PM	VORHANDEN	IMAX	AKTIV	Oct 23, 2025	>	
<input type="checkbox"/>	6002	3002	DE	3:00:00 PM	5:00:38 PM	N. V.	3D	AKTIV	Nov 2, 2025	>	
<input type="checkbox"/>	6003	3003	DE	6:00:00 PM	9:00:00 PM	N. V.	IMAX	AKTIV	Oct 15, 2025	>	
<input type="checkbox"/>	6004	3004	EN	6:00:00 PM	9:00:00 PM	VORHANDEN	IMAX	AKTIV	Oct 12, 2025	>	
<input type="checkbox"/>	6005	3005	DE	6:00:00 PM	8:30:00 PM	VORHANDEN	2D	AKTIV	Oct 18, 2025	>	
<input type="checkbox"/>	6006	3006	EN	5:00:00 PM	9:30:00 PM	VORHANDEN	2D	AKTIV	Nov 7, 2025	>	
<input type="checkbox"/>	6008	3007	DE	4:00:00 PM	5:30:00 PM	VORHANDEN	3D	AKTIV	Oct 25, 2025	>	

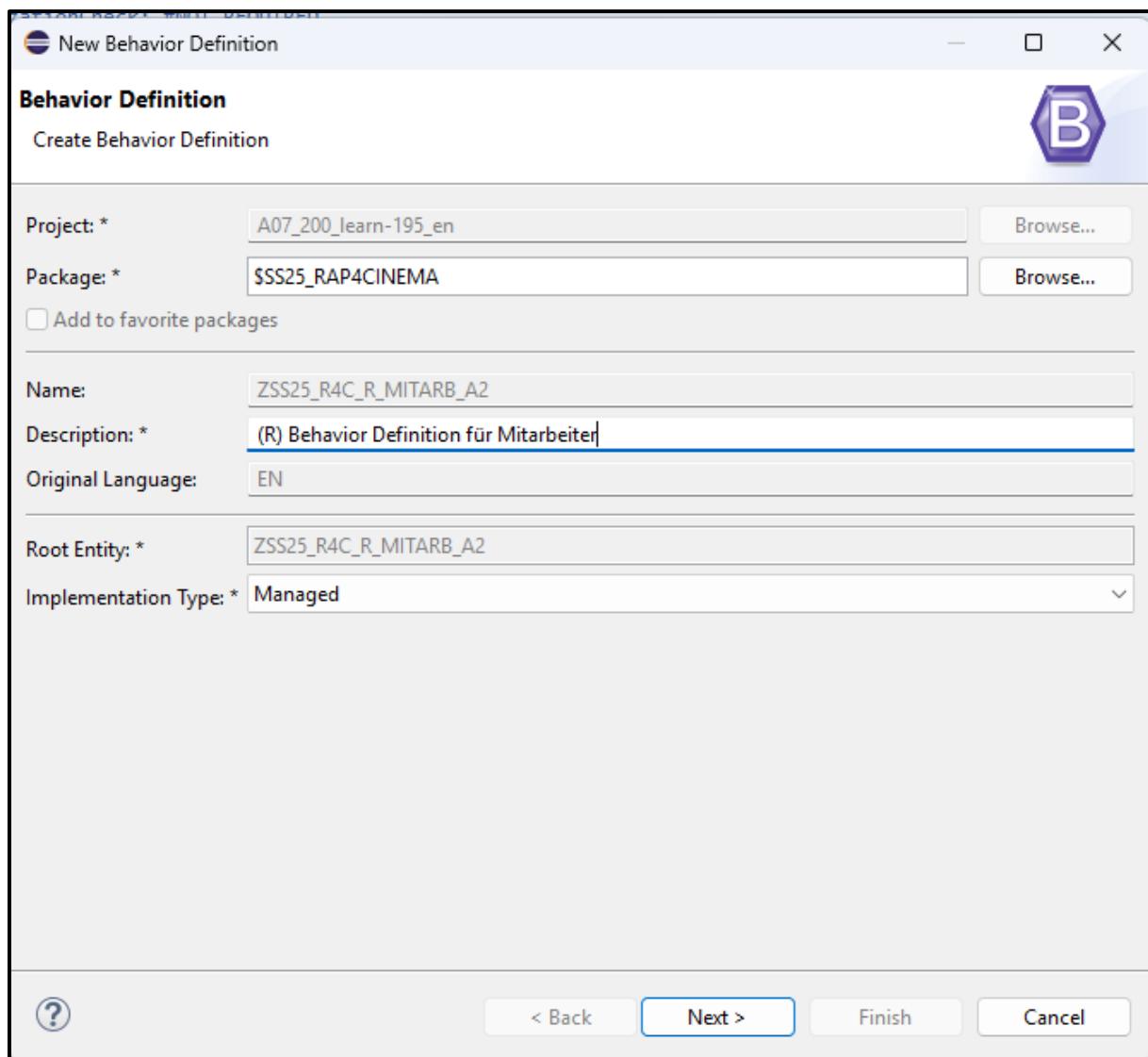
### 3.10. Behavior Definition

Die Behavior Definition legt das Verhalten einer Entität fest und definiert deren geschäftslogische Abläufe im Rahmen der erlaubten CRUD-Operationen (Create, Read, Update, Delete). Sie beschreibt, welche Aktionen zulässig sind und wie sich die Entität bei Änderungen verhalten soll - beispielsweise durch Validierungen, Berechnungen oder benutzerdefinierte Aktionen wie den Ticketkauf. Damit bildet die Behavior Definition die Business-Logik-Schicht innerhalb des RAP-Modells und ersetzt klassische ABAP-Implementierungen mit expliziten INSERT, UPDATE oder DELETE-Anweisungen. Sie sorgt für konsistente, nachvollziehbare und kontrollierte Datenänderungen und trägt wesentlich zur Modularität und Wartbarkeit der Anwendung bei.

## Schritt-für-Schritt-Anleitung:

Rechtsklick auf eine Reusable View → New Behavior Definition → Implementation Type und Beschreibung eingeben, z. B.: Implementation Type: Managed || Beschreibung: (R) Behavior Definition für Mitarbeiter || → Next → Finish





- Für **Mitarbeiter** (ZSS25\_R4C\_R\_MITARB\_A2) stellt sich der Inhalt der Behavior Definition wie folgt dar:

B [A07] ZSS25\_R4C\_R\_MITARB\_A2 X

```
1 managed implementation in class zbp_ss25_r4s_r_mitarb_a2 unique;
2 with draft;
3
4
5@define behavior for ZSS25_R4C_R_MITARB_A2 alias Besucher
6 persistent table ZSS25_R4C_MITARB
7 draft table ZSS25_R4C_DMITAR
8 lock master total etag LastChangedAt
9
10 etag master LocalLastChangedAt
11 {
12     create;
13     update;
14     delete;
15     draft action Activate;
16     draft action Discard;
17     draft action Edit;
18     draft action Resume;
19     draft determine action Prepare;
20
21
22     field ( mandatory: create ) MitarbeiterID;
23     field ( mandatory: create ) Name;
24     field ( mandatory: create ) geburtsdatum;
25     field ( mandatory: create ) rolle;
26     field ( mandatory: create ) abteilung;
27     field ( mandatory: create ) einstellungsdatum;
28     field ( mandatory: create ) gehalt;
29     field ( mandatory: create ) kontaktinfo;
30
31@ mapping for ZSS25_R4C_MITARB corresponding
32 {
33     MitarbeiterId = mitarbeiter_id;
34     Name = name;
35     Geburtsdatum = geburtsdatum;
36     Rolle = rolle;
37     Abteilung = abteilung;
38     Einstellungsdatum = einstellungsdatum;
39     Gehalt = gehalt;
40     Kontaktinfo = kontaktinfo;
```

```

40     Kontaktinfo = kontaktinfo;
41     CreatedAt = created_at;
42     CreatedBy = created_by;
43     LastChangedAt = last_changed_at;
44     LastChangedBy = last_changed_by;
45     LocalLastChangedAt = local_last_changed_at;
46 }
47
48 association _Verkauf { create; with draft; }
49 }
50
51
52@define behavior for ZSS25_R4C_R_VERKF_A2 alias Verkauf
53 persistent table ZSS25_R4C_VERKF
54 draft table ZSS25_R4C_DVERKF
55 lock dependent by _Mitarbeiter
56
57 etag master LocalLastChangedAt
58 {
59     update;
60     delete;
61
62     field ( readonly ) MitarbeiterID;
63     field ( mandatory: create ) VerkaufId;
64     field ( mandatory: create ) Rabatt;
65
66
67
68
69
70
71@ mapping for ZSS25_R4C_VERKF corresponding
72 {
73     VerkaufId = verkauf_id;
74     MitarbeiterId = mitarbeiter_id;
75     Verkaufsdatum = verkaufsdatum;
76     Zahlungsart = zahlungsart;
77     Rabatt = rabatt;
78     CreatedAt = created_at;
79     CreatedBy = created_by;

```

```

80     LastChangedAt = last_changed_at;
81     LastChangedBy = last_changed_by;
82     LocalLastChangedAt = local_last_changed_at;
83 }
84
85 validation checkVerkauf on save { create; update; }
86
87 association _Mitarbeiter { with draft; }
88 }

```

- Für **Kino** (ZSS25\_R4C\_R\_KINO\_A2) stellt sich der Inhalt der Behavior Definition wie folgt dar:

B [A07] ZSS25\_R4C\_R\_KINO\_A2 X

```
1 managed implementation in class zbp_ss25_r4s_r_kino_a2 unique;
2 strict ( 2 );
3 with draft;
4 define behavior for ZSS25_R4C_R_KINO_A2 alias Kino
5 persistent table ZSS25_R4C_KINO
6 draft table ZSS25_R4C_DKINO
7 lock master total etag LastChangedAt
8 authorization master ( instance )
9 etag master LocalLastChangedAt
10 {
11     create;
12     update;
13     delete;
14
15     draft action Activate;
16     draft action Discard;
17     draft action Edit;
18     draft action Resume;
19     draft determine action Prepare;
20
21     field ( mandatory: create ) KINOID;
22     field ( mandatory: create ) Name;
23
24
25 mapping for ZSS25_R4C_KINO corresponding
26 {
27     KinoId = kino_id;
28     Name = name;
29
30     Stadt = stadt;
31     Plz = plz;
32     AnzahlRaeume = anzahl_raeume;
33     Eroeffnungsdatum = eroeffnungsdatum;
34     Kontaktinfo = kontaktinfo;
35     CreatedAt = created_at;
36     CreatedBy = created_by;
37     LastChangedAt = last_changed_at;
38     LastChangedBy = last_changed_by;
39     LocalLastChangedAt = local_last_changed_at;
40 }
```

```

41    association _Raum { create; with draft; }
42
43
44
45 }
46
47
48 // ****
49 // ****
50 // ****
51 // ****
52 // ****
53@define behavior for ZSS25_R4C_R_RAUM_A2 alias Raum
54 persistent table ZSS25_R4C_RAUM
55 draft table ZSS25_R4C_DRAUM
56 lock dependent by _Kino
57 authorization dependent by _Kino
58 etag master LocalLastChangedAt
59 {
60     update;
61     delete;
62     field ( readonly ) KinoId;
63     field ( mandatory: create ) RaumId;
64
65@ mapping for ZSS25_R4C_RAUM corresponding
66 {
67
68     RaumId = raum_id;
69     KinoId = kino_id;
70     Name = name;
71     StuehlAnzahlProReihe = stuehl_anzahl_pro_reihe;
72     Sitzplaetze = sitzplaetze;
73     Ausstattung = ausstattung;
74     LocalLastChangedAt = local_last_changed_at;
75 }
76

```

```

77 validation checkRaum on save { create; update; delete; }
78 association _Kino { with draft; }
79 association _Vorstellung { create; with draft; }
80 }
81
82 // ****
83 // ****
84 // ****
85 // ****
86 // ****
87 // ****
88@ define behavior for ZSS25_R4C_R_VORST_A2 alias Vorstellung
89 persistent table ZSS25_R4C_VORST
90 draft table ZSS25_R4C_DVORST
91 lock dependent by _Kino
92 authorization dependent by _Kino
93 etag master LocalLastChangedAt
94 {
95     update;
96     delete;
97     field ( readonly ) Kinoid;
98     field ( readonly ) Raumid;
99     field ( mandatory: create ) Vorstellungid;
100
101@ mapping for ZSS25_R4C_VORST corresponding
102 {
103
104     Vorstellungid = vorstellung_id;
105     Raumid = raum_id;
106     Filmid = film_id;
107     Sprache = sprache;
108     Datum = datum;
109     Startzeit = startzeit;
110     Endzeit = endzeit;
111     Untertitel = untertitel;
112     VorstellungFormat = vorstellung_format;
113     Status = status;
114     LocalLastChangedAt    = local_last_changed_at;
115 }

```

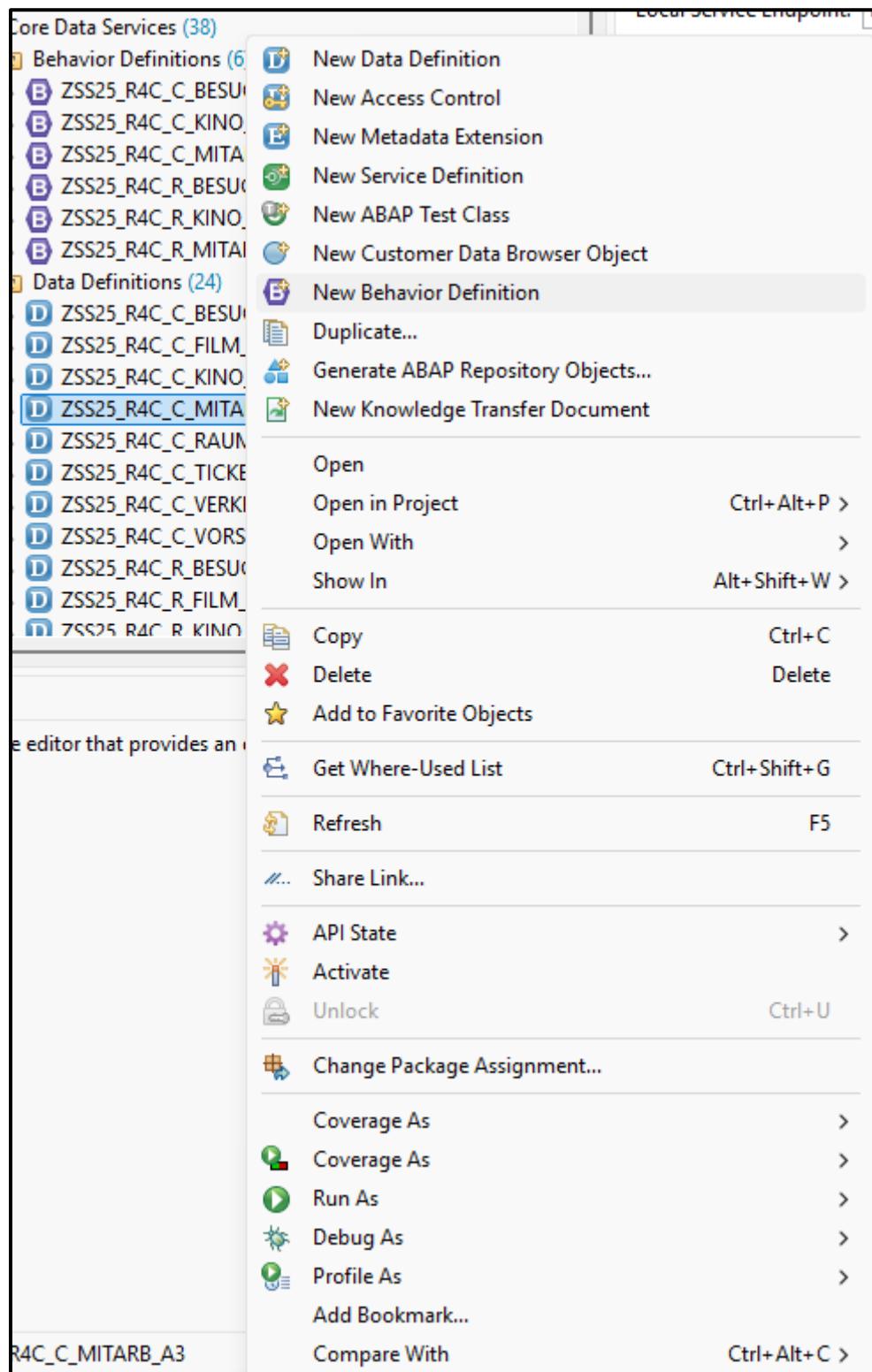
```

116
117     validation checkVorstellung on save { create; update; delete; }
118     association _Raum { with draft; }
119     association _Kino { with draft; }
120     association _Ticket { create; with draft; }
121 }
122
123
124 // ****
125 // ****
126 // ****
127 // ****
128 // ****
129 // ****
130 // ****
131 // ****
132 // ****
133 // ****
134 @define behavior for ZSS25_R4C_R_TICKET_A2 alias Ticket
135 persistent table ZSS25_R4C_TICKET
136 draft table ZSS25_R4C_DTICK
137 lock dependent by _Kino
138 authorization dependent by _Kino
139 etag master LocalLastChangedAt
140 {
141     update;
142     delete;
143     field ( readonly ) KinoId;
144
145     field ( readonly ) RaumId;
146     field ( readonly ) VorstellungId;
147     field ( mandatory: create ) TicketId;
148
149 @ mapping for ZSS25_R4C_TICKET corresponding
150 }
```

```

150 {
151     BesucherId      = besucher_id;
152     TicketId        = ticket_id;
153     Sitzplatznummer = sitzplatznummer;
154     Preis            = preis;
155     TicketTyp       = ticket_typ;
156     QrCode          = qr_code;
157     Status           = status;
158     LocalLastChangedAt = local_last_changed_at;
159 }
160
161 validation checkTicket on save { create; update; delete; }
162 association _Kino { with draft; }
163 association _Raum { with draft; }
164 association _Vorstellung { with draft; }
165
166 }
```

Rechtsklick auf eine Consumption View → New Behavior Definition → Implementation Type und Beschreibung eingeben, z. B.: Implementation Type: Projection || Beschreibung: (C) Behavior Definition für Mitarbeiter || → Next → Finish



New Behavior Definition

### Behavior Definition

Create Behavior Definition

Project: \* A07\_200\_learn-195\_en

Package: \* \$SS25\_RAP4CINEMA

Add to favorite packages

Name: ZSS25\_R4C\_C\_MITARB\_A3

Description: \* (C) Behavior Definition für Mitarbeiter

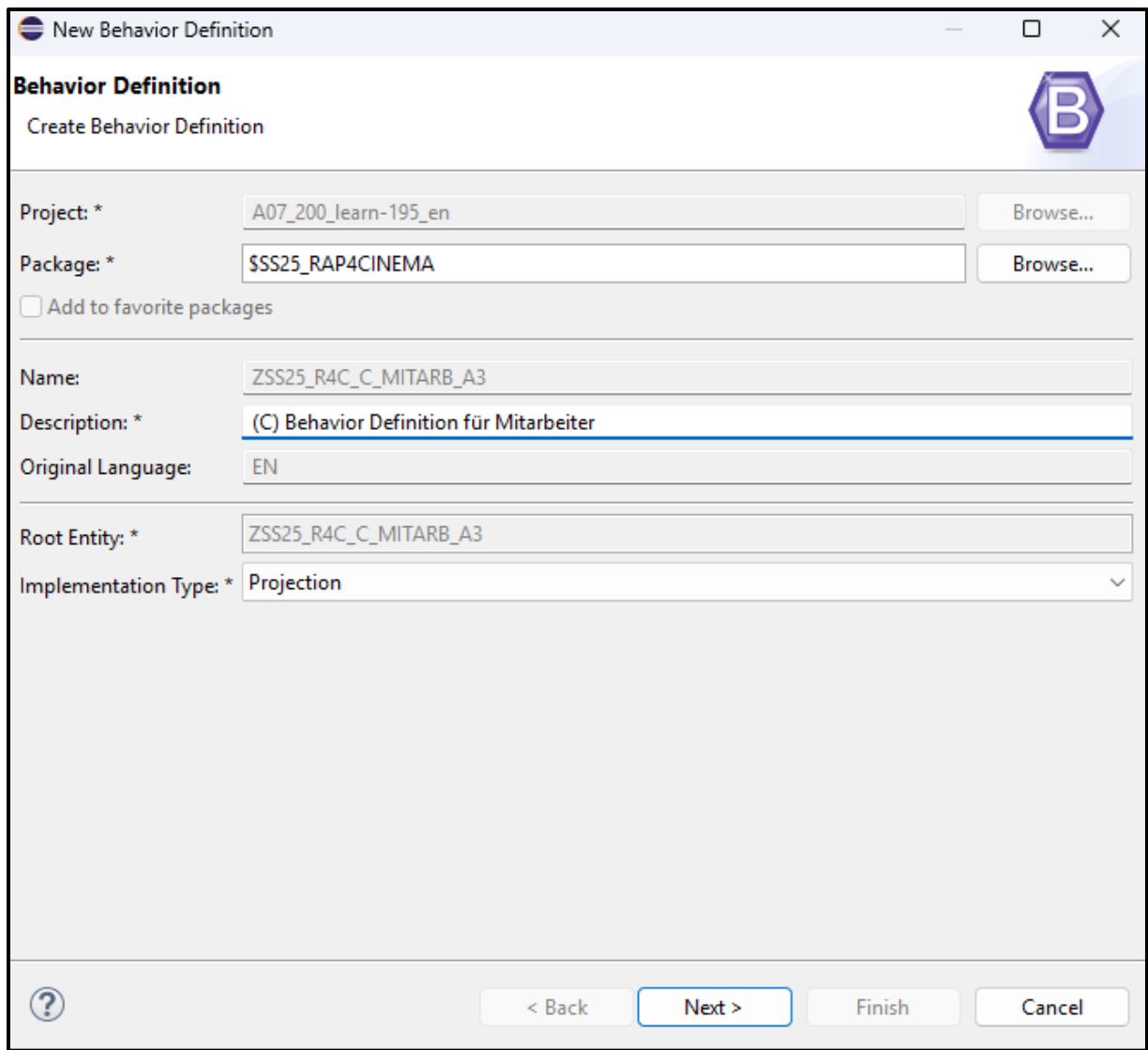
Original Language: EN

Root Entity: \* ZSS25\_R4C\_C\_MITARB\_A3

Implementation Type: \* Projection

?

< Back  Finish Cancel



- Für **Mitarbeiter** (ZSS25\_R4C\_C\_MITARB\_A3) stellt sich der Inhalt der Behavior Definition wie folgt dar:

B [A07] ZSS25\_R4C\_C\_MITARB\_A3 X

```
1 projection;
2 //strict ( 2 ); //Uncomment this line in order to enable strict mode 2.
3 use draft;
4
5 define behavior for ZSS25_R4C_C_MITARB_A3 alias Mitarbeiter
6 {
7   use create;
8   use update;
9   use delete;
10
11  use action Activate;
12  use action Discard;
13  use action Edit;
14  use action Resume;
15  use action Prepare;
16
17  use association _Verkauf { create; with draft; }
18 }
19
20 define behavior for ZSS25_R4C_C_VERKF_A3 alias Verkauf
21 {
22   use update;
23   use delete;
24
25   use association _Mitarbeiter { with draft; }
26 }
```

- Für **Kino** (ZSS25\_R4C\_C\_KINO\_A3) stellt sich der Inhalt der Behavior Definition wie folgt dar:

```

1 projection;
2 strict ( 2 );
3 use draft;
4
5@ define behavior for ZSS25_R4C_C_KINO_A3 alias Kino
6 {
7   use create;
8   use update;
9   use delete;
10
11  use action Activate;
12  use action Discard;
13  use action Edit;
14  use action Resume;
15  use action Prepare;
16
17  use association _Raum { create; with draft; }
18 }
19
20@ define behavior for ZSS25_R4C_C_RAUM_A3 alias Raum
21 {
22   use update;
23   use delete;
24
25   use association _Kino { with draft; }
26   use association _Vorstellung { create; with draft; }
27 }
28
29@ define behavior for ZSS25_R4C_C_VORST_A3 alias Vorstellung
30 {
31   use update;
32   use delete;
33
34   use association _Raum { with draft; }
35   use association _Kino { with draft; }
36   use association _Ticket { create; with draft; }
37 }
38

```

```

38
39@ define behavior for ZSS25_R4C_C_TICKET_A3 alias Ticket
40 {
41   use update;
42   use delete;
43
44   use association _Kino { with draft; }
45   use association _Raum { with draft; }
46   use association _Vorstellung { with draft; }
47 }

```

### Notiz:

- Nachdem die Behavior Definition erstellt wurde, muss sie mit Strg + S gespeichert und anschließend über das Aktivierungssymbol in SAP aktiviert werden.
- Nach erfolgreicher Aktivierung ist die Behavior Definition im Projektbaum unter dem Pfad Core Data Services → Behavior Definitions zu finden.

### 3.11. Draft Database Tables

Im SAP RAP-Modell dienen Draft Tables der temporären Zwischenspeicherung von Daten, die sich noch im Bearbeitungsprozess befinden. Sie ermöglichen es Benutzern, Änderungen an Datensätzen vorzunehmen, ohne diese sofort dauerhaft in die Haupttabelle zu schreiben. Dadurch wird die Arbeit mit Entwurfszuständen (Drafts) unterstützt - etwa beim Anlegen oder Ändern von Datensätzen, bevor diese final gespeichert oder verworfen werden.

Das Zusammenspiel zwischen Draft Tables und der Behavior Definition ist eng verzahnt:

- In der Behavior Definition wird festgelegt, ob eine Entität draft-fähig ist und wie sich das Draft-Verhalten konkret verhält (z. B. automatische Erzeugung, Speicherung oder Löschung von Entwürfen).
- Das RAP-Runtime-Framework übernimmt anschließend die Verwaltung der Draft-Daten, erstellt bei Bedarf automatisch Draft-Tabellen und sorgt für die Synchronisierung zwischen Entwurfs- und aktiven Daten.
- Entwickler können über Actions und Determinations innerhalb der Behavior Definition gezielt auf Draft-Ereignisse reagieren - etwa, um Validierungen oder Berechnungen während des Bearbeitungsvorgangs auszuführen.

Durch dieses Zusammenspiel wird ein transaktional konsistentes und benutzerfreundliches Bearbeitungserlebnis ermöglicht. Benutzer können Änderungen sicher vorbereiten, ohne bestehende Daten zu beeinflussen, während das System gleichzeitig sicherstellt, dass nur geprüfte und freigegebene Informationen in die aktive Datenhaltung übernommen werden.

#### Notiz:

- Nachdem die Draft Tabellenstruktur automatisch erstellt wurde, muss sie mit Strg + S gespeichert und anschließend über das Aktivierungssymbol in SAP aktiviert werden.
- Nach erfolgreicher Aktivierung sind die Database Draft Tables im Projektbaum unter dem Pfad Dictionary → Database Tables zu finden.

## 4. Erweiterungen und neue Funktionalitäten in SAP RAP

### 4.1. Pflichtfelder (Obligatorische Felder)

Zur Sicherstellung einer hohen Datenqualität wurden in RAP4Cinema obligatorische Filterfelder definiert. Diese Pflichtfelder müssen vom Benutzer im Fiori-Filterbereich

zwingend ausgefüllt werden, bevor eine Suche oder Datenselektion ausgeführt werden kann. Die technische Umsetzung erfolgt über die Annotation:

```
@Consumption.filter.mandatory: true
```

```
// --- Zahlungsart ---
@EndUserText.label: 'Zahlungsart'
@UI.selectionField: [{ position: 20 }]
@UI.identification: [{ position: 40 }]
@UI.lineItem: [{ position: 40 }]
@Consumption.filter.mandatory: true
Zahlungsart;
```

Zahlungsart:\*

A screenshot of a Fiori application interface. It shows a text input field with a red border and a red dashed outline, indicating it is required. To the right of the input field is a blue button labeled "Go". Below the input field, a red banner displays the message "Required value missing".

## 4.2. Verwendung der QuickInfo-Annotation

@EndUserText.quickInfo: 'Text' wird verwendet, um einem Feld in einer CDS-View oder einem Entity einen kurzen erklärenden Text zuzuweisen, der im Benutzerinterface angezeigt wird, meist als Tooltip oder QuickInfo. Dieser Text hilft den Endanwendern, die Bedeutung oder den Zweck des Feldes schnell zu verstehen, ohne zusätzliche Dokumentation oder Schulung. In unserer Meta-Extension haben wir diese Annotation für alle relevanten Felder eingesetzt, um die Benutzerfreundlichkeit der Anwendung zu erhöhen. So kann ein Benutzer im UI sofort erkennen, wofür ein Feld wie „Betrag“, „Tickettyp“ oder „Besuchergruppe“ steht, ohne die Logik dahinter kennen zu müssen. Die QuickInfo sorgt damit für klar verständliche, selbsterklärende Felder und unterstützt eine konsistente Darstellung in der Fiori-Oberfläche.

```
// --- Verkauf ---
@EndUserText.label: 'Rechnungsnummer'
@EndUserText.quickInfo: 'Eindeutige Nummer der Rechnung bzw. des Verkaufs.'
VerkaufId;
```

Rechnungsnummer	Ticket Art	
Eindeutige Nummer der Rechnung bzw. des Verkaufs.		
9001	KIND	>
9002	STANDARD	>
9005	VIP	>
9006	VIP	>
9007	KIND	>
9008	VIP	>

### 4.3. Unterstützung von Datumsbereichen

Hier wird beschrieben, wie Datumsbereiche innerhalb der Anwendung verarbeitet werden. Dies ermöglicht es beispielsweise, Verkaufsdaten nach einem bestimmten Datum oder einem Datumsbereich - wie dem heutigen oder gestrigen Datum - zu filtern.

Die technische Umsetzung erfolgt über die folgende Annotation:

```
@Consumption.filter:{ selectionType: #INTERVAL, multipleSelections: false }
```

```
// --- Verkaufsdatum ---
@EndUserText.label: 'Verkaufsdatum'
@UI.selectionField: [{ position: 10 }]
@UI.identification: [{ position: 30 }]
@UI.lineItem: [{ position: 30 }]
@Consumption.filter:{ selectionType: #INTERVAL, multipleSelections: false }
Verkaufsdatum;
```

Verkaufsdatum:

[Calendar icon]

**Single Dates**

Date [Calendar icon]

Today

Yesterday

Tomorrow

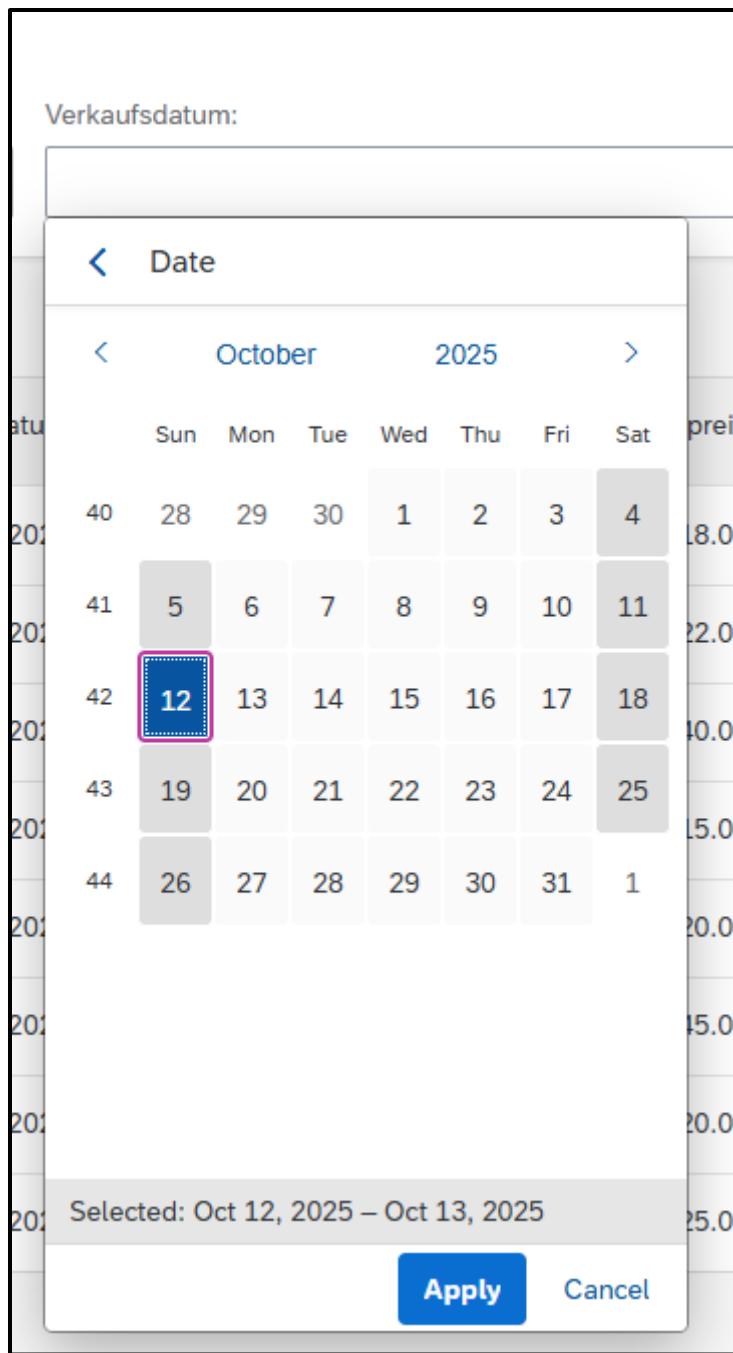
**Date Ranges**

From / To [Calendar icon]

From [Calendar icon]

To [Calendar icon]

Year to Date



Besuchername	Verkaufsdatum	Zahlungsart	Ticketpreis	Rabatt	Gesamtbetrag
LINH NGUYEN	Oct 12, 2025	KARTE	18.00	10.00	16.20

#### 4.4. Farbige Kennzeichnung von Datensätzen

Dieser Code-Abschnitt zeigt, wie Datensätze im SAP RAP farblich hervorgehoben werden können. Farbige Markierungen erleichtern die visuelle Differenzierung

bestimmter Kennzahlen. Beispielsweise wird ein Datensatz grün markiert, wenn der Eintritt eines Besuchers erlaubt ist, und rot, wenn er verweigert wird.

In ZSS25\_R4C\_V\_VERKF\_A1:

```
// ErlaubnisKritikalitaet
@EndUserText.label: 'Criticality für Rabatt'
case
  when bes.BesucherAlter < flm.Altersfreigabe then 1 -- Rot (negative)
  else 3 -- Grün (positive)
end as ErlaubnisKritikalitaet
```

In ZSS25\_R4C\_R\_VERKF\_A2:

```
D [A07] ZSS25_R4C_R_VERKF_A2 ×
1 @AccessControl.authorizationCheck: #NOT_REQUIRED
2 @EndUserText.label: 'Reusable View für Verkauf'
3 @Metadata.ignorePropagatedAnnotations: true
4 define view entity ZSS25_R4C_R_VERKF_A2 as select from ZSS25_R4C_V_VERKF_A1
5 association to parent ZSS25_R4C_R_MITARB_A2 as _Mitarbeiter
6   on _Mitarbeiter.MitarbeiterId = $projection.MitarbeiterId
7 association [1..1] to ZSS25_R4C_R_TICKET_A2 as _Ticket
8   on _Ticket.TicketId = $projection.TicketId
9 {
10   key VerkaufId,
11     TicketId,
12     MitarbeiterId,
13     MitarbeiterName,
14     BesucherName,
15     Verkaufsdatum,
16     Zahlungsart,
17     TicketPreis,
18     Rabatt,
19     Betrag,
20     BesucherAltersgruppe,
21     BesucherIstErlaubt,
22     TicketSitzplatznummer,
23     BesucherSitzReihe,
24     TicketIstGueltig,
25     Zeitkategorie,
26     TicketStatus,
27     ErlaubnisKritikalitaet,ErlaubnisKritikalitaet,
28     CreatedBy,
29     CreatedAt,
30     LastChangedBy,
31     LastChangedAt,
32     LocalLastChangedAt,
33     _Mitarbeiter,
34     _Ticket
35 }
36
```

In ZSS25\_R4C\_C\_VERKF\_A3:

```

D [A07] ZSS25_R4C_C_VERKF_A3 X
1 @AccessControl.authorizationCheck: #NOT_REQUIRED
2 @EndUserText.label: 'Consumption View für Verkauf'
3 @Metadata.ignorePropagatedAnnotations: true
4 @Metadata.allowExtensions: true
5 @Search.searchable: true
6 define view entity ZSS25_R4C_C_VERKF_A3 as projection on ZSS25_R4C_R_VERKF_A2
7 {
8     @EndUserText.label: 'Rechnungsnummer'
9     @Search.defaultSearchElement: true
10    key VerkaufId,
11    TicketId,
12    MitarbeiterId,
13    MitarbeiterName,
14    BesucherName,
15    Verkaufsdatum,
16    Zahlungsart,
17    TicketPreis,
18    Rabatt,
19    Betrag,
20    BesucherAltersgruppe,
21    BesucherIstErlaubt,
22    TicketSitzplatznummer,
23    BesucherSitzReihe,
24    TicketIstGueltig,
25    Zeitkategorie,
26    TicketStatus,
27    ErlaubnisKritikalitaet,
28    CreatedBy,
29    CreatedAt,
30    LastChangedBy,
31    LastChangedAt,
32    LocalLastChangedAt,
33
34    /* Associations */
35    _Mitarbeiter : redirected to parent ZSS25_R4C_C_MITARB_A3,
36    _Ticket: redirected to ZSS25_R4C_C_TICKET_A3
37 }

```

In Metadata Extension:

```

// --- Besucher ist erlaubt ---
@EndUserText.label: 'Besucher ist erlaubt'
@UI.identification: [{ position: 90 }]
@UI.lineItem: [{ position: 90 , criticality: 'ErlaubnisKritikalitaet' }]
BesucherIstErlaubt;

```

**Ergebnis:**

Besucher-Altersgr...	Besucher ist erl...
Kinds	Verboten
Erwachsen	Erlaubt
Senior	Erlaubt

## 4.5. Integration von Bildern und Symbolen

In diesem Abschnitt wird erläutert, wie in RAP4Cinema Bilder und Icons in Benutzeroberflächen eingebunden werden können. Dies umfasst sowohl die Anzeige von Medieninhalten wie Filmplakaten als auch die Nutzung von Symbolen zur Verbesserung der Benutzerführung und Intuitivität der App.

Eine neue Spalte (abap.string(256)) wird zur Tabelle ZSS25\_R4C\_MITARB hinzugefügt:

```

[ A07] ZSS25_R4C_MITARB X
1 @EndUserText.label : 'Tabelle für Mitarbeiter'
2 @AbapCatalog.enhancement.category : #NOT_EXTENSIBLE
3 @AbapCatalog.tableCategory : #TRANSPARENT
4 @AbapCatalog.deliveryClass : #A
5 @AbapCatalog.dataMaintenance : #ALLOWED
6 define table zss25_r4c_mitarb {
7   key mandt          : abap.clnt not null;
8   key mitarbeiter_id : abap.numc(10) not null;
9   name              : abap.char(100);
10  fotourl           : abap.string(256);
11  geburtsdatum      : abap.dats;
12  rolle              : abap.char(50);
13  abteilung          : abap.char(50);
14  einstellungsdatum : abap.dats;
15  gehalt              : abap.dec(10,2);
16  kontaktinfo        : abap.char(200);
17  created_by         : abp_creation_user;
18  created_at          : abp_creation_tstmp;
19  last_changed_by    : abp_locinst_lastchange_user;
20  last_changed_at     : abp_locinst_lastchange_tstmp;
21  local_last_changed_at : abp_lastchange_tstmp;
22
23 }

```

Die neue Spalte wird in den Views ZSS25\_R4C\_V\_VERKF\_A1, ZSS25\_R4C\_R\_VERKF\_A2 und ZSS25\_R4C\_C\_VERKF\_A3 selektiert:

```

[D] [ A07] ZSS25_R4C_V_VERKF_A1 X
28   vkf.created_by as CreatedBy,
29   vkf.created_at as CreatedAt,
30   vkf.last_changed_by as LastChangedBy,
31   vkf.last_changed_at as LastChangedAt,
32   vkf.local_last_changed_at as LocalLastChangedAt,
33
34
35
36@ // ****
37 // MITARBEITER
38 // ****
39 ma.Name as MitarbeiterName,
40 ma.FotoURL as MitarbeiterFoto,
41 ma.Geburtsdatum as MitarbeiterGeburtsdatum,
42 ma.Rolle as MitarbeiterRolle,
43 ma.Abteilung as MitarbeiterAbteilung,
44 ma.Einstellungsdatum as MitarbeiterEinstellungsdatum,
45 ma.Gehalt as MitarbeiterGehalt,
46 ma.Kontaktinfo as MitarbeiterKontaktinfo,
47

```

D [A07] ZSS25\_R4C\_R\_VERKF\_A2 X

```
1@AccessControl.authorizationCheck: #NOT_REQUIRED
2@EndUserText.label: 'Reusable View für Verkauf'
3@Metadata.ignorePropagatedAnnotations: true
4define view entity ZSS25_R4C_R_VERKF_A2 as select from ZSS25_R4C_V_VERKF_A1
5association to parent ZSS25_R4C_R_MITARB_A2 as _Mitarbeiter
6    on _Mitarbeiter.MitarbeiterId = $projection.MitarbeiterId
7association [1..1] to ZSS25_R4C_R_TICKET_A2 as _Ticket
8    on _Ticket.TicketId = $projection.TicketId
9{
10    key VerkaufId,
11    TicketId,
12    MitarbeiterId,
13    MitarbeiterName,
14    BesucherName,
15    MitarbeiterFoto, MitarbeiterFoto,
16    Verkaufsdatum,
17    Zahlungsart,
18    TicketPreis,
19    TicketTyp,
20    Rabatt,
21    Betrag,
22    BesucherAltersgruppe,
23    BesucherIstErlaubt,
24    TicketSitzplatznummer,
25    BesucherSitzReihe,
26    TicketIstGueltig,
27    Zeitkategorie,
28    TicketStatus,
29    ErlaubnisKritikalitaet,
30    CreatedBy,
31    CreatedAt,
32    LastChangedBy,
33    LastChangedAt,
34    LocalLastChangedAt,
35    _Mitarbeiter,
36    _Ticket
37}
38
```

In der Consumption-View wird über dieser Spalte die Annotation @Semantics.imageUrl: true hinzugefügt.

```
18
19    @Semantics.imageUrl: true
20    MitarbeiterFoto,
```

```

D [A07] ZSS25_R4C_C_VERKF_A3 X
1 @AccessControl.authorizationCheck: #NOT_REQUIRED
2 @EndUserText.label: 'Consumption View für Verkauf'
3 @Metadata.ignorePropagatedAnnotations: true
4 @Metadata.allowExtensions: true
5 @Search.searchable: true
6 @ObjectModel.resultSet.sizeCategory: #XS
7 define view entity ZSS25_R4C_C_VERKF_A3 as projection on ZSS25_R4C_R_VERKF_A2
8 {
9     @EndUserText.label: 'Rechnungsnummer'
10    @Search.defaultSearchElement: true
11    key VerkaufId,
12    TicketId,
13    MitarbeiterId,
14    @Consumption.valueHelpDefinition: [{entity: {name:'ZSS25_R4C_V_MITNAMEN',element: 'MitarbeiterName'} }]
15    @UI.textArrangement: #TEXT_ONLY
16    @ObjectModel.virtualElement: true
17    MitarbeiterName,
18
19    @Semantics.imageUrl: true
20    MitarbeiterFoto,
21    BesucherName,
22    Verkaufsdatum,
23    Zahlungsart,
24    TicketPreis,
25    TicketTyp,
26    Rabatt,
27    Betrag,
28    BesucherAltersgruppe,
29    BesucherIstErlaubt,
30    TicketSitzplatznummer,
31    BesucherSitzReihe,
32    TicketIstGueltig,
33    Zeitkategorie,
34    TicketStatus,
35    ErlaubnisKritikalitaet,
36    CreatedBy,
37    CreatedAt,
38    LastChangedBy,
39    LastChangedAt,
40    LocalLastChangedAt,
41
42
43    /* Associations */
44    _Mitarbeiter : redirected to parent ZSS25_R4C_C_MITARB_A3,
45    _Ticket: redirected to ZSS25_R4C_C_TICKET_A3
46 }
47

```

```

38
39 // --- Mitarbeiter-Foto ---
40 @EndUserText.label: ''
41 @EndUserText.quickInfo: 'Foto des Mitarbeiters, der den Verkauf durchgeführt hat.'
42 @UI.identification: [{ position: 20 }]
43 @UI.lineItem: [{ position: 20 }]
44 @UI.selectionField: [{ position: 20 }]
45 MitarbeiterFoto;

```

## Ergebnis:

Datum	Besucher	Besuchergruppe
Oct 11, 2025	 ANNA MÜLLER	Erwachsen
Oct 15, 2025	 TOBIAS SCHMIDT	Senior
Oct 14, 2025	 OMAR ALI	Erwachsen
Oct 12, 2025	 SOPHIE WAGNER	Erwachsen
Oct 27, 2025	 MARCO ROSSI	Erwachsen
Oct 15, 2025	 ANNA MÜLLER	Erwachsen
Oct 22, 2025	 LINH NGUYEN	Erwachsen
Oct 30, 2025	 JANA BECKER	Kinds
Oct 17, 2025	 SOPHIE WAGNER	Erwachsen
Oct 28, 2025	 MARCO ROSSI	Erwachsen
Oct 25, 2025	 ANNA MÜLLER	Erwachsen

### Für Zahlungsstatus:

Eine Tabelle ZSS25\_R4C\_ZAHL wird erstellt, und die URLs für die Fotos werden in der Datenbank eingetragen:

```

[A07] ZSS25_R4C_ZAHL X
1 @EndUserText.label : 'Tabelle für Zahlungsstatus'
2 @AbapCatalog.enhancement.category : #NOT_EXTENSIBLE
3 @AbapCatalog.tableCategory : #TRANSPARENT
4 @AbapCatalog.deliveryClass : #A
5 @AbapCatalog.dataMaintenance : #RESTRICTED
6 define table zss25_r4c_zahl {
7   key client : abap.clnt not null;
8   key status : abap.char(20) not null;
9   fotourl    : abap.string(256);
10
11 }

```

STATUS	FOTOURL
STORNIERT	<a href="https://cdn-icons-png.flaticon.com/128/7699/7699001.png">https://cdn-icons-png.flaticon.com/128/7699/7699001.png</a>
OFFEN	<a href="https://cdn-icons-png.flaticon.com/128/694/694572.png">https://cdn-icons-png.flaticon.com/128/694/694572.png</a>
BAZAHLT	<a href="https://cdn-icons-png.flaticon.com/128/4315/4315445.png">https://cdn-icons-png.flaticon.com/128/4315/4315445.png</a>

Auf dieser Tabelle wird ein CDS-View erstellt:

```

[D] [A07] ZSS25_R4C_V_ZAHL X
1 @AbapCatalog.viewEnhancementCategory: [#NONE]
2 @AccessControl.authorizationCheck: #NOT_REQUIRED
3 @EndUserText.label: 'View für Zahlungsstatus'
4 @Metadata.ignorePropagatedAnnotations: true
5 @ObjectModel.usageType:{ 
6   serviceQuality: #X,
7   sizeCategory: #S,
8   dataClass: #MIXED
9 }
10 define view entity ZSS25_R4C_V_ZAHL as select from zss25_r4c_zahl
11 {
12   key status as Status,
13   fotourl as Fotourl
14 }

```

In der View ZSS25\_R4C\_VERKF\_A1 wird die Foto-URL über einen Join eingebunden:

D [A07] ZSS25\_R4C\_V\_VERKF\_A1 X

```
1
2 @AbapCatalog.viewEnhancementCategory: [#NONE]
3 @AccessControl.authorizationCheck: #NOT_REQUIRED
4 @EndUserText.label: 'View für Verkauf (Faktentabelle)'
5 @Metadata.ignorePropagatedAnnotations: true
6 @ObjectModel.usageType:{ 
7     serviceQuality: #X,
8     sizeCategory: #S,
9     dataClass: #MIXED
10 }
11 define view entity ZSS25_R4C_V_VERKF_A1 as select from zss25_r4c_verkf as vkf
12 left outer join ZSS25_R4C_V_MITARB_A1 as ma on ma.MitarbeiterId = vkf.mitarbeiter_id
13 left outer join ZSS25_R4C_V_TICKET_A1 as tkt on tkt.TicketId = vkf.ticket_id
14 left outer join ZSS25_R4C_V_BESUCH_A1 as bes on bes.BesucherId = tkt.BesucherId
15 left outer join ZSS25_R4C_V_VORST_A1 as vorst on vorst.VorstellungId = tkt.VorstellungId
16 left outer join ZSS25_R4C_V_FILM_A1 as flm on flm.FilmId = vorst.FilmId
17 left outer join ZSS25_R4C_V_RAUM_A1 as rm on rm.RaumId = vorst.RaumId
18 left outer join ZSS25_R4C_V_KINO_A1 as kn on kn.KinoId = rm.KinoId
19 left outer join ZSS25_R4C_V_ZAHL as zhl on tkt.Status = zhl.Status
20
21 {
```

```
50
51 // ****
52 // TICKET
53 // ****
54 tkt.Sitzplatznummer as TicketSitzplatznummer,
55 tkt.Preis as TicketPreis,
56 tkt.TicketTyp as TicketTyp,
57 tkt.QrCode as TicketQrCode,
58 tkt.Status as TicketStatus,
59 zhl.Fotourl as ZahlungsstatusURL,
60
```

```

1@AccessControl.authorizationCheck: #NOT_REQUIRED
2@EndUserText.label: 'Data Definition ZSS25_R4C_R_VERKF_A2 [A07,200] - inactive - A07_200_le'
3@Metadata.ignorePropagatedAnnotations: true
4define view entity ZSS25_R4C_R_VERKF_A2 as select from ZSS25_R4C_V_VERKF_A1
5association to parent ZSS25_R4C_R_MITARB_A2 as _Mitarbeiter
6    on _Mitarbeiter.MitarbeiterId = $projection.MitarbeiterId
7association [1..1] to ZSS25_R4C_R_TICKET_A2 as _Ticket
8    on _Ticket.TicketId = $projection.TicketId
9{
10    key VerkaufId,
11    TicketId,
12    MitarbeiterId,
13    MitarbeiterName,
14    BesucherName,
15    MitarbeiterFoto,
16    Verkaufsdatum,
17    Zahlungsart,
18    TicketPreis,
19    TicketTyp,
20    Rabatt,
21    Betrag,
22    BesucherAltersgruppe,
23    BesucherIstErlaubt,
24    TicketSitzplatznummer,
25    BesucherSitzReihe,
26    TicketIstGueltig,
27    Zeitkategorie,
28    TicketStatus,
29    ZahlungsstatusURL,
30    ErlaubnisKritikalitaet,
31    CreatedBy,
32    CreatedAt,
33    LastChangedBy,
34    LastChangedAt,
35    LocalLastChangedAt,
36    _Mitarbeiter,
37    _Ticket
38}

```

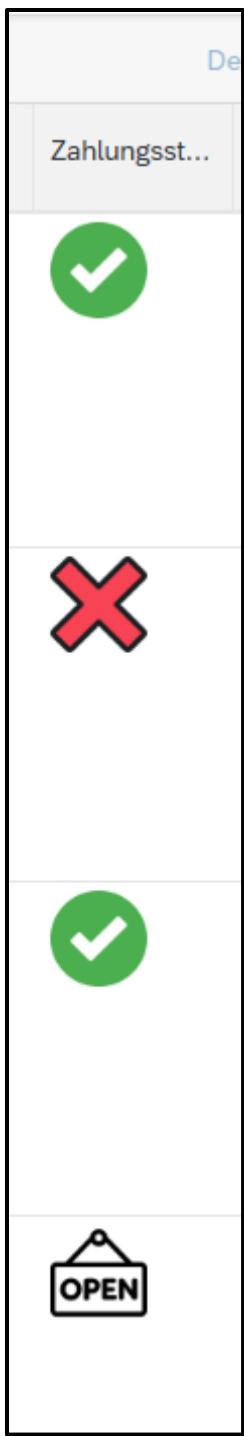
In der Consumption-View wird über dieser Spalte die Annotation @Semantics.imageUrl: true hinzugefügt:

```

36    TicketStatus,
37    |
38    @Semantics.imageUrl: true
39    ZahlungsstatusURL,
40    |
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104@EndUserText.label: 'Zahlungsstatus'
105@EndUserText.quickInfo: 'Status des Tickets (bezahllt, storniert, offen).'
106@UI.identification: [{ position: 110 }]
107@UI.lineItem: [{ position: 110 }]
108ZahlungsstatusURL;

```

**Ergebnis:**



Analog wird für andere Spalten genauso vorgegangen wie beim Zahlungsstatus.

## 4.6. Erstellung von Drop Down

In RAP4Cinema können Drop Downs verwendet werden, um die Auswahl von Werten für Filter oder Eingabefelder zu erleichtern. Ein typisches Szenario ist das Filtern von Besuchern nach ihrem Namen. Drop Downs werden in der Regel über CDS-Views in Verbindung mit UI-Annotations definiert.

## Erstellung einer CDS-View (ZSS25\_R4C\_V\_BESUCHNAMEN):

```
D [A07] ZSS25_R4C_V_BESUCHNAMEN X
1 @AbapCatalog.viewEnhancementCategory: [#NONE]
2 @AccessControl.authorizationCheck: #NOT_REQUIRED
3 @EndUserText.label: 'View für Besucherernamen'
4 @Metadata.ignorePropagatedAnnotations: true
5 @ObjectModel.usageType:{
6     serviceQuality: #X,
7     sizeCategory: #S,
8     dataClass: #MIXED
9 }
10 @ObjectModel.resultSet.sizeCategory: #XS
11 define view entity ZSS25_R4C_V_BESUCHNAMEN as select from zss25_r4c_besuch as bes
12 association [1..1] to ZSS25_R4C_V_VERKF_A1 as vkf
13 on bes.name = vkf.BesucherName
14 {
15     key
16     name as BesucherName
17 }
18
19
```

Folgende Annotationen werden für die CDS-View (ZSS25\_R4C\_V\_BESUCHNAMEN) verwendet:

```
6 @ObjectModel.resultSet.sizeCategory: #XS
```

Folgende Annotationen werden für die Consumption View (ZSS25\_R4C\_C\_VERKF\_A3) verwendet:

```
6 @ObjectModel.resultSet.sizeCategory: #XS
```

```
20
21 @Consumption.valueHelpDefinition: [{entity: {name: 'ZSS25_R4C_V_BESUCHNAMEN', element: ['BesucherName']} }]
22 @UI.textArrangement: #TEXT_ONLY
23 BesucherName,
```

```

[D] [A07] ZSS25_R4C_C_VERKF_A3 X
1 @AccessControl.authorizationCheck: #NOT_REQUIRED
2 @EndUserText.label: 'Consumption View für Verkauf'
3 @Metadata.ignorePropagatedAnnotations: true
4 @Metadata.allowExtensions: true
5 @Search.searchable: true
6 @ObjectModel.resultSet.sizeCategory: #XS
7 define view entity ZSS25_R4C_C_VERKF_A3 as projection on ZSS25_R4C_R_VERKF_A2
8 {
9     @EndUserText.label: 'Rechnungsnummer'
10    @Search.defaultSearchElement: true
11    key VerkaufId,
12    TicketId,
13    MitarbeiterId,
14
15    @ObjectModel.virtualElement: true
16    MitarbeiterName,
17
18    @Semantics.imageUrl: true
19    MitarbeiterFoto,
20
21    @Consumption.valueHelpDefinition: [{entity: {name:'ZSS25_R4C_V_BESUCHNAMEN',element: 'BesucherName'} }]
22    @UI.textArrangement: #TEXT_ONLY
23    BesucherName,
24    Verkaufsdatum,

```

In der Metadata Extension sieht der Besuchernname-Filter wie folgt aus:

```

52 // --- Besucher ---
53 @EndUserText.label: 'Besucher'
54 @EndUserText.quickInfo: 'Hier werden die Namen aller Besucher angezeigt.'
55 @UI.identification: [{ position: 30 }]
56 @UI.lineItem: [{ position: 30 }]
57 @UI.selectionField: [{ position: 20 }]
58 BesucherName;
59

```

**Ergebnis:**

Besucher:   Zahlu

<input type="checkbox"/>	ANNA MÜLLER
<input checked="" type="checkbox"/>	JANA BECKER
<input type="checkbox"/>	LINH NGUYEN
<input checked="" type="checkbox"/>	MARCO ROSSI
<input type="checkbox"/>	OMAR ALI
<input type="checkbox"/>	SOPHIE WAGNER
<input type="checkbox"/>	TOBIAS SCHMIDT

Verkäufe (4)

	Datum	Besucher	Besuchergruppe
<input type="checkbox"/>	Oct 27, 2025	 MARCO ROSSI	Erwachsen
<input type="checkbox"/>	Oct 30, 2025	 JANA BECKER	Kinds
<input type="checkbox"/>	Oct 28, 2025	 MARCO ROSSI	Erwachsen
<input type="checkbox"/>	Oct 12, 2025	 MARCO ROSSI	Erwachsen

Die Metadata Extension ZSS25\_R4C\_ME\_VERKF\_A6 sieht wie folgt aus:

```
E [A07] ZSS25_R4C_ME_VERKF_A6 X
1@Metadata.layer: #CUSTOMER
2
3 @UI: {
4   headerInfo: {
5     typeName: 'Verkauf',
6     typeNamePlural: 'Verkäufe'
7   }
8 }
9
10
11 annotate entity ZSS25_R4C_ME_VERKF_A3 with
12 {
13// -----
14// FACETS
15// -----
16  @UI.facet: [
17    { position: 10, label: 'Verkauf', type: #IDENTIFICATION_REFERENCE }
18 ]
19
20// -----
21// FELDER IN SINNVOLLER REIHENFOLGE
22// -----
23
24// --- Verkauf ---
25@EndUserText.label: 'Rechnungsnummer'
26@EndUserText.quickInfo: 'Eindeutige Nummer der Rechnung bzw. des Verkaufs.'
27VerkaufId;
28
29
30// --- Mitarbeiter-Foto ---
31@EndUserText.label: ''
32@EndUserText.quickInfo: 'Foto des Mitarbeiters, der den Verkauf durchgeführt hat.'
33@UI.identification: [{ position: 10 }]
34@UI.lineItem: [{ position: 10 }]
35MitarbeiterFoto;
36
```

```

37 @EndUserText.label: 'Datum'
38 @EndUserText.quickInfo: 'Datum, an dem der Verkauf stattgefunden hat.'
39 @UI.selectionField: [{ position: 10 }]
40 @UI.identification: [{ position: 20 }]
41 @UI.lineItem: [{ position: 20 }]
42 @Consumption.filter:{ selectionType: #INTERVAL, multipleSelections: false }
43 Verkaufsdatum;
44
45
46
47
48
49 @EndUserText.label: 'Mitarbeiter-ID'
50 @EndUserText.quickInfo: 'Eindeutige Kennung des Mitarbeiters (internes Feld).'
51 @UI.hidden: true
52 MitarbeiterId;
53
54
55 // --- Besucher ---
56 @EndUserText.label: 'Besucher'
57 @EndUserText.quickInfo: 'Hier werden die Namen aller Besucher angezeigt.'
58 @UI.identification: [{ position: 30 }]
59 @UI.lineItem: [{ position: 30 }]
60 @UI.selectionField: [{ position: 20 }]
61 BesucherName;
62
63 @EndUserText.label: 'Erlaubnis'
64 @EndUserText.quickInfo: 'Zeigt an, ob der Besucher zum Eintritt berechtigt ist.'
65 @UI.identification: [{ position: 40 }]
66 @UI.lineItem: [{ position: 40, criticality: 'ErlaubnisKritikalitaet' }]
67 BesucherIstErlaubt;
68
69 @EndUserText.label: 'Besuchergruppe'
70 @EndUserText.quickInfo: 'Gibt die Altersgruppe des Besuchers an.'
71 @UI.identification: [{ position: 50 }]
72 @UI.lineItem: [{ position: 50 }]
73 BesucherAltersgruppeURL;
74

```

```

76
77
78@ // --- Ticketart ---
79 @EndUserText.label: 'Ticketart'
80 @EndUserText.quickInfo: 'Art bzw. Typ des ausgestellten Tickets.'
81 @UI.identification: [{ position: 60 }]
82 @UI.lineItem: [{ position: 60 }]
83 TicketTypURL;
84
85
86
87@ @EndUserText.label: 'Ticketgültigkeit'
88 @EndUserText.quickInfo: 'Gibt an, ob das Ticket aktuell gültig ist.'
89 @UI.identification: [{ position: 70 }]
90 @UI.lineItem: [{ position: 70 }]
91 TicketIstGueltigURL;
92
93@ @EndUserText.label: 'Zeitkategorie'
94 @EndUserText.quickInfo: 'Zeitliche Kategorie der Vorstellung.'
95 @UI.identification: [{ position: 80 }]
96 @UI.lineItem: [{ position: 80 }]
97 ZeitkategorieURL;
98
99@ @EndUserText.label: 'Zahlungsstatus'
100 @EndUserText.quickInfo: 'Status des Tickets (bezahllt, storniert, offen).'
101 @UI.identification: [{ position: 90 }]
102 @UI.lineItem: [{ position: 90 }]
103 ZahlungsstatusURL;
104
105
106@ // --- Zahlung (Filter) ---
107 @UI.selectionField: [{ position: 30 }]
108 @Consumption.filter.mandatory: true
109 Zahlungsart;
110
111

```

```

112@ // --- Zahlung (Foto) ---
113 @EndUserText.label: 'Zahlungsart'
114 @EndUserText.quickInfo: 'Verwendete Methode zur Bezahlung (bar, Karte).'
115 @UI.identification: [{ position: 100 }]
116 @UI.lineItem: [{ position: 100 }]
117 ZahlungsartURL;
118
119
120@ @EndUserText.label: 'Ticketpreis (€)'
121 @EndUserText.quickInfo: 'Preis des Tickets vor Abzug von Rabatten.'
122 @UI.identification: [{ position: 110 }]
123 @UI.lineItem: [{ position: 110 }]
124 TicketPreis;
125
126@ @EndUserText.label: 'Rabatt (%)'
127 @EndUserText.quickInfo: 'Angewandter Rabattbetrag oder -prozentsatz.'
128 @UI.identification: [{ position: 120 }]
129 @UI.lineItem: [{ position: 120 }]
130 Rabatt;
131
132@ @EndUserText.label: 'Betrag (€)'
133 @EndUserText.quickInfo: 'Endbetrag nach Anwendung aller Rabatte.'
134 @UI.identification: [{ position: 130 }]
135 @UI.lineItem: [{ position: 130 }]
136 Betrag;
137
138
139@ @EndUserText.label: 'Sitzplatznr.'
140 @EndUserText.quickInfo: 'Nummer des Sitzplatzes, der dem Besucher zugewiesen ist.'
141 @UI.identification: [{ position: 140 }]
142 @UI.lineItem: [{ position: 140 }]
143 TicketSitzplatznummer;
144
145@ @EndUserText.label: 'Sitzreihe'
146 @EndUserText.quickInfo: 'Reihe, in der sich der Sitzplatz des Besuchers befindet.'
147 @UI.identification: [{ position: 150 }]
148 @UI.lineItem: [{ position: 150 }]
149 BesucherSitzReihe;
150

```

```

150
151
152
153
154
155
156
157@ // -----
158 // TECHNISCHE FELDER (versteckt)
159 // -----
160 @UI.hidden: true TicketId;
161 @UI.hidden: true CreatedBy;
162 @UI.hidden: true CreatedAt;
163 @UI.hidden: true LastChangedBy;
164 @UI.hidden: true LastChangedAt;
165 @UI.hidden: true LocalLastChangedAt;
166
167@ // -----
168 // ASSOCIATIONS (versteckt)
169 // -----
170 @UI.hidden: true _Mitarbeiter;
171 @UI.hidden: true _Ticket;
172 }

```

Nach diesen Änderungen ergibt sich folgendes Endergebnis:

Search		Datum:	Besucher:	Zahlungsart:									
	All			KARTE									
<b>Verkäufe (8)</b>													
		Datum	Besucher	Erlaubnis	Besucherg...	Ticketart	Ticketgültig...	Zeitkategorie	Zahlungsart...	Zahlungsart	Ticketpreis (€)	Rabatt (%)	Betrag (€)
<input type="checkbox"/>		Oct 12, 2025	LINH NGUYEN	<input checked="" type="checkbox"/> (Ja)							18.00	10.00	16.20 >
<input type="checkbox"/>		Oct 1, 2025	JANA BECKER	<input checked="" type="checkbox"/> (Nein)							22.00	0.00	22.00 >
<input type="checkbox"/>		Oct 30, 2025	ANNA MÜLLER	<input checked="" type="checkbox"/> (Ja)							40.00	5.00	38.00 >
<input type="checkbox"/>		Oct 2, 2025	TOBIAS SCHMIDT	<input checked="" type="checkbox"/> (Ja)							15.00	0.00	15.00 >
<input type="checkbox"/>		Oct 1, 2025	ANNA MÜLLER	<input checked="" type="checkbox"/> (Ja)							20.00	15.00	17.00 >
<input type="checkbox"/>		Oct 13, 2025	OMAR ALI	<input checked="" type="checkbox"/> (Ja)							45.00	15.00	38.25 >
<input type="checkbox"/>		Oct 23, 2025	TOBIAS SCHMIDT	<input checked="" type="checkbox"/> (Ja)							20.00	20.00	16.00 >
<input type="checkbox"/>		Oct 25, 2025	MARCO ROSSI	<input checked="" type="checkbox"/> (Ja)							25.00	0.00	25.00 >

## 4.7. Definition von Kennzahlen im Kopfbereich der Object Page

Innerhalb dieser Metadata Extension wurden für die relevanten Kennzahlen TicketPreis, Rabatt und Betrag DataPoints definiert, die über Facet mit purpose: #HEADER referenziert werden. Die Facets legen die Reihenfolge der Kennzahlen im Header fest und sorgen dafür, dass diese prominent auf der Object Page angezeigt werden. Durch diese Vorgehensweise können wichtige Werte direkt im Kopfbereich der Anwendung sichtbar gemacht werden, ohne die Original-Entity zu verändern:

```

3 @UI: {
4   headerInfo: {
5     typeName: 'Verkauf',
6     typeNamePlural: 'Verkäufe',
7     title: { type: '#STANDARD',
8               value: 'MitarbeiterName'
9             },
10    description : { value: 'VerkaufId'}
11  }
12}
13

15 annotate entity ZSS25_R4C_C_VERKF_A3 with
16 {
17  // -----
18  // FACETS
19  // -----
20  @UI.facet: [
21
22    {id:'ID-2', position: 10, purpose: '#HEADER', type: '#DATAPOINT_REFERENCE', targetQualifier: 'preisWert' },
23    {id:'ID-3', position: 20, purpose: '#HEADER', type: '#DATAPOINT_REFERENCE', targetQualifier: 'rabattWert' },
24    {id:'ID-4', position: 30, purpose: '#HEADER', type: '#DATAPOINT_REFERENCE', targetQualifier: 'betragWert' }
25  ]
26
27  // 

```

Die Kritikalität des Rabatts (RabattKritikalitaet) wird in der Entität ZSS25\_R4C\_V\_VERKF\_A1 berechnet und anschließend in den Entitäten ZSS25\_R4C\_R\_VERKF\_A2 und ZSS25\_R4C\_C\_VERKF\_A3 für die Anzeige und Selektion übernommen. Dadurch kann der Wert in unterschiedlichen Layern der Anwendung konsistent genutzt werden, ohne dass die Berechnung mehrfach durchgeführt werden muss.

```

215      // RabattKritikalitaet
216 @EndUserText.label: 'Criticality für Rabatt'
217 case
218     when vkf.rabatt = 0 then 2 -- Gelb (neutral)
219     else 3                  -- Grün (positive)
220 end as RabattKritikalitaet
221
222

```

D [A07] ZSS25\_R4C\_R\_VERKF\_A2 X

```

1 @AccessControl.authorizationCheck: #NOT_REQUIRED
2 @EndUserText.label: 'Reusable View für Verkauf'
3 @Metadata.ignorePropagatedAnnotations: true
4 define view entity ZSS25_R4C_R_VERKF_A2 as select from ZSS25_R4C_V_VERKF_A1
5 association to parent ZSS25_R4C_R_MITARB_A2 as _Mitarbeiter
6   on _Mitarbeiter.MitarbeiterId = $projection.MitarbeiterId
7 association [1..1] to ZSS25_R4C_R_TICKET_A2 as _Ticket
8   on _Ticket.TicketId = $projection.TicketId
9 {
10   key VerkaufId,
11   TicketId,
12   MitarbeiterId,
13   MitarbeiterName,
14   BesucherName,
15   MitarbeiterFoto,
16   Verkaufsdatum,
17   Zahlungsart,
18   ZahlungsartURL,
19   TicketPreis,
20   TicketTyp,
21   TicketTypURL,
22   Rabatt,
23   RabattKritikalitaet,

```

```

[D] [A07] ZSS25_R4C_C_VERKF_A3 X
1@ @AccessControl.authorizationCheck: #NOT_REQUIRED
2@ @EndUserText.label: 'Consumption View für Verkauf'
3@ @Metadata.ignorePropagatedAnnotations: true
4@ @Metadata.allowExtensions: true
5@ @Search.searchable: true
6@ @ObjectModel.resultSet.sizeCategory: #XS
7@ define view entity ZSS25_R4C_C_VERKF_A3 as projection on ZSS25_R4C_R_VERKF_A2
8{
9@     @EndUserText.label: 'Rechnungsnummer'
10@    @Search.defaultSearchElement: true
11@    key VerkaufId,
12@    TicketId,
13@    MitarbeiterId,
14@    @ObjectModel.virtualElement: true
15@    MitarbeiterName,
16@    @Semantics.imageUrl: true
17@    MitarbeiterFoto,
18@    @Semantics.imageUrl: true
19@    MitarbeiterFoto,
20@    @Semantics.imageUrl: true
21@    BesucherName,
22@    Verkaufsdatum,
23@    @UI.textArrangement: #TEXT_ONLY
24@    BesucherName,
25@    Verkaufsdatum,
26@    @UI.textArrangement: #TEXT_ONLY
27@    @Consumption.valueHelpDefinition: [{entity: {name:'ZSS25_R4C_V_BESUCHNAMEN',element: 'BesucherName'} }]
28@    Zahlungsart,
29@    @Semantics.imageUrl: true
30@    ZahlungsartURL,
31@    TicketPreis,
32@    TicketTyp,
33@    @Semantics.imageUrl: true
34@    TicketTypURL,
35@    Rabatt,
36@    RabattKritikalitaet,
37@    ...
38@    ...

180@ @UI.dataPoint: { qualifier: 'preisWert', title: 'Ticketpreis (€)' }

187@ @UI.dataPoint: { qualifier: 'rabattWert', title: 'Rabatt (%)', criticality: 'RabattKritikalitaet' }

194@ @UI.dataPoint: { qualifier: 'betragWert', title: 'Betrag (€)' }

126@ @EndUserText.label: 'Ticketpreis (€)'
127@ @EndUserText.quickInfo: 'Preis des Tickets vor Abzug von Rabatten.'
128@ @UI.identification: [{ position: 110 }]
129@ @UI.lineItem: [{ position: 110 }]
130@ @UI.dataPoint: { qualifier: 'preisWert', title: 'Ticketpreis (€)' }
131@ TicketPreis;
132@ 
133@ @EndUserText.label: 'Rabatt (%)'
134@ @EndUserText.quickInfo: 'Angewandter Rabattbetrag oder -prozentsatz.'
135@ @UI.identification: [{ position: 120 }]
136@ @UI.lineItem: [{ position: 120 }]
137@ @UI.dataPoint: { qualifier: 'rabattWert', title: 'Rabatt (%)', criticality: 'RabattKritikalitaet' }
138@ Rabatt;
139@ 
140@ @EndUserText.label: 'Betrag (€)'
141@ @EndUserText.quickInfo: 'Endbetrag nach Anwendung aller Rabatte.'
142@ @UI.identification: [{ position: 130 }]
143@ @UI.lineItem: [{ position: 130 }]
144@ @UI.dataPoint: { qualifier: 'betragWert', title: 'Betrag (€)' }
145@ Betrag;

```

Das nachfolgende Ergebnis zeigt die Wirkung der durchgeföhrten Anpassungen:



Sophie Braun		
9001		
Ticketpreis (€)	Rabatt (%)	Betrag (€)
15.00	0.00	15.00



Clara Schäfer		
9008		
Ticketpreis (€)	Rabatt (%)	Betrag (€)
45.00	10.00	40.50

## 4.8. Umsetzung der UI-Gruppierung auf der Object Page

Mit `@UI.facet` wird die Struktur der Object Page festgelegt. Facets definieren Abschnitte, Unterabschnitte oder Gruppierungen von Informationen. In deinem Fall wurden vier Unterabschnitte unter „Allg. Informationen“ erstellt:

- Kauf
- Besucher
- Ticket
- Zahlung

```
{ id: 'nav-1', position: 10, type: '#COLLECTION', label: 'Allg. Informationen' },
{ id: 'nav-1-gr-1', position: 10, type: '#FIELDGROUP_REFERENCE', label: 'Kauf', parentId: 'nav-1', targetQualifier: 'grpKauf' },
{ id: 'nav-1-gr-2', position: 20, type: '#FIELDGROUP_REFERENCE', label: 'Besucher', parentId: 'nav-1', targetQualifier: 'grpBesucher' },
{ id: 'nav-1-gr-3', position: 30, type: '#FIELDGROUP_REFERENCE', label: 'Ticket', parentId: 'nav-1', targetQualifier: 'grpTicket' },
{ id: 'nav-1-gr-4', position: 40, type: '#FIELDGROUP_REFERENCE', label: 'Zahlung', parentId: 'nav-1', targetQualifier: 'grpZahlung' }]
```

Jede FieldGroup-Referenz (`targetQualifier`) bekommt die zugehörigen Felder zugeordnet. Das geschieht über `@UI.fieldGroup` mit demselben Qualifier. So erscheinen die Felder in der richtigen Gruppe auf der Object Page.

```
@UI.fieldGroup: [{ qualifier: 'grpKauf', position: 10 }]
MitarbeiterFoto;
```

Im Folgenden ist die implementierte Metadata Extension dargestellt:

```
[E] [A07] ZSS25_R4C_ME_VERKF_A6 ×
1@Metadata.layer: #CUSTOMER
2
3 @UI: {
4   headerInfo: {
5     typeName: 'Verkauf',
6     typeNamePlural: 'Verkäufe',
7     title: { type: #STANDARD, value: 'MitarbeiterName' },
8     description: { value: 'VerkaufId' }
9   }
10 }
11
12 annotate entity ZSS25_R4C_C_VERKF_A3 with
13 {
14@ // -----
15 // FACETS
16 // -----
17 @UI.facet: [
18   { id:'ID-2', position: 10, purpose: #HEADER, type: #DATAPoint_REFERENCE, targetQualifier: 'preisWert' },
19   { id:'ID-3', position: 20, purpose: #HEADER, type: #DATAPoint_REFERENCE, targetQualifier: 'rabattWert' },
20   { id:'ID-4', position: 30, purpose: #HEADER, type: #DATAPoint_REFERENCE, targetQualifier: 'betragWert' },
21
22   { id: 'nav-1', position: 10, type: #COLLECTION, label: 'Allg. Informationen' },
23
24   { id: 'nav-1-gr-1', position: 10, type: #FIELDGROUP_REFERENCE, label: 'Kauf', parentId: 'nav-1', targetQualifier: 'grpKauf' },
25   { id: 'nav-1-gr-2', position: 20, type: #FIELDGROUP_REFERENCE, label: 'Besucher', parentId: 'nav-1', targetQualifier: 'grpBesucher' },
26   { id: 'nav-1-gr-3', position: 30, type: #FIELDGROUP_REFERENCE, label: 'Ticket', parentId: 'nav-1', targetQualifier: 'grpTicket' },
27   { id: 'nav-1-gr-4', position: 40, type: #FIELDGROUP_REFERENCE, label: 'Zahlung', parentId: 'nav-1', targetQualifier: 'grpZahlung' }
28 ]
29
30
31 // -----
32 // FELDER IN SINNVOLLER REIHENFOLGE / FIELDGROUPS
33 // -----
34
35 // --- Verkauf ---
36 @EndUserText.label: 'Rechnungsnummer'
37 @EndUserText.quickInfo: 'Eindeutige Nummer der Rechnung bzw. des Verkaufs.'
38 VerkaufId;
```

```
39
40@ // --- Mitarbeiter-Foto ---
41 @EndUserText.label: ''
42 @EndUserText.quickInfo: 'Foto des Mitarbeiters, der den Verkauf durchgeführt hat.'
43 @UI.identification: [{ position: 10 }]
44 @UI.lineItem: [{ position: 10 }]
45 @UI.fieldGroup: [{ qualifier: 'grpKauf', position: 10 }]
46 MitarbeiterFoto;
47
48@ @EndUserText.label: 'Datum'
49 @EndUserText.quickInfo: 'Datum, an dem der Verkauf stattgefunden hat.'
50 @UI.selectionField: [{ position: 10 }]
51 @UI.identification: [{ position: 20 }]
52 @UI.lineItem: [{ position: 20 }]
53 @Consumption.filter: { selectionType: #INTERVAL, multipleSelections: false }
54 @UI.fieldGroup: [{ qualifier: 'grpKauf', position: 20 }]
55 Verkaufsdatum;
56
57@ @EndUserText.label: 'Mitarbeiter-ID'
58 @EndUserText.quickInfo: 'Eindeutige Kennung des Mitarbeiters (internes Feld).'
59 @UI.hidden: true
60 MitarbeiterId;
61
62
63@ // --- Besucher ---
64 @EndUserText.label: 'Besucher'
65 @EndUserText.quickInfo: 'Hier werden die Namen aller Besucher angezeigt.'
66 @UI.identification: [{ position: 30 }]
67 @UI.lineItem: [{ position: 30 }]
68 @UI.selectionField: [{ position: 20 }]
69 @UI.fieldGroup: [{ qualifier: 'grpBesucher', position: 10 }]
70 BesucherName;
71
72@ @EndUserText.label: 'Erlaubnis'
73 @EndUserText.quickInfo: 'Zeigt an, ob der Besucher zum Eintritt berechtigt ist.'
74 @UI.identification: [{ position: 40 }]
75 @UI.lineItem: [{ position: 40, criticality: 'ErlaubnisKritikalitaet' }]
76 @UI.fieldGroup: [{ qualifier: 'grpBesucher', position: 20 }]
77 BesucherIstErlaubt;
```

```

78@ @EndUserText.label: 'Besuchergruppe'
79@ @EndUserText.quickInfo: 'Gibt die Altersgruppe des Besuchers an.'
80@ @UI.identification: [{ position: 50 }]
81@ @UI.lineItem: [{ position: 50 }]
82@ @UI.fieldGroup: [{ qualifier: 'grpBesucher', position: 30 }]
83@ BesucherAltersgruppeURL;
84
85
86@ // --- Ticket ---
87@ @EndUserText.label: 'Ticketart'
88@ @EndUserText.quickInfo: 'Art bzw. Typ des ausgestellten Tickets.'
89@ @UI.identification: [{ position: 60 }]
90@ @UI.lineItem: [{ position: 60 }]
91@ @UI.fieldGroup: [{ qualifier: 'grpTicket', position: 10 }]
92@ TicketTypURL;
93
94@ @EndUserText.label: 'Ticketgültigkeit'
95@ @EndUserText.quickInfo: 'Gibt an, ob das Ticket aktuell gültig ist.'
96@ @UI.identification: [{ position: 70 }]
97@ @UI.lineItem: [{ position: 70 }]
98@ @UI.fieldGroup: [{ qualifier: 'grpTicket', position: 20 }]
99@ TicketIstGueltigURL;
100
101@ @EndUserText.label: 'Sitzplatznr.'
102@ @EndUserText.quickInfo: 'Nummer des Sitzplatzes, der dem Besucher zugewiesen ist.'
103@ @UI.identification: [{ position: 140 }]
104@ @UI.lineItem: [{ position: 140 }]
105@ @UI.fieldGroup: [{ qualifier: 'grpTicket', position: 30 }]
106@ TicketSitzplatznummer;
107
108@ @EndUserText.label: 'Sitzreihe'
109@ @EndUserText.quickInfo: 'Reihe, in der sich der Sitzplatz des Besuchers befindet.'
110@ @UI.identification: [{ position: 150 }]
111@ @UI.lineItem: [{ position: 150 }]
112@ @UI.fieldGroup: [{ qualifier: 'grpTicket', position: 40 }]
113@ BesucherSitzReihe;
114
115
116

```

```

117@ // --- Zahlung ---
118@ @EndUserText.label: 'Zeitkategorie'
119@ @EndUserText.quickInfo: 'Zeitliche Kategorie der Vorstellung.'
120@ @UI.identification: [{ position: 80 }]
121@ @UI.lineItem: [{ position: 80 }]
122@ @UI.fieldGroup: [{ qualifier: 'grpZahlung', position: 10 }]
123@ ZeitkategorieURL;
124
125@ @EndUserText.label: 'Zahlungsstatus'
126@ @EndUserText.quickInfo: 'Status des Tickets (bezahlt, storniert, offen).'
127@ @UI.identification: [{ position: 90 }]
128@ @UI.lineItem: [{ position: 90 }]
129@ @UI.fieldGroup: [{ qualifier: 'grpZahlung', position: 20 }]
130@ ZahlungsstatusURL;
131
132@ @UI.selectionField: [{ position: 30 }]
133@ @Consumption.filter.mandatory: true
134@ Zahlungsaart;
135
136@ @EndUserText.label: 'Zahlungsart'
137@ @EndUserText.quickInfo: 'Verwendete Methode zur Bezahlung (bar, Karte).'
138@ @UI.identification: [{ position: 100 }]
139@ @UI.lineItem: [{ position: 100 }]
140@ @UI.dataPoint: { qualifier: 'zahlungsartFoto', title: 'Zahlungsart' }
141@ @UI.fieldGroup: [{ qualifier: 'grpZahlung', position: 30 }]
142@ ZahlungsartURL;
143
144@ @EndUserText.label: 'Ticketpreis (€)'
145@ @EndUserText.quickInfo: 'Preis des Tickets vor Abzug von Rabatten.'
146@ @UI.identification: [{ position: 110 }]
147@ @UI.lineItem: [{ position: 110 }]
148@ @UI.dataPoint: { qualifier: 'preisWert', title: 'Ticketpreis (€)' }
149@ TicketPreis;
150
151@ @EndUserText.label: 'Rabatt (%)'
152@ @EndUserText.quickInfo: 'Angewandter Rabattbetrag oder -prozentsatz.'
153@ @UI.identification: [{ position: 120 }]
154@ @UI.lineItem: [{ position: 120 }]
155@ @UI.dataPoint: { qualifier: 'rabattWert', title: 'Rabatt (%)', criticality: 'RabattKritikalitaet' }
156@ Rabatti;
157

```

```

158@ @EndUserText.label: 'Betrag (€)'
159@ @EndUserText.quickInfo: 'Endbetrag nach Anwendung aller Rabatte.'
160@ @UI.identification: [{ position: 130 }]
161@ @UI.lineItem: [{ position: 130 }]
162@ @UI.dataPoint: { qualifier: 'betragWert', title: 'Betrag (€)' }
163@ Betrag;
164
165
166@ // -----
167@ // TECHNISCHE FELDER (versteckt)
168@ // -----
169@ @UI.hidden: true TicketId;
170@ @UI.hidden: true CreatedBy;
171@ @UI.hidden: true CreatedAt;
172@ @UI.hidden: true LastChangedBy;
173@ @UI.hidden: true LastChangedAt;
174@ @UI.hidden: true LocalLastChangedAt;
175@ @UI.hidden: true _Mitarbeiter;
176@ @UI.hidden: true _Ticket;
177}

```

Das nachfolgende Ergebnis zeigt die Wirkung der durchgeföhrten Anpassungen:

The screenshot shows a purchase summary for Maximilian Hoffmann (ID 9002). At the top, it displays the ticket price (20.00), discount (25.00), and amount (15.00). Below this, there are four main sections: Kauf (Purchase), Besucher (Visitor), Ticket, and Zahlung (Payment). Each section contains icons and specific data points. A 'Delete' button is located in the top right corner.

Kauf	Besucher	Ticket	Zahlung
	<b>Besucher:</b> TOBIAS SCHMIDT <b>Erlaubnis:</b> (Ja) <b>Besucherguppe:</b>	<b>Ticketart:</b> <b>Sitzplatznr.:</b> 32 <b>Sitzreihe:</b> 4	<b>Zeitkategorie:</b> <b>Zahlungsstatus:</b> <b>Zahlungsart:</b>

## 4.9. Umsetzung mehrerer UI-Gruppierungen auf der Object Page

Es werden neue Gruppen erstellt. Im Folgenden ist die implementierte Metadata Extension dargestellt:

```

1@ [A07] ZSS25_R4C_ME_VERKF_A6 ×
1@ @Metadata.layer: #CUSTOMER
2
3 @UI: {
4   headerInfo: {
5     typeName: 'Verkauf',
6     typeNamePlural: 'Verkäufe',
7     title: { type: #STANDARD, value: 'MitarbeiterName' },
8     description: { value: 'VerkaufId' }
9   }
10 }
11
12 annotate entity ZSS25_R4C_C_VERKF_A3 with
13 {
14@ // -----
15 // FACETS
16 // -----
17 @UI.facet: [
18   { id:'ID-2', position: 10, purpose: #HEADER, type: #DATAPoint_REFERENCE, targetQualifier: 'preisWert' },
19   { id:'ID-3', position: 20, purpose: #HEADER, type: #DATAPoint_REFERENCE, targetQualifier: 'rabattWert' },
20   { id:'ID-4', position: 30, purpose: #HEADER, type: #DATAPoint_REFERENCE, targetQualifier: 'betragWert' },
21
22   // Gruppe: Allgemeine Informationen
23   { id: 'nav-1', position: 10, type: #COLLECTION, label: 'Allg. Informationen' },
24   { id: 'nav-1-gr-1', position: 10, type: #FIELDGROUP_REFERENCE, label: 'Kauf', parentId: 'nav-1', targetQualifier: 'grpKauf' },
25   { id: 'nav-1-gr-2', position: 20, type: #FIELDGROUP_REFERENCE, label: 'Besucher', parentId: 'nav-1', targetQualifier: 'grpBesucher' },
26   { id: 'nav-1-gr-3', position: 30, type: #FIELDGROUP_REFERENCE, label: 'Ticket', parentId: 'nav-1', targetQualifier: 'grpTicket' },
27   { id: 'nav-1-gr-4', position: 40, type: #FIELDGROUP_REFERENCE, label: 'Zahlung', parentId: 'nav-1', targetQualifier: 'grpZahlung' },
28 }
```

```

28 // Gruppe: Vorstellung
29 { id: 'nav-2', position: 20, type: #COLLECTION, label: 'Vorstellung' },
30 { id: 'nav-2-gr-1', position: 10, type: #FIELDGROUP_REFERENCE, label: 'Vorstellungsdatum', parentId: 'nav-2', targetQualifier: 'grpVorstellungsdatum' }
31 { id: 'nav-2-gr-2', position: 20, type: #FIELDGROUP_REFERENCE, label: 'Vorstellungszeit', parentId: 'nav-2', targetQualifier: 'grpVorstellungszeit' }
32 { id: 'nav-2-gr-3', position: 30, type: #FIELDGROUP_REFERENCE, label: 'Sprache & Untertitel', parentId: 'nav-2', targetQualifier: 'grpSpracheUntertitel' }
33 { id: 'nav-2-gr-4', position: 40, type: #FIELDGROUP_REFERENCE, label: 'Format', parentId: 'nav-2', targetQualifier: 'grpFormat' }
34 { id: 'nav-2-gr-5', position: 50, type: #FIELDGROUP_REFERENCE, label: 'Status', parentId: 'nav-2', targetQualifier: 'grpStatus' }
35
36 // Gruppe: Film
37 { id: 'nav-3', position: 30, type: #COLLECTION, label: 'Film' },
38 { id: 'nav-3-gr-1', position: 10, type: #FIELDGROUP_REFERENCE, label: 'Titel & Genre', parentId: 'nav-3', targetQualifier: 'grpTitelGenre' }
39 { id: 'nav-3-gr-2', position: 20, type: #FIELDGROUP_REFERENCE, label: 'Dauer & Altersfreigabe', parentId: 'nav-3', targetQualifier: 'grpDauerAltersfreigabe' }
40 { id: 'nav-3-gr-3', position: 30, type: #FIELDGROUP_REFERENCE, label: 'Produktionsdetails', parentId: 'nav-3', targetQualifier: 'grpProduktionsdetails' }
41 { id: 'nav-3-gr-4', position: 40, type: #FIELDGROUP_REFERENCE, label: 'Regisseur', parentId: 'nav-3', targetQualifier: 'grpRegisseur' }
42
43
44
45
46 // Gruppe: Verkäufer
47 { id: 'nav-4', position: 40, type: #COLLECTION, label: 'Verkäufer' },
48 { id: 'nav-4-gr-1', position: 10, type: #FIELDGROUP_REFERENCE, label: 'Allgemeine Informationen', parentId: 'nav-4', targetQualifier: 'grpVerkaeuferAllgemeineInformationen' }
49 { id: 'nav-4-gr-2', position: 20, type: #FIELDGROUP_REFERENCE, label: 'Rolle & Abteilung', parentId: 'nav-4', targetQualifier: 'grpVerkaeuferRolleAbteilung' }
50 { id: 'nav-4-gr-3', position: 30, type: #FIELDGROUP_REFERENCE, label: 'Gehalt', parentId: 'nav-4', targetQualifier: 'grpVerkaeuferGehalt' }
51 { id: 'nav-4-gr-4', position: 40, type: #FIELDGROUP_REFERENCE, label: 'Kontakt', parentId: 'nav-4', targetQualifier: 'grpVerkaeuferKontakt' }
52
53
54

```

```

54 // Gruppe: Kino
55 { id: 'nav-5', position: 50, type: #COLLECTION, label: 'Kino' },
56 { id: 'nav-5-gr-1', position: 10, type: #FIELDGROUP_REFERENCE, label: 'Allgemeine Informationen', parentId: 'nav-5', targetQualifier: 'grpKinoAllgemeineInformationen' }
57 { id: 'nav-5-gr-2', position: 20, type: #FIELDGROUP_REFERENCE, label: 'Adresse', parentId: 'nav-5', targetQualifier: 'grpKinoAdresse' }
58 { id: 'nav-5-gr-3', position: 30, type: #FIELDGROUP_REFERENCE, label: 'Einrichtung', parentId: 'nav-5', targetQualifier: 'grpKinoEinrichtung' }
59 { id: 'nav-5-gr-4', position: 40, type: #FIELDGROUP_REFERENCE, label: 'Kontakt', parentId: 'nav-5', targetQualifier: 'grpKinoKontakt' }
60
61 // Gruppe: Raum
62 { id: 'nav-6', position: 60, type: #COLLECTION, label: 'Raum' },
63 { id: 'nav-6-gr-1', position: 10, type: #FIELDGROUP_REFERENCE, label: 'Allgemeine Informationen', parentId: 'nav-6', targetQualifier: 'grpRaumAllgemeineInformationen' }
64 { id: 'nav-6-gr-2', position: 20, type: #FIELDGROUP_REFERENCE, label: 'Ausstattung', parentId: 'nav-6', targetQualifier: 'grpRaumAusstattung' }
65
66 // Gruppe: Besucher
67 { id: 'nav-7', position: 70, type: #COLLECTION, label: 'Besucher' },
68 { id: 'nav-7-gr-1', position: 10, type: #FIELDGROUP_REFERENCE, label: 'Persönliche Daten', parentId: 'nav-7', targetQualifier: 'grpBesucherPersoenlicheDaten' }
69 { id: 'nav-7-gr-2', position: 20, type: #FIELDGROUP_REFERENCE, label: 'Kontaktdaten', parentId: 'nav-7', targetQualifier: 'grpBesucherKontaktdaten' }
70 { id: 'nav-7-gr-3', position: 30, type: #FIELDGROUP_REFERENCE, label: 'Registrierungsdetails', parentId: 'nav-7', targetQualifier: 'grpBesucherRegistrierungsdetails' }
71
72 // Gruppe: Ticket
73 { id: 'nav-8', position: 80, type: #COLLECTION, label: 'Ticket' },
74 { id: 'nav-8-gr-1', position: 10, type: #FIELDGROUP_REFERENCE, label: 'Ticket-URL', parentId: 'nav-8', targetQualifier: 'grpTicketQrCodeLink' }
75 { id: 'nav-8-gr-2', position: 20, type: #FIELDGROUP_REFERENCE, label: 'QR-Code', parentId: 'nav-8', targetQualifier: 'grpTicketQrCode' }
76 { id: 'nav-8-gr-3', position: 30, type: #FIELDGROUP_REFERENCE, label: 'QR-Code', parentId: 'nav-8', targetQualifier: 'grpTicketDQrCode' }
77 ]
78
79
80
81 // -----

```

```

80
81 // -----
82 // FELDER IN SINNVOLLER REIHENFOLGE / FIELDGROUPS
83 // -----
84
85 // --- Verkauf ---
86 @EndUserText.label: 'Rechnungsnummer'
87 @EndUserText.quickInfo: 'Eindeutige Nummer der Rechnung bzw. des Verkaufs.'
88 VerkaufId;
89
90// --- Mitarbeiter-Foto ---
91 @EndUserText.label: 'Mitarbeiter'
92 @EndUserText.quickInfo: 'Foto des Mitarbeiters, der den Verkauf durchgeführt hat.'
93 @UI.identification: [{ position: 10 }]
94 @UI.lineItem: [{ position: 10 }]
95 @UI.fieldGroup: [{ qualifier: 'grpKauf', position: 10 }]
96 MitarbeiterFoto;
97
98@EndUserText.label: 'Datum'
99 @EndUserText.quickInfo: 'Datum, an dem der Verkauf stattgefunden hat.'
00 @UI.selectionField: [{ position: 10 }]
01 @UI.identification: [{ position: 20 }]
02 @UI.lineItem: [{ position: 20 }]
03 @Consumption.filter: { selectionType: #INTERVAL, multipleSelections: false }
04 @UI.fieldGroup: [{ qualifier: 'grpKauf', position: 20 }]
05 Verkaufsdatum;
06
07

```

```

108@ // --- Besucher ---
109 @EndUserText.label: 'Besucher'
110 @EndUserText.quickInfo: 'Hier werden die Namen aller Besucher angezeigt.'
111 @UI.identification: [{ position: 30 }]
112 @UI.lineItem: [{ position: 30 }]
113 @UI.selectionField: [{ position: 20 }]
114 @UI.fieldGroup: [{ qualifier: 'grpBesucher', position: 10 }]
115 BesucherName;
116
117@ @EndUserText.label: 'Erlaubnis'
118 @EndUserText.quickInfo: 'Zeigt an, ob der Besucher zum Eintritt berechtigt ist.'
119 @UI.identification: [{ position: 40 }]
120 @UI.lineItem: [{ position: 40, criticality: 'ErlaubnisKritikalitaet' }]
121 @UI.fieldGroup: [{ qualifier: 'grpBesucher', position: 20 }]
122 BesucherIstErlaubt;
123
124@ @EndUserText.label: 'Besuchergruppe'
125 @EndUserText.quickInfo: 'Gibt die Altersgruppe des Besuchers an.'
126 @UI.identification: [{ position: 50 }]
127 @UI.lineItem: [{ position: 50 }]
128 @UI.fieldGroup: [{ qualifier: 'grpBesucher', position: 30 }]
129 BesucherAltersgruppeURL;
130
131

```

```

51
52@ // --- Ticket ---
53 @EndUserText.label: 'Ticketart'
54 @EndUserText.quickInfo: 'Art bzw. Typ des ausgestellten Tickets.'
55 @UI.identification: [{ position: 60 }]
56 @UI.lineItem: [{ position: 60 }]
57 @UI.fieldGroup: [{ qualifier: 'grpTicket', position: 10 }]
58 TicketTypURL;
59
60@ @EndUserText.label: 'Ticketgültigkeit'
61 @EndUserText.quickInfo: 'Gibt an, ob das Ticket aktuell gültig ist.'
62 @UI.identification: [{ position: 70 }]
63 @UI.lineItem: [{ position: 70 }]
64 @UI.fieldGroup: [{ qualifier: 'grpTicket', position: 20 }]
65 TicketIstGueltigURL;
66
67@ @EndUserText.label: 'Sitzplatznr.'
68 @EndUserText.quickInfo: 'Nummer des Sitzplatzes, der dem Besucher zugewiesen ist.'
69 @UI.identification: [{ position: 140 }]
70 @UI.lineItem: [{ position: 140 }]
71 @UI.fieldGroup: [{ qualifier: 'grpTicket', position: 30 }]
72 TicketSitzplatznummer;
73

```

```

153
154@ @EndUserText.label: 'Sitzreihe'
155 @EndUserText.quickInfo: 'Reihe, in der sich der Sitzplatz des Besuchers befindet.'
156 @UI.identification: [{ position: 150 }]
157 @UI.lineItem: [{ position: 150 }]
158 @UI.fieldGroup: [{ qualifier: 'grpTicket', position: 40 }]
159 BesucherSitzReihe;
160
161
162@ // --- Zahlung ---
163 @EndUserText.label: 'Zeitkategorie'
164 @EndUserText.quickInfo: 'Zeitliche Kategorie der Vorstellung.'
165 @UI.identification: [{ position: 80 }]
166 @UI.lineItem: [{ position: 80 }]
167 @UI.fieldGroup: [{ qualifier: 'grpZahlung', position: 10 }]
168 ZeitkategorieURL;
169
170@ @EndUserText.label: 'Zahlungsstatus'
171 @EndUserText.quickInfo: 'Status des Tickets (bezahlt, storniert, offen).'
172 @UI.identification: [{ position: 90 }]
173 @UI.lineItem: [{ position: 90 }]
174 @UI.fieldGroup: [{ qualifier: 'grpZahlung', position: 20 }]
175 ZahlungsstatusURL;
176
177@ @UI.selectionField: [{ position: 30 }]
178 @Consumption.filter.mandatory: true
179 Zahlungsart;

```

```

180
181@ @EndUserText.label: 'Zahlungsart'
182 @EndUserText.quickInfo: 'Verwendete Methode zur Bezahlung (bar, Karte).'
183 @UI.identification: [{ position: 100 }]
184 @UI.lineItem: [{ position: 100 }]
185 @UI.dataPoint: { qualifier: 'zahlungsartFoto', title: 'Zahlungsart' }
186 @UI.fieldGroup: [{ qualifier: 'grpZahlung', position: 30 }]
187 ZahlungsartURL;
188
189@ @EndUserText.label: 'Ticketpreis (€)'
190 @EndUserText.quickInfo: 'Preis des Tickets vor Abzug von Rabatten.'
191 @UI.identification: [{ position: 110 }]
192 @UI.lineItem: [{ position: 110 }]
193 @UI.dataPoint: { qualifier: 'preisWert', title: 'Ticketpreis (€)' }
194 TicketPreis;
195
196@ @EndUserText.label: 'Rabatt (%)'
197 @EndUserText.quickInfo: 'Angewandter Rabattbetrag oder -prozentsatz.'
198 @UI.identification: [{ position: 120 }]
199 @UI.lineItem: [{ position: 120 }]
200 @UI.dataPoint: { qualifier: 'rabattWert', title: 'Rabatt (%)', criticality: 'RabattKritikalitaet' }
201 Rabatt;
202

```

```

202
203@ @EndUserText.label: 'Betrag (€)'
204 @EndUserText.quickInfo: 'Endbetrag nach Anwendung aller Rabatte.'
205 @UI.identification: [{ position: 130 }]
206 @UI.lineItem: [{ position: 130 }]
207 @UI.dataPoint: { qualifier: 'betragWert', title: 'Betrag (€)' }
208 Betrag;
209
210
211@ // =====
212 // Vorstellung
213 // =====
214 @UI.lineItem: [{ position: 140 }]
215 @UI.fieldGroup: [{ qualifier: 'grpVorstellungsDatum', position: 10 }]
216 Vorstellungsdatum;
217
218@ @UI.lineItem: [{ position: 150 }]
219 @UI.fieldGroup: [{ qualifier: 'grpVorstellungsZeit', position: 10 }]
220 VorstellungStartzeit;
221
222@ @UI.lineItem: [{ position: 160 }]
223 @UI.fieldGroup: [{ qualifier: 'grpVorstellungsZeit', position: 20 }]
224 VorstellungEndzeit;
225
226@ @UI.lineItem: [{ position: 170 }]
227 @UI.fieldGroup: [{ qualifier: 'grpVorstellungsSpracheUntertitel', position: 10 }]
228 VorstellungSprache;

```

```

229
230@ @UI.lineItem: [{ position: 180 }]
231 @UI.fieldGroup: [{ qualifier: 'grpVorstellungsSpracheUntertitel', position: 20 }]
232 VorstellungUntertitel;
233
234@ @UI.lineItem: [{ position: 190 }]
235 @UI.fieldGroup: [{ qualifier: 'grpVorstellungsFormat', position: 10 }]
236 VorstellungFormat;
237
238@ @UI.lineItem: [{ position: 200 }]
239 @UI.fieldGroup: [{ qualifier: 'grpVorstellungsStatus', position: 10 }]
240 VorstellungStatus;
241
242@ // =====
243 // Film
244 // =====
245 @UI.lineItem: [{ position: 210 }]
246 @UI.fieldGroup: [{ qualifier: 'grpFilmTitelGenre', position: 10 }]
247 FilmTitel;
248
249@ @UI.lineItem: [{ position: 220 }]
250 @UI.fieldGroup: [{ qualifier: 'grpFilmTitelGenre', position: 20 }]
251 FilmGenre;
252
253@ @UI.lineItem: [{ position: 230 }]
254 @UI.fieldGroup: [{ qualifier: 'grpFilmDauerAltersfreigabe', position: 10 }]
255 FilmDauerMinuten;
256

```

```

250
257@ @UI.lineItem: [{ position: 240 }]
258 @UI.fieldGroup: [{ qualifier: 'grpFilmDauerAltersfreigabe', position: 20 }]
259 FilmAltersfreigabe;
260
261@ @UI.lineItem: [{ position: 250 }]
262 @UI.fieldGroup: [{ qualifier: 'grpFilmProduktion', position: 10 }]
263 FilmProduktionsjahr;
264
265@ @UI.lineItem: [{ position: 260 }]
266 @UI.fieldGroup: [{ qualifier: 'grpFilmRegisseur', position: 10 }]
267 FilmRegisseur;
268
269
270
271@ // =====
272 // Mitarbeiter
273 // =====
274
275 @UI.lineItem: [{ position: 270 }]
276 @UI.fieldGroup: [{ qualifier: 'grpVerkaeuferAllgemein', position: 10 }]
277 MitarbeiterId;
278
279@ @UI.lineItem: [{ position: 280 }]
280 @UI.fieldGroup: [{ qualifier: 'grpVerkaeuferAllgemein', position: 20 }]
281 MitarbeiterName;
282

```

```

282
283@ @UI.lineItem: [{ position: 290 }]
284 @UI.fieldGroup: [{ qualifier: 'grpVerkaeuferAllgemein', position: 30 }]
285 MitarbeiterGeburtsdatum;
286
287@ @UI.lineItem: [{ position: 300 }]
288 @UI.fieldGroup: [{ qualifier: 'grpVerkaeuferRolleAbteilung', position: 10 }]
289 MitarbeiterRolle;
290
291@ @UI.lineItem: [{ position: 310 }]
292 @UI.fieldGroup: [{ qualifier: 'grpVerkaeuferRolleAbteilung', position: 20 }]
293 MitarbeiterAbteilung;
294
295@ @UI.lineItem: [{ position: 320 }]
296 @UI.fieldGroup: [{ qualifier: 'grpVerkaeuferGehalt', position: 10 }]
297 MitarbeiterGehalt;
298
299@ @UI.lineItem: [{ position: 330 }]
300 @UI.fieldGroup: [{ qualifier: 'grpMitarbeiterKontakt', position: 10 }]
301 MitarbeiterKontaktinfo;
302
303
304@ // --- Gruppe: Kino ---
305 @UI.lineItem: [{ position: 340 }]
306 @UI.fieldGroup: [{ qualifier: 'grpKinoAllgemein', position: 10 }]
307 KinoName;

```

```

308
309@ @UI.lineItem: [{ position: 350 }]
310 @UI.fieldGroup: [{ qualifier: 'grpKinoAllgemein', position: 20 }]
311 KinoEroeffnungsdatum;
312
313@ @UI.lineItem: [{ position: 360 }]
314 @UI.fieldGroup: [{ qualifier: 'grpKinoAdresse', position: 10 }]
315 KinoAdresse;
316
317@ @UI.lineItem: [{ position: 370 }]
318 @UI.fieldGroup: [{ qualifier: 'grpKinoAdresse', position: 20 }]
319 KinoPlz;
320
321@ @UI.lineItem: [{ position: 380 }]
322 @UI.fieldGroup: [{ qualifier: 'grpKinoAdresse', position: 30 }]
323 KinoStadt;
324
325@ @UI.lineItem: [{ position: 390 }]
326 @UI.fieldGroup: [{ qualifier: 'grpKinoEinrichtung', position: 10 }]
327 KinoAnzahlRaeume;
328
329@ @UI.lineItem: [{ position: 400 }]
330 @UI.fieldGroup: [{ qualifier: 'grpKinoKontakt', position: 10 }]
331 KinoKontaktinfo;
332

```

```

333
334@ // --- Gruppe: Raum
335 @UI.lineItem: [{ position: 410 }]
336 @UI.fieldGroup: [{ qualifier: 'grpRaumAllgemein', position: 10 }]
337 RaumName;
338
339@ @UI.lineItem: [{ position: 420 }]
340 @UI.fieldGroup: [{ qualifier: 'grpRaumAllgemein', position: 20 }]
341 RaumSitzplaetze;
342
343@ @UI.lineItem: [{ position: 430 }]
344 @UI.fieldGroup: [{ qualifier: 'grpRaumAusstattung', position: 10 }]
345 RaumAusstattung;
346
347
348
349@ // --- Gruppe: Besucher ---
350 @UI.lineItem: [{ position: 440 }]
351 @UI.fieldGroup: [{ qualifier: 'grpBesucherPerson', position: 10 }]
352 BesucherGeburtsdatum;
353
354@ @UI.lineItem: [{ position: 450 }]
355 @UI.fieldGroup: [{ qualifier: 'grpBesucherPerson', position: 20 }]
356 BesucherAlter;
357
358@ @UI.lineItem: [{ position: 460 }]
359 @UI.fieldGroup: [{ qualifier: 'grpBesucherKontakt', position: 10 }]
360 BesucherEmail;

```

```

361
362@ @UI.lineItem: [{ position: 470 }]
363 @UI.fieldGroup: [{ qualifier: 'grpBesucherKontakt', position: 20 }]
364 BesucherTelefon;
365
366@ @UI.lineItem: [{ position: 480 }]
367 @UI.fieldGroup: [{ qualifier: 'grpBesucherKontakt', position: 30 }]
368 BesucherAdresse;
369
370@ @UI.lineItem: [{ position: 490 }]
371 @UI.fieldGroup: [{ qualifier: 'grpBesucherRegistrierung', position: 10 }]
372 BesucherRegistrierung;
373
374
375@ // --- Gruppe: Ticket ---
376 @UI.lineItem: [{ position: 500 }]
377 @UI.fieldGroup: [{ qualifier: 'grpTicketQrCode', position: 10}]
378 TicketQrCode;
379
380@ @UI.lineItem: [{ position: 510 }]
381 @UI.fieldGroup: [{ qualifier: 'grpTicketCQrCodeBild', position: 10 }]
382 TicketQrCodeCFoto;
383
384@ @UI.lineItem: [{ position: 520 }]
385 @UI.fieldGroup: [{ qualifier: 'grpTicketDQrCodeBild', position: 10 }]
386 TicketQrCodeDFoto;
387

```

```

387
388
389@ // -----
390 // TECHNISCHE FELDER (versteckt)
391 //
392 @UI.hidden: true TicketId;
393 @UI.hidden: true CreatedBy;
394 @UI.hidden: true CreatedAt;
395 @UI.hidden: true LastChangedBy;
396 @UI.hidden: true LastChangedAt;
397 @UI.hidden: true LocalLastChangedAt;
398 @UI.hidden: true _Mitarbeiter;
399 @UI.hidden: true _Ticket;
400 }

```

Das nachfolgende Ergebnis zeigt die Wirkung der durchgeföhrten Anpassungen:

Search		Datum:	Besucher:	Zahlungsart:	Go							
		All		BAR								
<b>Verkäufe (15)</b>												
	Datum	Besucher	Erlaubnis	Besucherg...	Ticketart	Ticketgülti...	Zeitkategorie	Zahlungsst...	Zahlungsart	Ticketpreis (€)	Rabatt (%)	Betrag (€)
<input type="checkbox"/>	Oct 11, 2025	ANNA MÜLLER	(Ja)							15.00	0.00	15.00
<input type="checkbox"/>	Oct 15, 2025	TOBIAS SCHMIDT	(Ja)							20.00	25.00	15.00
<input type="checkbox"/>	Oct 14, 2025	OMAR ALI	(Ja)							35.00	0.00	35.00
<input type="checkbox"/>	Oct 12, 2025	SOPHIE WAGNER	(Ja)							35.00	5.00	33.25
<input type="checkbox"/>	Oct 27, 2025	MARCO ROSSI	(Ja)							15.00	0.00	15.00

Maximilian Hoffmann  
9002

Ticketpreis (€)	Rabatt (%)	Betrag (€)
20.00	25.00	15.00

[Delete](#) [Ctrl+Delete](#)

[Allg. Informationen](#) [Vorstellung](#) [Film](#) [Verkäufer](#) [Kino](#) [Raum](#) [Besucher](#) [Ticket](#)

<b>Kauf</b>	<b>Besucher</b>	<b>Ticket</b>	<b>Zahlung</b>
Mitarbeiter: 	Besucher: TOBIAS SCHMIDT	Ticketart: 	Zeitkategorie: 
Datum: Oct 15, 2025	Erlaubnis: (Ja)	Sitzplatznr.: 32	Zahlungsstatus: 
	Besuchergruppe: 	Sitzreihe: 4	Zahlungsart: 

Vorstellung					
Vorstellungsdatum	Vorstellungszeit	Sprache & Untertitel	Format	Status	
Nov 9, 2025	2:00:00 PM	6:00:00 PM	DE	3D	ABGESAGT
VORHANDEN					
Film					
Titel & Genre		Dauer & Altersfreigabe	Produktionsdetails	Regisseur	
INCEPTION	SCI-FI	180 12	2016	RAFAEL MARTINEZ	

Verkäufer					
Allgemeine Informationen	Adresse	Rolle & Abteilung	Gehalt	Kontakt	
5005	Maximilian Hoffmann	May 6, 2001	Schichtleiter/in VKF2	3,150.00	
Kino					
Allgemeine Informationen	Adresse	Einrichtung	Kontakt		
RAP4YOU	POTSDAMER PLATZ 1	10785	BAD HOMBURG	5	01514 1223334
Dec 1, 2001					

Raum			
Allgemeine Informationen		Ausstattung	
SAAL 1	120	3D, DOLBY SURROUND	
Besucher			
Persönliche Daten		Kontaktdaten	Registrierungsdetails
Jul 22, 2010	15	T.SCHMIDT@DUMMYMAIL.DE +49 176 996877	LINDENWEG 5, 50667 KÖLN Jul 25, 2012

Ticket			
Ticket-URL	QR-Code	QR-Code	
HTTPS://GITHUB.COM/KAMAL-BADAWI?TAB=REPOSITORIES			

## 5. Auswertungen und Kennzahlen

### 5.1. Verkaufsleistung der Mitarbeiter

In dieser CDS-View (ZSS25\_R4C\_V\_TOPVERKFMITARB) wird die Summe der Verkaufsbeträge pro Mitarbeiter berechnet. Dazu werden alle Datensätze aus der Quell-View ZSS25\_R4C\_V\_VERKF\_A1 berücksichtigt, bei denen der Ticketstatus nicht „STORNIERT“ ist.

```
D [A07] ZSS25_R4C_V_TOPVERKFMITARB ×
1④ @AbapCatalog.viewEnhancementCategory: [#NONE]
2 @AccessControl.authorizationCheck: #NOT_REQUIRED
3 @EndUserText.label: 'View für Top Mitarbeiter'
4 @Metadata.ignorePropagatedAnnotations: true
5 @ObjectModel.usageType:{
6   serviceQuality: #X,
7   sizeCategory: #S,
8   dataClass: #MIXED
9 }
10 define view entity ZSS25_R4C_V_TOPVERKFMITARB as
11   select from ZSS25_R4C_V_VERKF_A1
12 {
13   key MitarbeiterId,
14   MitarbeiterName,
15   sum( Betrag ) as Gesamtbetrag
16 }
17 where TicketStatus <> 'STORNIERT'
18 group by MitarbeiterId, MitarbeiterName
19
```

Für Top-Mitarbeiter und deren Foto-URLs werden die neue CDS-, Reusable-, Consumption-Views, Service Definition, Binding und Metadata Extension erstellt:

```
D [A07] ZSS25_R4C_V_TOPVERKFMITARB_A1 ×
1 @AbapCatalog.viewEnhancementCategory: [#NONE]
2 @AccessControl.authorizationCheck: #NOT_REQUIRED
3 @EndUserText.label: 'View für Top Mitarbeiter'
4 @Metadata.ignorePropagatedAnnotations: true
5 @ObjectModel.usageType:{ ...
6   serviceQuality: #X,
7   sizeCategory: #S,
8   dataClass: #MIXED
9 }
10 define view entity ZSS25_R4C_V_TOPVERKFMITARB_A1 as
11   select from ZSS25_R4C_V_TOPVERKFMITARB as top
12 left outer join ZSS25_R4C_V_MITARB_A1 as ma on ma.MitarbeiterId = top.MitarbeiterId
13
14 {
15   key top.MitarbeiterId,
16   top.MitarbeiterName,
17   top.Gesamtbetrag,
18   ma.FotoURL as MitarbeiterFoto
19 }
```

```
D [A07] ZSS25_R4C_R_TOPVERKFMITARB_A2 ×
1 @AccessControl.authorizationCheck: #NOT_REQUIRED
2 @EndUserText.label: 'Reusable View für Top Mitarbeiter'
3 @Metadata.ignorePropagatedAnnotations: true
4 define root view entity ZSS25_R4C_R_TOPVERKFMITARB_A2 as select from ZSS25_R4C_V_TOPVERKFMITARB_A1
5
6 {
7   key MitarbeiterId,
8   MitarbeiterName,
9   Gesamtbetrag,
10  MitarbeiterFoto
11 }
12
```

```
D [A07] ZSS25_R4C_C_TOPVERKFMITARB_A3 ×
1 @AccessControl.authorizationCheck: #NOT_REQUIRED
2 @EndUserText.label: 'Consumpt Data Definition ZSS25_R4C_C_TOPVERKFMITARB_A3 [A07,200] - active - A07_200_learn-195_en'
3 @Metadata.ignorePropagatedAnnotations: true
4
5 @Metadata.allowExtensions: true
6 @Search.searchable: true
7 @ObjectModel.resultSet.sizeCategory: #XS
8 define root view entity ZSS25_R4C_C_TOPVERKFMITARB_A3 as projection on ZSS25_R4C_R_TOPVERKFMITARB_A2
9 {
10
11   @Search.defaultSearchElement: true
12   key MitarbeiterId,
13   MitarbeiterName,
14   Gesamtbetrag,
15   @Semantics.imageUrl: true
16   MitarbeiterFoto
17 }
```

```
[A07] ZSS25_R4C_SD_TOPVERKFMITARB_A4 ×
1 @EndUserText.label: 'Service Definition für Top Mitarbeiter'
2 define service ZSS25_R4C_SD_TOPVERKFMITARB_A4 {
3   expose ZSS25_R4C_C_TOPVERKFMITARB_A3;
4 }
```

[A07] ZSS25\_R4C\_SB\_TOPVERKFMITARB\_A5 ×

### Service Binding: ZSS25\_R4C\_SB\_TOPVERKFMITARB\_A5

General Information

This section describes general information about this service binding

Binding Type: OData V4 - UI

Services

Define services associated with the Service Binding

Local Service Endpoint: Published

Unpublish

Add Service...

Remove

Service Version Details

View information on selected service version

Service Information

Service URL: /sap/opu/odata4/sap/zss25\_r4c\_sb\_topverkfmitarb\_a5/srvd/sap/zss25\_r4c\_sd\_topverkfmitarb\_a4

Entity Set and Association

ZSS25\_R4C\_C\_TOPVERKFMITARB\_A3

type filter text

Service Name	Version	Service Definition
▼ ZSS25_R4C_SD_TOPVERKFMITARB	1.0.0	ZSS25_R4C_SD_TOPVERKFMITARB_A4

```

E [A07] ZSS25_R4C_ME_TOPVERKFMITARB_A6 ×
12⊖ // -----
13 // FACETS
14 // -----
15 @UI.facet: [
16   { position: 10, label: 'Mitarbeiter', type: #IDENTIFICATION_REFERENCE }
17 ]
18 // -----
19 // FELDER IN SINNVOLLER REIHENFOLGE
20 // -----
21 // -----
22
23 // --- Mitarbeiter-Foto ---
24 @EndUserText.label: ''
25 @EndUserText.quickInfo: 'Foto des Mitarbeiters.'
26 @UI.identification: [{ position: 10 }]
27 @UI.lineItem: [{ position: 10 }]
28 MitarbeiterFoto;
29
30⊖ // --- Mitarbeiter-ID (technisch, versteckt) ---
31 @EndUserText.label: 'Mitarbeiter-ID'
32 @EndUserText.quickInfo: 'Eindeutige Kennung des Mitarbeiters.'
33 @UI.hidden: true
34 MitarbeiterId;
35
36⊖ // --- Mitarbeiter-Name ---
37 @EndUserText.label: 'Name'
38 @EndUserText.quickInfo: 'Name des Mitarbeiters.'
39 @UI.identification: [{ position: 20 }]
40 @UI.lineItem: [{ position: 20 }]
41 @UI.selectionField: [{ position: 10 }]
42 MitarbeiterName;
43
44⊖ // --- Gesamtbetrag ---
45 @EndUserText.label: 'Gesamtbetrag (€)'
46 @EndUserText.quickInfo: 'Gesamtumsatz des Mitarbeiters.'
47 @UI.identification: [{ position: 30 }]
48 @UI.lineItem: [{ position: 30 }]
49 @UI.selectionField: [{ position: 20 }]
50 Gesamtbetrag;

```

Das Ergebnis nach diesen Änderungen sieht wie folgt aus:

Name:	Gesamtbetrag (€):		
<input type="text" value="Search"/>			
Top-Verkäufer (7)			
	Name	Gesamtbetrag (€)	
	Maximilian Hoffmann	425.00	>
	Kamal Badawii	344.40	>
	Lukas Schneider	304.00	>
	Clara Schäfer	287.00	>
	Lisa Müller	285.00	>
	Sophie Braun	223.00	>
	Anna Müller	48.00	>

## 5.2. Treuekunden- und Top-Kundenanalyse

In dieser CDS-View (ZSS25\_R4C\_V\_TREUEKUNDEN) wird ermittelt, welche Besucher als Treuekunden gelten, also mehr als einen Kauf getätigt haben.

```
D [A07] ZSS25_R4C_V_TREUEKUNDEN X
1 @AbapCatalog.viewEnhancementCategory: [#NONE]
2 @AccessControl.authorizationCheck: #NOT_REQUIRED
3 @EndUserText.label: 'View für Treuekunden'
4 @Metadata.ignorePropagatedAnnotations: true
5 @ObjectModel.usageType: {
6   serviceQuality: #X,
7   sizeCategory: #S,
8   dataClass: #MIXED
9 }
10 define view entity ZSS25_R4C_V_TREUEKUNDEN as
11   select from ZSS25_R4C_V_VERKF_A1
12 {
13   key BesucherName,
14   count( * ) as AnzahlKaeufe
15 }
16 group by BesucherName
17 having count( * ) > 1
```

BesucherName	AnzahlKaeufe
ANNA MÜLLER	5
SOPHIE WAGNER	4
MARCO ROSSI	4
TOBIAS SCHMIDT	3
LINH NGUYEN	2
OMAR ALI	2
JANA BECKER	2

In dieser CDS-View (ZSS25\_R4C\_V\_Topkunden) wird der Umsatz pro Kunde (Besucher) berechnet, um daraus die Top-Kunden zu ermitteln

```
D [A07] ZSS25_R4C_V_TOPKUNDEN X
1@AbapCatalog.viewEnhancementCategory: [#NONE]
2@AccessControl.authorizationCheck: #NOT_REQUIRED
3@EndUserText.label: 'View für Top Kunden'
4@Metadata.ignorePropagatedAnnotations: true
5@ObjectModel.usageType: {
6    serviceQuality: #X,
7    sizeCategory: #S,
8    dataClass: #MIXED
9}
10define view entity ZSS25_R4C_V_Topkunden as select from ZSS25_R4C_V_VERKF_A1
11{
12    key BesucherName,
13    sum( Betrag ) as Umsatz
14}
15where TicketStatus <> 'STORNIERT'
16group by BesucherName
```

BesucherName	Umsatz
ANNA MÜLLER	150.500000
MARCO ROSSI	85.00
SOPHIE WAGNER	83.250000
OMAR ALI	35.00
LINH NGUYEN	27.200000
JANA BECKER	26.00
TOBIAS SCHMIDT	25.500000

Für Kunden und deren Foto-URLs werden die neue CDS-, Reusable-, Consumption-Views, Service Definition, Binding und Metadata Extension erstellt:

```

D [A07] ZSS25_R4C_V_KUNDENANALYSE_A1 X
1@AbapCatalog.viewEnhancementCategory: [#NONE]
2@AccessControl.authorizationCheck: #NOT_REQUIRED
3@EndUserText.label: 'View für Kundenanalyse'
4@Metadata.ignorePropagatedAnnotations: true
5@ObjectModel.usageType:{}
6    serviceQuality: #X,
7    sizeCategory: #S,
8    dataClass: #MIXED
9}
10define view entity ZSS25_R4C_V_Kundenanalyse_A1 as select distinct from ZSS25_R4C_V_VERKF_A1 as vrkf
11right outer join ZSS25_R4C_V_TREUEKUNDEN as trkd on trkd.BesucherName = vrkf.BesucherName
12inner | join ZSS25_R4C_V_Topkunden as tpkd on tpkd.BesucherName = vrkf.BesucherName
13{
14    key vrkf.BesucherName as BesucherName,
15    trkd.AnzahlKaeufe as AnzahlKaeufe,
16    tpkd.Umsatz as Umsatz,
17    cast( ROUND( tpkd.Umsatz / trkd.AnzahlKaeufe,2) as abap.dec(10,2)) as DurchschnittlicherUmsatz,
18    vrkf.BesucherAltersgruppeURL as BesucherAltersgruppeFoto
19
20}
21

```

```

D [A07] ZSS25_R4C_R_KUDNENANALYSE_A2 X
1@AccessControl.authorizationCheck: #NOT_REQUIRED
2@EndUserText.label: 'Reusable View für Top Kundenanalyse'
3@Metadata.ignorePropagatedAnnotations: true
4define root view entity ZSS25_R4C_R_KUDNENANALYSE_A2 as select from ZSS25_R4C_V_Kundenanalyse_A1
5
6{
7    key BesucherName,
8    AnzahlKaeufe,
9    Umsatz,
10   DurchschnittlicherUmsatz,
11   BesucherAltersgruppeFoto
12}
13

```

```

D [A07] ZSS25_R4C_C_KUDNENANALYSE_A3 X
1@AccessControl.authorizationCheck: #NOT_REQUIRED
2@EndUserText.label: 'Consumption View für Top Kundenanalyse'
3@Metadata.ignorePropagatedAnnotations: true
4@Metadata.allowExtensions: true
5@Search.searchable: true
6@ObjectModel.resultSet.sizeCategory: #XS
7define root view entity ZSS25_R4C_C_KUDNENANALYSE_A3 as projection on ZSS25_R4C_R_KUDNENANALYSE_A2
8{
9    @Search.defaultSearchElement: true
10   key BesucherName,
11   AnzahlKaeufe,
12   Umsatz,
13   DurchschnittlicherUmsatz,
14   @Semantics.imageUrl: true
15   BesucherAltersgruppeFoto
16}
17

```

```

D [A07] ZSS25_R4C_SD_KUNDENANALYSE_A4 X
1@EndUserText.label: 'Service Definition für Top Kundenanalyse'
2define service ZSS25_R4C_SD_KUNDENANALYSE_A4 {
3    expose ZSS25_R4C_C_KUDNENANALYSE_A3;
4}

```

```

E [A07] ZSS25_R4C_ME_KUNDENANALYSE_A6 X
1 @Metadata.layer: #CUSTOMER
2
3 @UI: {
4   headerInfo: {
5     typeName: 'Kundenanalyse',
6     typeNamePlural: 'Kundenanalyse'
7   }
8 }
9
10 annotate entity ZSS25_R4C_C_KUDNENANALYSE_A3 with
11 {
12 // -----
13 // FACETS
14 // -----
15 @UI.facet: [
16   { position: 10, label: 'Besucher', type: #IDENTIFICATION_REFERENCE }
17 ]
18 // -----
19 // FELDER IN SINNVOLLER REIHENFOLGE
20 // -----
21
22 // --- Besucher-Foto ---
23 @EndUserText.label: ''
24 @EndUserText.quickInfo: 'Foto oder Symbol der Altersgruppe des Besuchers.'
25 @UI.identification: [{ position: 10 }]
26 @UI.lineItem: [{ position: 10 }]
27 BesucherAltersgruppeFoto;
28
29 // --- Besucher-Name ---
30 @EndUserText.label: 'Besucher'
31 @EndUserText.quickInfo: 'Name des Besuchers.'
32 @UI.identification: [{ position: 20 }]
33 @UI.lineItem: [{ position: 20 }]
34 @UI.selectionField: [{ position: 10 }]
35 BesucherName;
36
37

```

```
37
38@ // --- Anzahl Käufe ---
39 @EndUserText.label: 'Anzahl Käufe'
40 @EndUserText.quickInfo: 'Anzahl der Käufe, die der Besucher getätigt hat.'
41 @UI.identification: [{ position: 30 }]
42 @UI.lineItem: [{ position: 30 }]
43 @UI.selectionField: [{ position: 20 }]
44 AnzahlKaeufe;
45
46@ // --- Umsatz ---
47 @EndUserText.label: 'Umsatz (€)'
48 @EndUserText.quickInfo: 'Gesamtumsatz des Besuchers.'
49 @UI.identification: [{ position: 40 }]
50 @UI.lineItem: [{ position: 40 }]
51 @UI.selectionField: [{ position: 30 }]
52 Umsatz;
53
54@ // --- Durchschnittlicher Umsatz ---
55 @EndUserText.label: 'Durchschnittlicher Umsatz (€)'
56 @EndUserText.quickInfo: 'Durchschnittlicher Umsatz pro Kauf des Besuchers.'
57 @UI.identification: [{ position: 50 }]
58 @UI.lineItem: [{ position: 50 }]
59 @UI.selectionField: [{ position: 40 }]
60 DurchschnittlicherUmsatz;
61 }
```

Das Ergebnis sieht wie folgt aus:

Search	Besucher:	Anzahl Käufe:	Umsatz (€)	Durchschnittlicher Umsatz (€)	
<input type="text"/>	<input type="button" value="🔍"/>	<input type="button" value="⟳"/>	<input type="button" value="⟳"/>	<input type="button" value="⟳"/>	<input type="button" value="⟳"/>
Kundenanalyse (7)					
	Besucher	Anzahl Käufe	Umsatz (€)	Durchschnittlicher Umsatz (€)	
	ANNA MÜLLER	36	903.00	25.08	>
	MARCO ROSSI	16	340.00	21.25	>
	SOPHIE WAGNER	16	333.00	20.81	>
	TOBIAS SCHMIDT	9	138.00	15.33	>
	LINH NGUYEN	4	72.40	18.10	>
	OMAR ALI	4	70.00	17.50	>
	JANA BECKER	4	60.00	15.00	>

## 6. Fiori-Anwendungen in ABAP RAP

### 6.1. Von RAP zu Fiori: Implementierung und Anpassung einer Kinoverwaltungs-App

Um SAP Fiori mit dem RAP-Projekt zu verbinden, wurden die folgenden Schritte durchgeführt:

The screenshot shows the Fiori Open Application Generator interface. At the top, there is a dark header bar with the text '> app' on the left and 'recently used' with a gear icon on the right. Below this is a blue navigation bar with the text 'Fiori: Open Application Generator'. The main content area has a dark background and features a large title 'List Report Page' in white. Below the title, there is a description: 'Create an SAP Fiori elements application containing a list report and an object page.' At the bottom left of this section, there is a link 'More Information' and a blue circular icon containing a white document with horizontal lines, representing a list report.

## SAP Fiori generator ⓘ

### Template Selection

**Template:** List Report Page

### Data Source and Service Selection

Configure the data source and select a service.

#### Data Source and Service Selection

**Data Source:** Connect to a System

**System:** A07 Tunnel [LEARN-195]

**Service:** ZSS25\_R4C\_SB\_KIN

[More...](#)

Data Source \*

Connect to a System

System ⓘ \*

A07 Tunnel [LEARN-195]

Service (for user [LEARN-195]) \*

ZSS25\_R4C\_SB\_KINO\_A5 > ZSS25\_R4C\_...

#### Entity Selection

**Main Entity:** ZSS25\_R4C\_C\_K  
INO\_A3

**Navigation Entity:** \_Raum

**Generate Annotations:** Yes

[More...](#)

#### Project Attributes

[More...](#)

◀ Back

Next ▶

# SAP Fiori generator ⓘ

## Template Selection

**Template:** List Report Page

## Data Source and Service Selection

**Data Source:** Connect to a System

**System:** A07 Tunnel [LEARN-195]

**Service:** ZSS25 R4C SB KIN

[More...](#)

## Entity Selection

Configure the selected service.

Main Entity \*

ZSS25\_R4C\_C\_VERKF\_A3

Automatically add table columns to the list page and a section to the object page if none already exists?

Yes  No

Table Type ⓘ \*

Responsive

## Entity Selection

**Main Entity:** ZSS25\_R4C\_C\_VERKF\_A3

**Generate Annotations:** No

**Table Type:** Responsive

## Project Attributes

< Back

Next >

# SAP Fiori generator ⓘ

## Template Selection

**Template:** List Report Page

## Data Source and Service Selection

**Data Source:** Connect to a System

**System:** A07 Tunnel [LEAR N-195]

**Service:** ZSS25 R4C SB KI  
[More...](#)

## Entity Selection

**Main Entity:** ZSS25\_R4C\_C\_VERKF\_A3

**Generate Annotations:** No

**Table Type:** Responsive

## Project Attributes

**Module Name:** zss25\_r4c\_kino

**Application Title:** Kinoverwaltung

**Application Namespace:**

## Project Attributes

Configure the main project attributes.

Module Name ⓘ \*

zss25\_r4c\_kino

Application Title ⓘ

Kinoverwaltung

Application Namespace ⓘ

com.zss25.learnr4c.kinoverwaltung

Description ⓘ

Fiori App zur Verwaltung von Kinos

Project Folder Path \*

c:\Users\kamal\OneDrive\Desktop\RAP\F

Minimum SAPUI5 Version ⓘ \*

1.108.33 (Source system version)

Enable TypeScript

Yes  No

< Back

Next >

# SAP Fiori generator ⓘ

## ● Template Selection

**Template:** List Report Page

## ● Data Source and Service Selection

**Data Source:** Connect to a System

**System:** A07 Tunnel [LEARN-195]

**Service:** ZSS25\_R4C\_SB\_KI  
[More...](#)

## ● Entity Selection

**Main Entity:** ZSS25\_R4C\_C\_VERKF\_A3

**Generate Annotations:** No

**Table Type:** Responsive  
[More...](#)

## ● Project Attributes

**Module Name:** zss25\_r4c\_kino

**Application Title:** Kinoverwaltung

## Deployment Configuration

Configure deployment settings.

Please choose the target. \*

ABAP

Select Target System \*

A07 Tunnel [LEARN-195] (Source system) ▾

SAPUI5 ABAP Repository ⓘ \*

ZSS25\_R4C\_RK

Deployment Description ⓘ

Kinoverwaltung

Select How You Want to Enter the Package \*

Choose from Existing

Package ⓘ \*

\$SS25\_RAP4CINEMA

\$SS25\_RAP4CINEMA

< Back

Next >

## SAP Fiori generator ⓘ

### Template Selection

**Template:** List Report Page

### Data Source and Service Selection

**Data Source:** Connect to a System

**System:** A07 Tunnel [LEAR N-195]

**Service:** ZSS25 R4C SB KI  
[More...](#)

### Entity Selection

**Main Entity:** ZSS25\_R4C\_C\_VERKF\_A3

**Generate Annotations:** No

**Table Type:** Responsive  
[More...](#)

### Project Attributes

**Module Name:** zss25\_r4c\_kino

**Application Title:** Kinoverwaltung

## SAP Fiori Launchpad Configuration

Configure SAP Fiori launchpad settings .

Semantic Object \*

kinos

Action \*

manage

Title \*

Kinoverwaltung

Subtitle

Fiori App zur Verwaltung von Kinos

Back

Finish

# Application Information



## Project Details

<b>Identifier</b>	com.zss25.learnr4c.kinoverwaltung.z...
<b>Title</b>	Kinoverwaltung
<b>Namespace</b>	com.zss25.learnr4c.kinoverwaltung
<b>Location</b>	c:\Users\kamal\OneDrive\Desktop\R...
<b>Files</b>	<a href="#">package.json</a>   <a href="#">manifest.json</a>
<b>Type</b>	SAP Fiori elements
<b>Min UI5 Version</b>	1.108.33
<b>Main Service</b>	<a href="#">zss25_r4c_sd_kino_a4 (V4.0)</a>

## Pages [Page Map](#)

- [List Report](#)  
ZSS25\_R4C\_C\_VERKF\_A3List
- [Object Page](#)  
ZSS25\_R4C\_C\_VERKF\_A3ObjectPage

## Status

- ✓ Latest version of node module '@sap/ux-ui5-tooling@1.19.1' installed.
- ✓ Latest version of node module '@sap-ux/ui5-middleware-fe-mockserver@2.3.11' installed.

## Configuration

Zur Ausführung von SAP Fiori im Frontend wurden die folgenden Schritte vorgenommen:

```

● PS C:\Users\kamal\OneDrive\Desktop\RAP\Fiori-App> cd zss25_r4c_kino
● PS C:\Users\kamal\OneDrive\Desktop\RAP\Fiori-App\zss25_r4c_kino> npm install
    up to date, audited 1104 packages in 3s

  168 packages are looking for funding
    run `npm fund` for details

  6 low severity vulnerabilities

  To address all issues, run:
    npm audit fix

  Run `npm audit` for details.
○ PS C:\Users\kamal\OneDrive\Desktop\RAP\Fiori-App\zss25_r4c_kino> npm audit report

```

```

PS C:\Users\kamal\OneDrive\Desktop\RAP\Fiori-App\zss25_r4c_kino> npm audit report
  Depends on vulnerable versions of inquirer
    node_modules/@sap-ux/adp-tooling
      @sap-ux/preview-middleware 0.13.74 - 0.23.18
        Depends on vulnerable versions of @sap-ux/adp-tooling
          node_modules/@sap-ux/preview-middleware
            @sap/ux-ui5-tooling >=1.14.2
              Depends on vulnerable versions of @sap-ux/preview-middleware
                node_modules/@sap/ux-ui5-tooling

  6 low severity vulnerabilities

  To address all issues, run:
    npm audit fix
PS C:\Users\kamal\OneDrive\Desktop\RAP\Fiori-App\zss25_r4c_kino> npm run start
>>

```

Verkäufe (15)											
Mitarbeiter	Datum	Besucher	Erlaubnis	Besucherg...	Ticketart	Ticketgültig...	Zeitkategorie	Zahlungsst...	Zahlungsart	Ticketpreis (€)	Rabatt (%)
	Oct 11, 2025	ANNA MÜLLER	(Ja)							15.00	0.00
	Oct 15, 2025	TOBIAS SCHMIDT	(Ja)							20.00	25.00
	Oct 14, 2025	OMAR ALI	(Ja)							35.00	0.00
	Oct 12, 2025	SOPHIE WAGNER	(Ja)							35.00	5.00
	Oct 27, 2025	MARCO ROSSI	(Ja)							15.00	0.00
	Oct 15, 2025	ANNA MÜLLER	(Ja)							45.00	10.00
	Oct 22, 2025	LINH NGUYEN	(Ja)							20.00	0.00
	Oct 30, 2025	JANA BECKER	(Nein)							8.00	0.00

Für die englische Benennung der Spalten im SAP-Fiori-Frontend wurden die folgenden Schritte durchgeführt:

```

annotation.xml M X
zss25_r4c_kino > webapp > annotations > annotation.xml
1 <edmx:Edmx xmlns:edmx="http://docs.oasis-open.org/odata/ns/edmx" Version="4.0">
2   <edmx:Reference Uri="https://sap.github.io/odata-vocabularies/vocabularies/Common.xml">
3     <edmx:Include Namespace="com.sap.vocabularies.Common.v1" Alias="Common"/>
4   </edmx:Reference>
5   <edmx:Reference Uri="https://sap.github.io/odata-vocabularies/vocabularies/UI.xml">
6     <edmx:Include Namespace="com.sap.vocabularies.UI.v1" Alias="UI"/>
7   </edmx:Reference>
8   <edmx:Reference Uri="https://sap.github.io/odata-vocabularies/vocabularies/Communication.xml">
9     <edmx:Include Namespace="com.sap.vocabularies.Communication.v1" Alias="Communication"/>
10    </edmx:Reference>
11    <edmx:Reference Uri="/sap/opu/odata4/sap/zss25_r4c_sb_kino_a5/srvd/sap/zss25_r4c_sd_kino_a4/0001/$metadata">
12      <edmx:Include Namespace="com.sap.gateway.srvd.zss25_r4c_sd_kino_a4.v0001" Alias="SAP_self"/>
13    </edmx:Reference>
14  </edmx:DataServices>
15  <edmx:Schema xmlns="http://docs.oasis-open.org/odata/ns/edm" Namespace="local">
16    <!-- ===== -->
17    <!-- General -->
18    <!-- ===== -->
19    <Annotations Target="SAP_self.ZSS25_R4C_C_VERKF_A3Type/VerkaufId">
20      <Annotation Term="Common.Label" String="{@i18n>VerkaufId}"/>
21    </Annotations>
22    <Annotations Target="SAP_self.ZSS25_R4C_C_VERKF_A3Type/MitarbeiterFoto">
23      <Annotation Term="Common.Label" String="{@i18n>MitarbeiterFoto}"/>
24    </Annotations>
25    <Annotations Target="SAP_self.ZSS25_R4C_C_VERKF_A3Type/Verkaufsdatum">
26      <Annotation Term="Common.Label" String="{@i18n>Verkaufsdatum}"/>
27    </Annotations>
28    <!-- ===== -->
29    <!-- Visitor -->
30    <!-- ===== -->
31    <Annotations Target="SAP_self.ZSS25_R4C_C_VERKF_A3Type/BesucherName">
32      <Annotation Term="Common.Label" String="{@i18n>BesucherName}"/>
33    </Annotations>
34    <Annotations Target="SAP_self.ZSS25_R4C_C_VERKF_A3Type/BesucherIstErlaubt">
35      <Annotation Term="Common.Label" String="{@i18n>BesucherIstErlaubt}"/>
36    </Annotations>
37    <Annotations Target="SAP_self.ZSS25_R4C_C_VERKF_A3Type/BesucherAltersgruppeURL">
38      <Annotation Term="Common.Label" String="{@i18n>BesucherAltersgruppeURL}"/>
39    </Annotations>
40    <Annotations Target="SAP_self.ZSS25_R4C_C_VERKF_A3Type/BesucherGeburtsdatum">
41      <Annotation Term="Common.Label" String="{@i18n>BesucherGeburtsdatum}"/>
42    </Annotations>
43    <Annotations Target="SAP_self.ZSS25_R4C_C_VERKF_A3Type/BesucherAlter">
44      <Annotation Term="Common.Label" String="{@i18n>BesucherAlter}"/>
45    </Annotations>
46    <Annotations Target="SAP_self.ZSS25_R4C_C_VERKF_A3Type/BesucherEmail">
47      <Annotation Term="Common.Label" String="{@i18n>BesucherEmail}"/>
48    </Annotations>
49    <Annotations Target="SAP_self.ZSS25_R4C_C_VERKF_A3Type/BesucherTelefon">
50      <Annotation Term="Common.Label" String="{@i18n>BesucherTelefon}"/>
51    </Annotations>
52    <Annotations Target="SAP_self.ZSS25_R4C_C_VERKF_A3Type/BesucherAdresse">
53      <Annotation Term="Common.Label" String="{@i18n>BesucherAdresse}"/>
54    </Annotations>
55  </Annotations>
56  </edmx:Schema>
57  </edmx:DataServices>
58  </edmx:Edmx>

```

```

annotation.xml M X
zss25_r4c_kino > webapp > annotations > annotation.xml
1 <edmx:Edmx xmlns:edmx="http://docs.oasis-open.org/odata/ns/edmx" Version="4.0">
2   <edmx:DataServices>
3     <edmx:Schema xmlns="http://docs.oasis-open.org/odata/ns/edm" Namespace="local">
4       <!-- ===== -->
5       <!-- Visitor -->
6       <!-- ===== -->
7       <Annotations Target="SAP_self.ZSS25_R4C_C_VERKF_A3Type/BesucherName">
8         <Annotation Term="Common.Label" String="{@i18n>BesucherName}"/>
9       </Annotations>
10      <Annotations Target="SAP_self.ZSS25_R4C_C_VERKF_A3Type/BesucherIstErlaubt">
11        <Annotation Term="Common.Label" String="{@i18n>BesucherIstErlaubt}"/>
12      </Annotations>
13      <Annotations Target="SAP_self.ZSS25_R4C_C_VERKF_A3Type/BesucherAltersgruppeURL">
14        <Annotation Term="Common.Label" String="{@i18n>BesucherAltersgruppeURL}"/>
15      </Annotations>
16      <Annotations Target="SAP_self.ZSS25_R4C_C_VERKF_A3Type/BesucherGeburtsdatum">
17        <Annotation Term="Common.Label" String="{@i18n>BesucherGeburtsdatum}"/>
18      </Annotations>
19      <Annotations Target="SAP_self.ZSS25_R4C_C_VERKF_A3Type/BesucherAlter">
20        <Annotation Term="Common.Label" String="{@i18n>BesucherAlter}"/>
21      </Annotations>
22      <Annotations Target="SAP_self.ZSS25_R4C_C_VERKF_A3Type/BesucherEmail">
23        <Annotation Term="Common.Label" String="{@i18n>BesucherEmail}"/>
24      </Annotations>
25      <Annotations Target="SAP_self.ZSS25_R4C_C_VERKF_A3Type/BesucherTelefon">
26        <Annotation Term="Common.Label" String="{@i18n>BesucherTelefon}"/>
27      </Annotations>
28      <Annotations Target="SAP_self.ZSS25_R4C_C_VERKF_A3Type/BesucherAdresse">
29        <Annotation Term="Common.Label" String="{@i18n>BesucherAdresse}"/>
30      </Annotations>
31    </Annotations>
32    </edmx:Schema>
33  </edmx:DataServices>
34  </edmx:Edmx>

```

```

annotation.xml M X
zss25_r4c_kino > webapp > annotations > annotation.xml
  2   <edmx:Edmx xmlns:edmx="http://docs.oasis-open.org/odata/ns/edmx" Version="4.0">
  6     <edmx:DataServices>
 17       <Schema xmlns="http://docs.oasis-open.org/odata/ns/edm" Namespace="local">
 56         <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/BesucherAdresse">
 58           </Annotations>
 59           <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/BesucherRegistrierung">
 60             <Annotation Term="Common.Label" String="{@i18n>BesucherRegistrierung}"/>
 61           </Annotations>
 62
 63           <!-- ===== -->
 64           <!-- Ticket -->
 65           <!-- ===== -->
 66           <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/TicketTypURL">
 67             <Annotation Term="Common.Label" String="{@i18n>TicketTypURL}"/>
 68           </Annotations>
 69           <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/TicketIstGueltigURL">
 70             <Annotation Term="Common.Label" String="{@i18n>TicketIstGueltigURL}"/>
 71           </Annotations>
 72           <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/TicketSitzplatznummer">
 73             <Annotation Term="Common.Label" String="{@i18n>TicketSitzplatznummer}"/>
 74           </Annotations>
 75           <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/BesucherSitzReihe">
 76             <Annotation Term="Common.Label" String="{@i18n>BesucherSitzReihe}"/>
 77           </Annotations>
 78           <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/TicketQrCode">
 79             <Annotation Term="Common.Label" String="{@i18n>TicketQrCode}"/>
 80           </Annotations>
 81           <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/TicketQrCodeFoto">
 82             <Annotation Term="Common.Label" String="{@i18n>TicketQrCodeFoto}"/>
 83           </Annotations>
 84           <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/TicketQrCodeDFoto">

```

```

annotation.xml M X
zss25_r4c_kino > webapp > annotations > annotation.xml
  2   <edmx:Edmx xmlns:edmx="http://docs.oasis-open.org/odata/ns/edmx" Version="4.0">
  6     <edmx:DataServices>
 17       <Schema xmlns="http://docs.oasis-open.org/odata/ns/edm" Namespace="local">
 57         <Annotations>
 58           <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/TicketQrCodeDFoto">
 59             <Annotation Term="Common.Label" String="{@i18n>TicketQrCodeDFoto}"/>
 60           </Annotations>
 61
 62           <!-- ===== -->
 63           <!-- Payment -->
 64           <!-- ===== -->
 65           <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/ZeitkategorieURL">
 66             <Annotation Term="Common.Label" String="{@i18n>ZeitkategorieURL}"/>
 67           </Annotations>
 68           <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/ZahlungsstatusURL">
 69             <Annotation Term="Common.Label" String="{@i18n>ZahlungsstatusURL}"/>
 70           </Annotations>
 71           <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/Zahlungsart">
 72             <Annotation Term="Common.Label" String="{@i18n>Zahlungsart}"/>
 73           </Annotations>
 74           <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/ZahlungsartURL">
 75             <Annotation Term="Common.Label" String="{@i18n>ZahlungsartURL}"/>
 76           </Annotations>
 77           <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/TicketPreis">
 78             <Annotation Term="Common.Label" String="{@i18n>TicketPreis}"/>
 79           </Annotations>
 80           <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/Rabatt">
 81             <Annotation Term="Common.Label" String="{@i18n>Rabatt}"/>
 82           </Annotations>
 83           <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/Betrag">
 84             <Annotation Term="Common.Label" String="{@i18n>Betrag}"/>
 85           </Annotations>
 86         </Annotations>

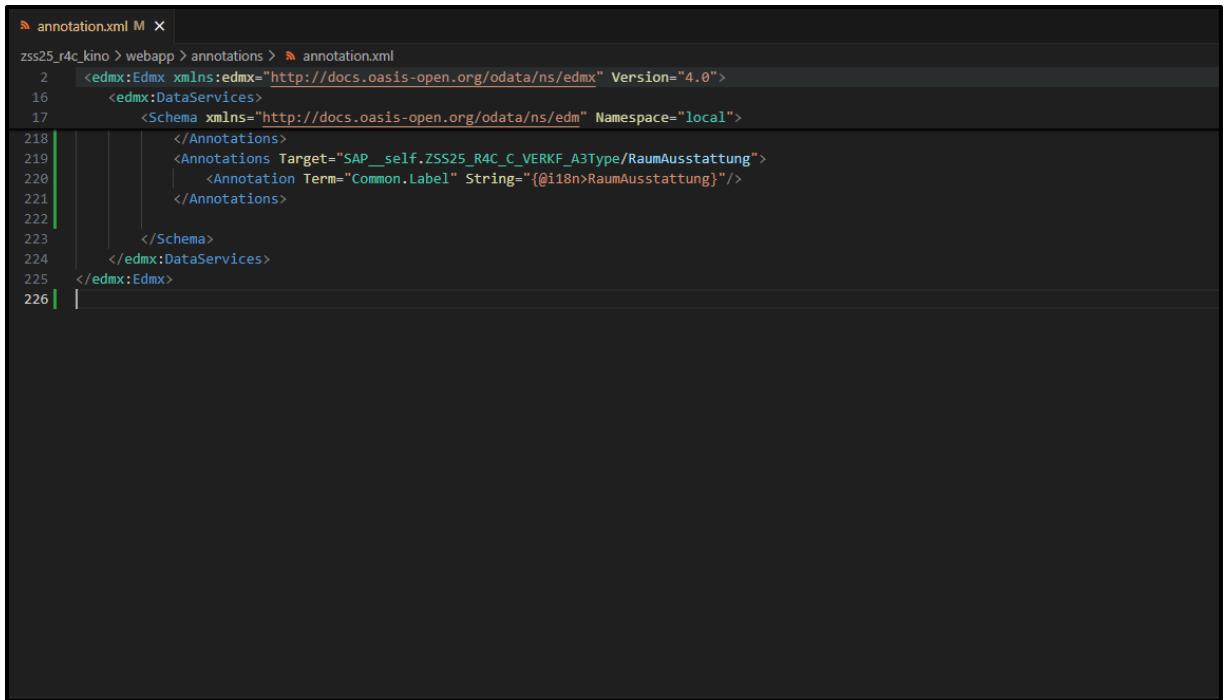
```

```
annotation.xml M X
zss25_r4c_kino > webapp > annotations > annotation.xml
  2   <edmx:Edmx xmlns:edmx="http://docs.oasis-open.org/odata/ns/edmx" Version="4.0">
  6     <edmx:DataServices>
  7       <Schema xmlns="http://docs.oasis-open.org/odata/ns/edm" Namespace="local">
 10         <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/Betrag">
 11           </Annotations>
 12
 13           <!-- ===== -->
 14           <!-- Screening -->
 15           <!-- ===== -->
 16           <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/Vorstellungsdatum">
 17             <Annotation Term="Common.Label" String="{@i18n>Vorstellungsdatum}"/>
 18           </Annotations>
 19           <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/VorstellungStartzeit">
 20             <Annotation Term="Common.Label" String="{@i18n>VorstellungStartzeit}"/>
 21           </Annotations>
 22           <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/VorstellungEndzeit">
 23             <Annotation Term="Common.Label" String="{@i18n>VorstellungEndzeit}"/>
 24           </Annotations>
 25           <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/VorstellungSprache">
 26             <Annotation Term="Common.Label" String="{@i18n>VorstellungSprache}"/>
 27           </Annotations>
 28           <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/VorstellungUntertitel">
 29             <Annotation Term="Common.Label" String="{@i18n>VorstellungUntertitel}"/>
 30           </Annotations>
 31           <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/VorstellungFormat">
 32             <Annotation Term="Common.Label" String="{@i18n>VorstellungFormat}"/>
 33           </Annotations>
 34           <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/VorstellungStatus">
 35             <Annotation Term="Common.Label" String="{@i18n>VorstellungStatus}"/>
 36           </Annotations>
 37         </Annotations>
```

```
annotation.xml M X
zss25_r4c_kino > webapp > annotations > annotation.xml
  2   <edmx:Edmx xmlns:edmx="http://docs.oasis-open.org/odata/ns/edmx" Version="4.0">
  6     <edmx:DataServices>
  7       <Schema xmlns="http://docs.oasis-open.org/odata/ns/edm" Namespace="local">
 139         <!-- Movie -->
 140         <!-- ===== -->
 141         <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/FilmTitel">
 142           <Annotation Term="Common.Label" String="{@i18n>FilmTitel}"/>
 143         </Annotations>
 144         <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/FilmGenre">
 145           <Annotation Term="Common.Label" String="{@i18n>FilmGenre}"/>
 146         </Annotations>
 147         <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/FilmDauerMinuten">
 148           <Annotation Term="Common.Label" String="{@i18n>FilmDauerMinuten}"/>
 149         </Annotations>
 150         <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/FilmAltersfreigabe">
 151           <Annotation Term="Common.Label" String="{@i18n>FilmAltersfreigabe}"/>
 152         </Annotations>
 153         <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/FilmProduktionsjahr">
 154           <Annotation Term="Common.Label" String="{@i18n>FilmProduktionsjahr}"/>
 155         </Annotations>
 156         <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/FilmRegisseur">
 157           <Annotation Term="Common.Label" String="{@i18n>FilmRegisseur}"/>
 158         </Annotations>
 159
 160         <!-- ===== -->
 161         <!-- Employee -->
 162         <!-- ===== -->
 163         <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/MitarbeiterId">
 164           <Annotation Term="Common.Label" String="{@i18n>MitarbeiterId}"/>
 165         </Annotations>
 166         <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/MitarbeiterName">
```

```

annotation.xml M X
zss25_r4c_kino > webapp > annotations > annotation.xml
  2   <edmx:Edmx xmlns:edmx="http://docs.oasis-open.org/odata/ns/edmx" Version="4.0">
  6     <edmx:DataServices>
 17       <Schema xmlns="http://docs.oasis-open.org/odata/ns/edm" Namespace="local">
103         <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/MitarbeiterName">
166           <Annotation Term="Common.Label" String="{@i18n>MitarbeiterName}"/>
167         </Annotations>
168         <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/MitarbeiterGeburtsdatum">
169           <Annotation Term="Common.Label" String="{@i18n>MitarbeiterGeburtsdatum}"/>
170         </Annotations>
171         <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/MitarbeiterRolle">
172           <Annotation Term="Common.Label" String="{@i18n>MitarbeiterRolle}"/>
173         </Annotations>
174         <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/MitarbeiterAbteilung">
175           <Annotation Term="Common.Label" String="{@i18n>MitarbeiterAbteilung}"/>
176         </Annotations>
177         <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/MitarbeiterGehalt">
178           <Annotation Term="Common.Label" String="{@i18n>MitarbeiterGehalt}"/>
179         </Annotations>
180         <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/MitarbeiterKontaktinfo">
181           <Annotation Term="Common.Label" String="{@i18n>MitarbeiterKontaktinfo}"/>
182         </Annotations>
183
184         <!-- ===== -->
185         <!-- Cinema -->
186         <!-- ===== -->
187         <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/KinoName">
188           <Annotation Term="Common.Label" String="{@i18n>KinoName}"/>
189         </Annotations>
190         <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/KinoEroeffnungsdatum">
191           <Annotation Term="Common.Label" String="{@i18n>KinoEroeffnungsdatum}"/>
192         </Annotations>
193
194         <!-- ===== -->
195         <!-- Room -->
196         <!-- ===== -->
197         <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/KinoAdresse">
198           <Annotation Term="Common.Label" String="{@i18n>KinoAdresse}"/>
199         </Annotations>
200         <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/KinoPlz">
201           <Annotation Term="Common.Label" String="{@i18n>KinoPlz}"/>
202         </Annotations>
203         <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/KinoStadt">
204           <Annotation Term="Common.Label" String="{@i18n>KinoStadt}"/>
205         </Annotations>
206         <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/KinoAnzahlRaeume">
207           <Annotation Term="Common.Label" String="{@i18n>KinoAnzahlRaeume}"/>
208         </Annotations>
209
210         <!-- ===== -->
211         <!-- Room -->
212         <!-- ===== -->
213         <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/RaumName">
214           <Annotation Term="Common.Label" String="{@i18n>RaumName}"/>
215         </Annotations>
216         <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/RaumSitzplaetze">
217           <Annotation Term="Common.Label" String="{@i18n>RaumSitzplaetze}"/>
218         </Annotations>
219         <Annotations Target="SAP__self.ZSS25_R4C_C_VERKF_A3Type/RaumAusstattung">
```



The screenshot shows a code editor window with a dark theme. The file being edited is named "annotation.xml". The code is an XML snippet defining annotations for a schema. It includes declarations for namespaces, a schema with annotations, and a data service. The annotations define a label for a specific term in the schema.

```
annotation.xml M X
zss25_r4c_kino > webapp > annotations > annotation.xml
  2   <edmx:Edmx xmlns:edmx="http://docs.oasis-open.org/odata/ns/edmx" Version="4.0">
  6     <edmx:DataServices>
  7       <Schema xmlns="http://docs.oasis-open.org/odata/ns/edm" Namespace="local">
  8         </Annotations>
  9         <Annotations Target="SAP_self.ZSS25_R4C_C_VERKF_A3Type/RaumAusstattung">
 10           <Annotation Term="Common.Label" String="{@i18n>RaumAusstattung}" />
 11         </Annotations>
 12       </Schema>
 13     </edmx:DataServices>
 14   </edmx:Edmx>
 15
 16
 17
 18
 19
 20
 21
 22
 23
 24
 25
 26
```

```
i18n.properties M X
zss25_r4c_kino > webapp > i18n > i18n.properties
1 # This is the resource bundle for com.zss25.learnr4c.kinoverwaltung.zss25r4ckino
2
3 #Texts for manifest.json
4
5 #XTIT: Application name
6 appTitle=Kinoverwaltung
7
8 #YDES: Application description
9 appDescription=Fiori App zur Verwaltung von Kinos
10
11 #XFLD_42
12 kinos-manage.flpTitle=Kinoverwaltung
13
14 #XFLD_51
15 kinos-manage.flpSubtitle=Fiori App zur Verwaltung von Kinos
16
17 # =====
18 # General / Header
19 # =====
20 VerkaufId=Sale ID
21 MitarbeiterFoto=Employee Photo
22 Verkaufsdatum=Sale Date
23
24 # =====
25 # Visitor
26 # =====
27 BesucherName=Visitor Name
28 BesucherIstErlaubt=Permission
29 BesucherAltersgruppeURL=Visitor Group
30 BesucherGeburtsdatum=Birth Date
31 BesucherAlter=Age
```

```
i18n.properties M X
zss25_r4c_kino > webapp > i18n > i18n.properties
30 BesucherGeburtsdatum=Birth Date
31 BesucherAlter=Age
32 BesucherEmail=Email
33 BesucherTelefon=Phone
34 BesucherAdresse=Address
35 BesucherRegistrierung=Registration
36
37 # =====
38 # Ticket
39 # =====
40 TicketTypURL=Ticket Type
41 TicketIstGueltigURL=Ticket Validity
42 TicketSitzplatznummer=Seat Number
43 BesucherSitzReihe=Seat Row
44 TicketQrCode=Ticket QR Code
45 TicketQrCodeCFoto=Ticket C QR Image
46 TicketQrCodeDFoto=Ticket D QR Image
47
48 # =====
49 # Payment
50 # =====
51 ZeitkategorieURL=Time Category
52 ZahlungsstatusURL=Payment Status
53 Zahlungsart=Payment Method Key
54 ZahlungsartURL=Payment Method
55 TicketPreis=Ticket Price (€)
56 Rabatt=Discount (%)
57 Betrag=Amount (€)
```

```

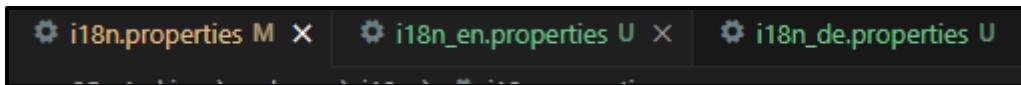
i18n.properties M X
zss25_r4c_kino > webapp > i18n > i18n.properties
59 # =====
60 # Screening / Vorstellung
61 # =====
62 Vorstellungdatum=Screening Date
63 VorstellungStartzeit=Start Time
64 VorstellungEndzeit=End Time
65 VorstellungSprache=Language
66 VorstellungUntertitel=Subtitles
67 VorstellungFormat=Format
68 VorstellungStatus=Status
69
70 # =====
71 # Film
72 # =====
73 FilmTitel=Movie Title
74 FilmGenre=Genre
75 FilmDauerMinuten=Duration (Minutes)
76 FilmAltersfreigabe=Age Rating
77 FilmProduktionsjahr=Production Year
78 FilmRegisseur=Director
79
80 # =====
81 # Employee / Seller
82 # =====
83 MitarbeiterId=Employee ID
84 MitarbeiterName=Employee Name
85 MitarbeiterGeburtsdatum=Birth Date
86 MitarbeiterRolle=Role
87 MitarbeiterAbteilung=Department
88 MitarbeiterGehalt=Salary
89 MitarbeiterKontaktinfo=Contact Information

i18n.properties M X
zss25_r4c_kino > webapp > i18n > i18n.properties
90 # =====
91 # Cinema
92 # =====
93 KinoName=Cinema Name
94 KinoEroeffnungsdatum=Opening Date
95 KinoAdresse=Address
96 KinoPlz=Postal Code
97 KinoStadt=City
98 KinoAnzahlRaeume=Number of Rooms
99 KinoKontaktinfo=Contact Information
100
101 # =====
102 # Room
103 # =====
104 RaumName=Room Name
105 RaumSitzplaetze=Number of Seats
106 RaumAusstattung=Equipment
107

```

Die Spaltennamen wurden erfolgreich auf Englisch umgestellt:

Zur Umsetzung der Mehrsprachigkeit (Englisch und Deutsch) wurden zwei entsprechende Dateien erstellt:



Bei Übergabe des deutschen Werts an die URL-Parameter wird folgendes Ergebnis angezeigt:

localhost:8080/test/fipSandbox.html?sap-ui-xx-viewCache=false&sap-language=de#comss25learnr4ckinoverwaltung-tile&/?sap-iapp-state=TASHGBB8DVRTL5LVUBO02SBT2V1IDCJ02KHMWXXH

Bei Übergabe des englischen Werts an die URL-Parameter wird folgendes Ergebnis angezeigt:

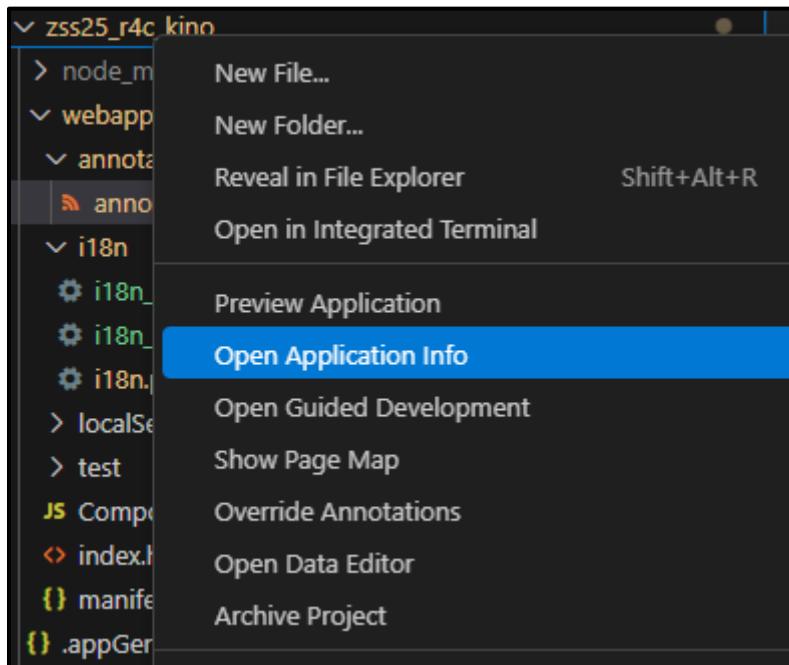
Employee ...	Sale Date	Visitor Name	Permission	Visitor Group	Ticket Type	Ticket Val...	Time Cate...	Payment S...	Ticket Price (€)	Discount (%)	Amount (€)	Screening Date	Seat Number
	Oct 11, 2025	ANNA MÜLLER	(Ja)	Normal	VIP	Normal	Normal	✓	15.00	0.00	15.00	Nov 9, 2025	13 >
	Oct 15, 2025	TOBIAS SCHMIDT	(Ja)	Normal	VIP	Normal	Normal	✓	20.00	25.00	15.00	Nov 9, 2025	32 >
	Oct 14, 2025	OMAR ALI	(Ja)	Normal	VIP	Normal	Normal	✓	35.00	0.00	35.00	Oct 15, 2025	2 >
	Oct 12, 2025	SOPHIE WAGNER	(Ja)	Normal	VIP	Normal	Normal	✓	35.00	5.00	33.25	Oct 12, 2025	1 >
	Oct 27, 2025	MARCO ROSSI	(Ja)	Normal	VIP	Normal	Normal	✓	15.00	0.00	15.00	Oct 18, 2025	5 >

Zur absteigenden Sortierung der Daten nach dem Betrag (€) wurde der folgende Code eingebunden:

```
<Annotations Target="SAP_self.ZSS25_R4C_C_VERKF_A3Type">
  <Annotation Term="UI.PresentationVariant">
    <Record>
      <PropertyValue Property="Visualizations">
        <Collection>
          <AnnotationPath>@UI.LineItem</AnnotationPath>
        </Collection>
      </PropertyValue>
      <PropertyValue Property="SortOrder">
        <Collection>
          <Record Type="Common.SortOrderType">
            <PropertyValue Property="Property" PropertyPath="Betrag" />
            <PropertyValue Property="Descending" Bool="true" />
          </Record>
        </Collection>
      </PropertyValue>
    </Record>
  </Annotation>
</Annotations>
```

Employee ...	Sale Date	Visitor Name	Permission	Visitor Group	Ticket Type	Ticket Val...	Time Cate...	Payment S...	Ticket Price (€)	Discount (%)	Amount (€)	Screening Date	Seat Number
	Oct 15, 2025	ANNA MÜLLER	(Ja)	Normal	VIP	Normal	Normal	✓	45.00	10.00	40.50	Nov 2, 2025	1 >
	Sep 27, 2025	ANNA MÜLLER	(Ja)	Normal	VIP	Normal	Normal	✓	40.00	0.00	40.00	Nov 2, 2025	5 >
	Oct 25, 2025	ANNA MÜLLER	(Ja)	Normal	VIP	Normal	Normal	✓	40.00	5.00	38.00	Nov 2, 2025	4 >
	Oct 14, 2025	OMAR ALI	(Ja)	Normal	VIP	Normal	Normal	✓	35.00	0.00	35.00	Oct 15, 2025	2 >
	Oct 28, 2025	MARCO ROSSI	(Ja)	Normal	VIP	Normal	ONLY TODAY	OPEN	35.00	0.00	35.00	Oct 18, 2025	3 >
	Oct 12, 2025	MARCO ROSSI	(Ja)	Normal	VIP	Normal	Normal	✓	35.00	0.00	35.00	Oct 25, 2025	5 >
	Oct 12, 2025	SOPHIE WAGNER	(Ja)	Normal	VIP	Normal	Normal	✓	35.00	5.00	33.25	Oct 12, 2025	1 >

Zur farblichen Hervorhebung werden Beträge mit Rabatt grün und Beträge ohne Rabatt gelb markiert. Die Umsetzung erfolgt in den folgenden Schritten:



This is a screenshot of a report page titled 'List Report Page'. Below the title, there is a section with the heading 'Add Status Colors and Icons to a Column in a Table'. This section is highlighted with a light gray background.

```

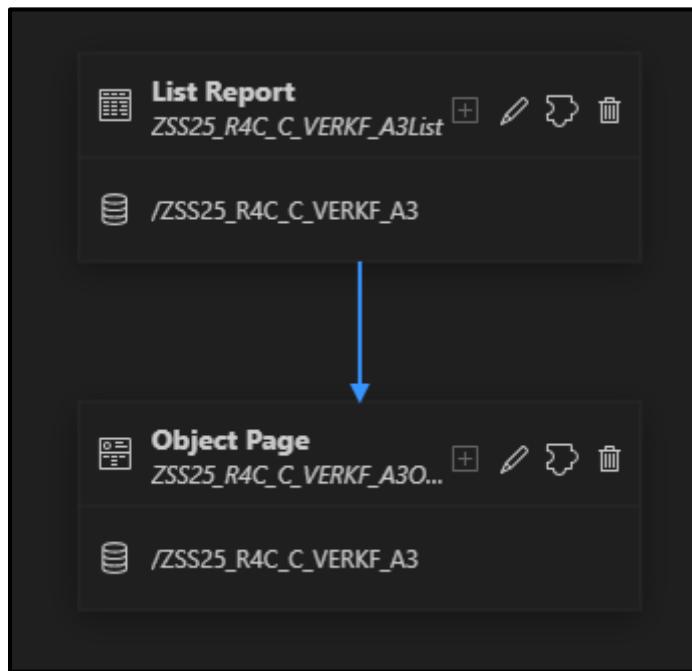
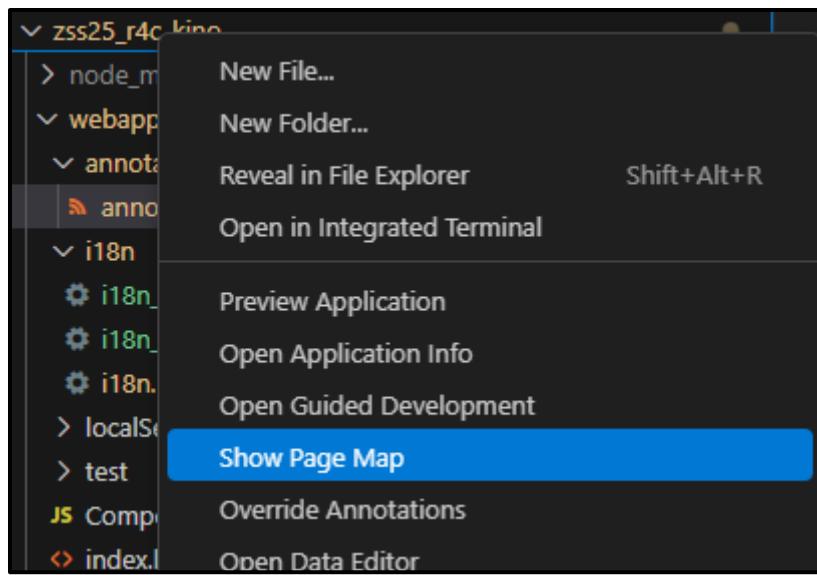
217
218    // RabattKritikalitaet
219    @EndUserText.label: 'Criticality für Rabatt'
220    case
221        when vkf.rabatt = 0 then 2 -- Gelb (neutral)
222        else 3 -- Grün (positive)
223    end as RabattKritikalitaet
224

```

This screenshot shows a table titled 'Verkäufe (15)' displaying sales history. The columns include Employee ..., Sale Date, Visitor Name, Permission, Visitor Group, Ticket Type, Ticket Vali..., Time Cate..., Payment S..., Payment ..., Ticket Price (€), Discount (%), Amount (€), Screening Date, and Price. Each row contains a small icon representing the payment method (e.g., BAR, Cash, Credit Card) and a color-coded circle indicating the criticality of the discount (green for positive, yellow for neutral).

Employee ...	Sale Date	Visitor Name	Permission	Visitor Group	Ticket Type	Ticket Vali...	Time Cate...	Payment S...	Payment ...	Ticket Price (€)	Discount (%)	Amount (€)	Screening Date	Price
	Oct 15, 2025	ANNA MÜLLER	(J) Ja							45.00	10.00	40.50	Nov 2, 2025	
	Sep 27, 2025	ANNA MÜLLER	(J) Ja							40.00	0.00	40.00	Nov 2, 2025	
	Oct 25, 2025	ANNA MÜLLER	(J) Ja							40.00	5.00	38.00	Nov 2, 2025	
	Oct 14, 2025	OMAR ALI	(J) Ja							35.00	0.00	35.00	Oct 15, 2025	

Zur Umsetzung der Responsivität der Tabellen wurden die folgenden Schritte durchgeführt:



**Flexible Column Layout** Object Configuration Manifest

The flexible column layout allows users to see more details on the page, and to expand and collapse the screen areas.

Flexible Column Layout

**2-Column Layout**

- Begin-Expanded
- Mid-Expanded

**3-Column Layout**

- Mid-Expanded
- End-Expanded

Employee ...	Sale Date	Price
	Sep 27, 2025	40.00 >
	Oct 25, 2025	40.00 >
	Oct 14, 2025	35.00 >
	Oct 28, 2025	35.00 >
	Oct 12, 2025	35.00 >
	Oct 12, 2025	35.00 >
	Oct 17, 2025	25.00 >
	Oct 15, 2025	25.00 >
	Oct 23, 2025	25.00 >
	Oct 22, 2025	20.00 >
	Oct 11, 2025	15.00 >
	Oct 15, 2025	20.00 >
	Oct 27, 2025	15.00 >

**Clara Schäfer**  
9008

**Allg. Informationen** **Vorstellung** **Film** **Verkäufer** **Kino** **Raum** **Besucher** **Ticket**

**Kauf**

Employee Photo:   
Sale Date: Oct 15, 2025

**Besucher**

Visitor Name: ANNA MÜLLER  
Permission: (sa)  
Visitor Group:

**Ticket**

Ticket Type: VIP  
Ticket Validity:   
Seat Number: 1  
Seat Row: 1

**Zahlung**

Time Category:   
Payment Status:   
Payment Method:

Zur Konfiguration der Auswahlmöglichkeiten innerhalb einer Tabelle werden die folgenden Schritte durchgeführt:

## ZSS25\_R4C\_C\_VERKF\_A3List

Path: /ZSS25\_R4C\_C\_VERKF\_A3

### Table

#### Initial Load

String

Configuration

Manifest



Determines whether the data in the table is automatically loaded.

- Auto (default): Data is loaded automatically if a default filter value has been set in the filter bar.
- Enabled: Data is loaded automatically, as defined by the standard variant
- Disabled: Data is not loaded automatically. Users have to click the Go button.

Enabled



## ZSS25\_R4C\_C\_VERKF\_A3List

Path: /ZSS25\_R4C\_C\_VERKF\_A3

### Table

#### Selection Mode

String

Configuration

Manifest



Allows you to enable or disable row selection and choose between single or multiple row selection.

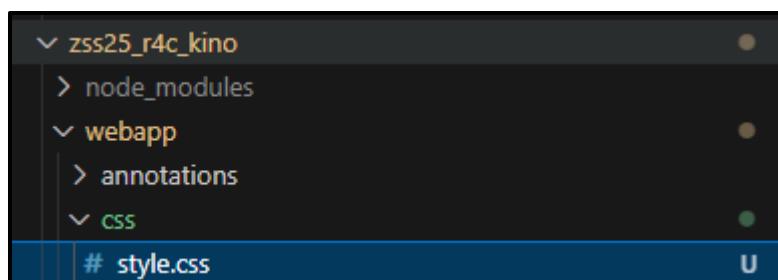
- Auto: This type is deprecated. Choose any of the following modes:
- Multi (default): This type allows multiple table selections if relevant actions are available in the toolbar.
- Single: This type allows single table selection if relevant actions are available in the toolbar.
- None: No table selection is possible in display mode.

Single



## 6.2. Fiori CSS-Einbindung in Fiori-Anwendungen

CSS bietet die Möglichkeit, das Erscheinungsbild von SAP Fiori-Anwendungen individuell anzupassen und so Layout, Farben und Schriftarten nach Bedarf zu gestalten. Zur Einbindung von CSS in die Fiori-Anwendung werden die nachfolgenden Schritte umgesetzt:



JS Component.js M X

```
zss25_r4c_kino > webapp > JS Component.js > ...
1 sap.ui.define(["sap/fe/core/AppComponent"], function(Component) {
2     "use strict";
3     return Component.extend("com.zss25.learnr4c.kinoverwaltung.zss25r4ckino.Component", {
4         metadata: { manifest: "json" },
5         init: function() {
6             Component.prototype.init.apply(this, arguments);
7             // CSS einbinden
8             jQuery.sap.includeStyleSheet(jQuery.sap.getModulePath("com.zss25.learnr4c.kinoverwaltung.zss25r4ckino.css.style"));
9         }
10    });
11});
12});
```

{ manifest.json M X

```
zss25_r4c_kino > webapp > {} manifest.json > {} sap.ui5 > {} resources > [ ] css
69     "sap.ui5": {
85         "models": {
92             "": {
100                "@i18n": {
101                    type: "sap.ui.model.resource.ResourceModel",
102                    uri: "i18n/i18n.properties"
103                }
104            },
105        },
106        "resources": {
107            "css": [
108                {
109                    uri: "css/style.css"
110                }
111            ]
112        }
113    }
114};
```

# style.css U X

```
zss25_r4c_kino > webapp > css > # style.css > .sapFAvatarImageHolder
1 .sapFAvatarImageHolder {
2     display: inline-block;
3     animation: rotateAvatar 5s linear infinite;
4 }
5
6 @keyframes rotateAvatar {
7     from {
8         transform: rotate(0deg);
9     }
10    to {
11        transform: rotate(360deg);
12    }
13 }
14 }
```

Nach den vorgenommenen CSS-Anpassungen drehen sich alle Icons in der Fiori-Anwendung:

Verkäufe (15)														
Employee ...	Sale Date	Visitor Name	Permission	Visitor Group	Ticket Type	Ticket Vali...	Time Cate...	Payment S...	Payment ...	Ticket Price (€)	Discount (%)	Amount (€) $\equiv$	Screening Date	Price
	Oct 15, 2025	ANNA MÜLLER	( Ja)							45.00	10.00	40.50	Nov 2, 2025	45.00 >
	Sep 27, 2025	ANNA MÜLLER	( Ja)							40.00	0.00	40.00	Nov 2, 2025	40.00 >
	Oct 25, 2025	ANNA MÜLLER	( Ja)							40.00	5.00	38.00	Nov 2, 2025	40.00 >
	Oct 14, 2025	OMAR ALI	( Ja)							35.00	0.00	35.00	Oct 15, 2025	35.00 >
	Oct 28, 2025	MARCO ROSSI	( Ja)							35.00	0.00	35.00	Oct 18, 2025	35.00 >
	Oct 12, 2025	MARCO ROSSI	( Ja)							35.00	0.00	35.00	Oct 25, 2025	35.00 >
	Oct 12, 2025	SOPHIE WAGNER	( Ja)							35.00	5.00	33.25	Oct 12, 2025	35.00 >
	Oct 17, 2025	SOPHIE WAGNER	( Ja)							25.00	0.00	25.00	Oct 12, 2025	25.00 >
	Oct 15, 2025	SOPHIE WAGNER	( Ja)							25.00	0.00	25.00	Oct 25, 2025	25.00 >

Verkäufe (15)														
Employee ...	Sale Date	Visitor Name	Permission	Visitor Group	Ticket Type	Ticket Vali...	Time Cate...	Payment S...	Payment ...	Ticket Price (€)	Discount (%)	Amount (€) $\equiv$	Screening Date	Price
	Oct 15, 2025	ANNA MÜLLER	( Ja)							45.00	10.00	40.50	Nov 2, 2025	45.00 >
	Sep 27, 2025	ANNA MÜLLER	( Ja)							40.00	0.00	40.00	Nov 2, 2025	40.00 >
	Oct 25, 2025	ANNA MÜLLER	( Ja)							40.00	5.00	38.00	Nov 2, 2025	40.00 >
	Oct 14, 2025	OMAR ALI	( Ja)							35.00	0.00	35.00	Oct 15, 2025	35.00 >
	Oct 28, 2025	MARCO ROSSI	( Ja)							35.00	0.00	35.00	Oct 18, 2025	35.00 >
	Oct 12, 2025	MARCO ROSSI	( Ja)							35.00	0.00	35.00	Oct 25, 2025	35.00 >
	Oct 12, 2025	SOPHIE WAGNER	( Ja)							35.00	5.00	33.25	Oct 12, 2025	35.00 >
	Oct 17, 2025	SOPHIE WAGNER	( Ja)							25.00	0.00	25.00	Oct 12, 2025	25.00 >
	Oct 15, 2025	SOPHIE WAGNER	( Ja)							25.00	0.00	25.00	Oct 25, 2025	25.00 >
	Oct 23, 2025	SOPHIE WAGNER	( Ja)							25.00	0.00	25.00	Nov 7, 2025	25.00 >

## 7. CRUD-Tests der Kinoverwaltungs-App

### 7.1. Anlegen von Einträgen (Create)

Für Mitarbeiter: Ein neuer Datensatz wird in der Mitarbeiterertabelle angelegt.

Mitarbeiter (7)						
Name	Geburtsdatum	Rolle	Abteilung	Einstellungsdatum	Gehalt	Kontaktinformationen
Kamal Badawi	Jan 15, 1993	Kassierer	VKF1	Oct 1, 2023	2,575.00	Seifgrundstr. 4, 61348 Bad Homburg
Clara Schäfer	Sep 5, 1998	Schichtleiter/in	VKF1	Jan 3, 2016	3,375.00	Louisenstraße 32, 61348 Bad Homburg
Sophie Braun	Sep 5, 1995	Kassierer/in	VKF1	Jan 18, 2016	2,800.00	Kaiser-Friedrich-Promenade 85, 61348 Bad Homburg
Lukas Schneider	May 6, 2000	Kassierer/in	VKF2	Jan 1, 2019	1,950.00	Saalburgstraße 12, 61350 Bad Homburg
Maximilian Hoffmann	May 6, 2001	Schichtleiter/in	VKF2	Jan 1, 2019	3,150.00	Friedrichstraße 7, 61348 Bad Homburg
Anna Müller	May 6, 2005	Kassierer/in	VKF2	Jan 1, 2020	2,500.00	Gartenfeldstraße 22, 61348 Bad Homburg
Lisa Müller	Oct 30, 1997	Kassierer/in	VKF2	Jan 1, 2022	1,980.00	Seifgrundstr. 1, 61348 Bad Homburg

## Create

Mitarbeiter-ID: \*

5008

Name: \*

Harald Schmidt

Geburtsdatum: \*

Aug 6, 1982



Rolle: \*

KASSIERER/IN

Abteilung: \*

VKF1

Einstellungsdatum: \*

Oct 18, 2025



Gehalt: \*

2,650.00

Kontaktinformationen: \*

Am Rathaus 2, 61348 Bad Homburg

**Continue**

**Cancel**

Mitarbeiter (8)							<a href="#">Create</a>	<a href="#">Delete</a>
	Name	Geburtsdatum	Rolle	Abteilung	Einstellungsdatum	Gehalt	Kontaktinformationen	
<input type="checkbox"/>	Kamal Badawi	Jan 15, 1993	Kassierer	VKF1	Oct 1, 2023	2,575.00	Seifgrundstr. 4, 61348 Bad Homburg	>
<input type="checkbox"/>	Clara Schäfer	Sep 5, 1998	Schichtleiter/in	VKF1	Jan 3, 2016	3,375.00	Louisenstraße 32, 61348 Bad Homburg	>
<input type="checkbox"/>	Sophie Braun	Sep 5, 1995	Kassierer/in	VKF1	Jan 18, 2016	2,800.00	Kaiser-Friedrich-Promenade 85, 61348 Bad Homburg	>
<input type="checkbox"/>	Lukas Schneider	May 6, 2000	Kassierer/in	VKF2	Jan 1, 2019	1,950.00	Saalburgstraße 12, 61350 Bad Homburg	>
<input type="checkbox"/>	Maximilian Hoffmann	May 6, 2001	Schichtleiter/in	VKF2	Jan 1, 2019	3,150.00	Friedrichstraße 7, 61348 Bad Homburg	>
<input type="checkbox"/>	Anna Müller	May 6, 2005	Kassierer/in	VKF2	Jan 1, 2020	2,500.00	Gartenfeldstraße 22, 61348 Bad Homburg	>
<input type="checkbox"/>	Lisa Müller	Oct 30, 1997	Kassierer/in	VKF2	Jan 1, 2022	1,980.00	Seifgrundstr. 1, 61348 Bad Homburg	>
<input type="checkbox"/>	Harald Schmidt	Aug 6, 1982	KASSIERER/IN	VKF1	Oct 18, 2025	2,650.00	Am Rathaus 2, 61348 Bad Homburg	>

## Für Kino: Ein neuer Datensatz wird in der Kinotabelle angelegt.

Kinos (5)							<a href="#">Create</a>	<a href="#">Ctrl+Enter</a>
	Name	Adresse	Stadt	PLZ	Anzahl Räume	Eröffnungsdatum	Kontaktinformationen	
<input type="checkbox"/>	RAP4YOU	POTSDAMER PLATZ 1	BAD HOMBURG	10785	5	Dec 1, 2001	01514 1223334	
<input type="checkbox"/>	UCI DXA	AM WEHRHAHN 45	FRIEDBERG	40211	4	Mar 30, 2003	01710 8846622	
<input type="checkbox"/>	MATHÄSER STARS	BAYERSTRÄE 3	MARBURG	80335	6	Dec 8, 2004	01725 8636633	
<input type="checkbox"/>	CINEMINNAXX	DAMMTORSTRÄE 7	LIMBURG	20354	5	Sep 15, 2006	01600 4446644	
<input type="checkbox"/>	FILM ONE	HOHENZOLLERNRING 35	FRANKFURT	50672	3	May 24, 2010	01500 2223355	

## Create

Kino-ID: \*

Name: \*

Adresse: \*

Stadt: \*

PLZ: \*

Anzahl Räume: \*

Eröffnungsdatum: \*



Kontaktinformationen: \*

Kinos (6)						
Name	Adresse	Stadt	PLZ	Anzahl Räume	Eröffnungsdatum	Kontaktinformationen
RAP4YOU	POTSDAMER PLATZ 1	BAD HOMBURG	10785	5	Dec 1, 2001	01514 1223334
UCI DXA	AM WEHRHAHN 45	FRIEDBERG	40211	4	Mar 30, 2003	01710 8846622
MATHÄSER STARS	BAYERSTRÄÙE 3	MARBURG	80335	6	Dec 8, 2004	01725 8636633
CINEMINNAXX	DAMMTORSTRÄÙE 7	LIMBURG	20354	5	Sep 15, 2006	01600 4446644
FILM ONE	HOHENZOLLERNRING 35	FRANKFURT	50672	3	May 24, 2010	01500 2223355
Kino Taunus	Am Bahnhof 12	Bad Homburg	61348	4	Oct 18, 2025	01532 5577456

## 7.2. Bearbeiten von Einträgen (Update)

**Für Mitarbeiter:** Die Rolle des Mitarbeiters wird von Kassierer auf KASSIERER/IN geändert.

Mitarbeiter (8)						
Name	Geburtsdatum	Rolle	Abteilung	Einstellungsdatum	Gehalt	Kontaktinformationen
Kamal Badawi	Jan 15, 1993	Kassierer	VKF1	Oct 1, 2023	2,575.00	Seiggrundstr. 4, 61348 Bad Homburg >
Clara Schäfer	Sep 5, 1998	Schichtleiter/in	VKF1	Jan 3, 2016	3,375.00	Louisenstraße 32, 61348 Bad Homburg >

### Mitarbeiterinformationen

Name:	Rolle:
Kamal Badawi	Kassierer
Geburtsdatum:	Abteilung:
Jan 15, 1993	VKF1

### Mitarbeiterinformationen

Name:	Rolle:
Kamal Badawi	KASSIERER/IN
Geburtsdatum:	Abteilung:
Jan 15, 1993	VKF1

## Mitarbeiterinformationen

Name:  
Kamal Badawi  
Geburtsdatum:  
Jan 15, 1993

Rolle:  
KASSIERER/IN  
Abteilung:  
VKF1

Mitarbeiter (9)							Create	Delete
	Name	Geburtsdatum	Rolle	Abteilung	Einstellungsdatum	Gehalt	Kontaktinformationen	
<input type="checkbox"/>	Kamal Badawi	Jan 15, 1993	KASSIERER/IN	VKF1	Oct 1, 2023	2,575.00	Seifgrundstr. 4, 61348 Bad Homburg	>
<input type="checkbox"/>	Clara Schäfer	Sep 5, 1998	Schichtleiter/in	VKF1	Jan 3, 2016	3,375.00	Louisenstraße 32, 61348 Bad Homburg	>

Für Kino: Der Name des Kinos wird von RAP4YOU\$ auf RAP4YOU geändert.

Kinos (5)							Create	Delete	Import
	Name	Adresse	Stadt	PLZ	Anzahl Räume	Eroffnungsdatum	Kontaktinformationen		
<input type="checkbox"/>	RAP4YOU\$	POTSDAMER PLATZ 1	BAD HOMBURG	10785	5	Dec 1, 2001	01514 1223334	>	

## Kinoinformationen

Name:  
RAP4YOU\$  
Adresse:  
POTSDAMER PLATZ 1

Stadt:  
BAD HOMBURG

PLZ:  
10785

Header	Kinoinformationen
Kinoinformationen	
Name:	RAP4YOU
Adresse:	POTSDAMER PLATZ 1

Kinoinformationen	
Name: RAP4YOU	Stadt: BAD HOMBURG
Adresse: POTSDAMER PLATZ 1	PLZ: 10785

Kinos (5)						
<input type="checkbox"/> Name	Adresse	Stadt	PLZ	Anzahl Räume	Eröffnungsdatum	Kontaktinformationen
<input checked="" type="checkbox"/> RAP4YOU	POTSDAMER PLATZ 1	BAD HOMBURG	10785	5	Dec 1, 2001 01514 1223334	>
<input type="checkbox"/> UCI DXA	AM WEHRHAHN 45	FRIEDBERG	40211	4	Mar 30, 2003 01710 8846622	>

### 7.3. Löschen von Einträgen (Delete)

**Für Mitarbeiter:** Beim Löschen des Mitarbeiters mit der MitarbeiterId 5002 werden aufgrund der relationalen Abhängigkeiten automatisch alle zugehörigen Datensätze in der Tabelle Verkauf gelöscht.

Bevor der Mitarbeiter mit der MitarbeiterId 5002 gelöscht wird, stellt sich der Tabelleninhalt wie folgt dar:

[A07] ZSS25\_R4C\_V\_MITARB\_A1 X [A07] ZSS25\_R4C\_V\_VERKF\_A1

ZSS25\_R4C\_V\_MITARB\_A1 >

Data Preview

find pattern 8 rows retrieved - 34 ms

SQL Console | Nur

MitarbeiterId	Name	FotoURL	Geburtsdatum	Rolle	Abteilung	Einstellungsdatum	Gehalt	Kontaktinfo
0000005001	Kamal Bada...	https://cdn.pi...	1993-01-15	KASSIERER...	VKF1	2023-10-01	2575.00	Seifgrundstr. 4, 61...
0000005002	Clara Schäfer	https://cdn.pi...	1998-09-05	Schichtleiter...	VKF1	2016-01-03	3375.00	Louisenstraße 32, ...
0000005003	Sophie Braun	https://cdn.pi...	1995-09-05	Kassierer/in	VKF1	2016-01-18	2800.00	Kaiser-Friedrich-P...
0000005004	Lukas Schn...	https://cdn.pi...	2000-05-06	Kassierer/in	VKF2	2019-01-01	1950.00	Saalburgstraße 12...
0000005005	Maximilian ...	https://cdn.pi...	2001-05-06	Schichtleiter...	VKF2	2019-01-01	3150.00	Friedrichstraße 7, ...
0000005006	Anna Müller	https://cdn.pi...	2005-05-06	Kassierer/in	VKF2	2020-01-01	2500.00	Gartenfeldstraße ...
0000005007	Lisa Müller	https://cdn.pi...	1997-10-30	Kassierer/in	VKF2	2022-01-01	1980.00	Seifgrundstr. 1, 61...
0000005008	Harald Sch...		1982-08-06	KASSIERER...	VKF1	2025-10-18	2650.00	Am Rathaus 2, 61...

[A07] ZSS25\_R4C\_V\_MITARB\_A1 X [A07] ZSS25\_R4C\_V\_VERKF\_A1

ZSS25\_R4C\_V\_MITARB\_A1 >

Data Preview

find pattern 23 rows retrieved - 36 ms

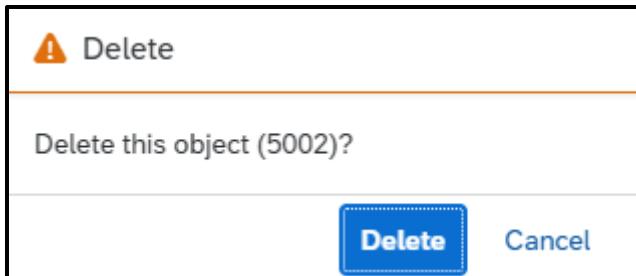
VerkaufId	TicketId	MitarbeiterId	Verkaufsdatum	Zahlungsart	ZahlungsartURL	Rabatt
0000009013	0000008013	0000005001	2025-10-01	KARTE	https://cdn-icons-pn...	15.00
0000009020	0000008020	0000005001	2025-10-15	BAR	https://cdn-icons-pn...	0.00
0000009003	0000008003	0000005001	2025-10-12	KARTE	https://cdn-icons-pn...	10.00
0000009005	0000008005	0000005001	2025-10-14	BAR	https://cdn-icons-pn...	0.00
0000009015	0000008015	0000005001	2025-10-17	BAR	https://cdn-icons-pn...	0.00
0000009011	0000008011	0000005001	2025-10-22	BAR	https://cdn-icons-pn...	0.00
0000009004	0000008004	0000005002	2025-10-01	KARTE	https://cdn-icons-pn...	0.00
0000009008	0000008008	0000005002	2025-10-15	BAR	https://cdn-icons-pn...	10.00
0000009014	0000008014	0000005003	2025-10-13	KARTE	https://cdn-icons-pn...	15.00
0000009001	0000008001	0000005003	2025-10-11	BAR	https://cdn-icons-pn...	0.00
0000009006	0000008006	0000005003	2025-10-12	BAR	https://cdn-icons-pn...	5.00
0000009007	0000008007	0000005004	2025-10-27	BAR	https://cdn-icons-pn...	0.00
0000009017	0000008017	0000005004	2025-10-25	BAR	https://cdn-icons-pn...	5.00
0000009012	0000008012	0000005004	2025-10-30	BAR	https://cdn-icons-pn...	0.00
0000009002	0000008002	0000005005	2025-10-15	BAR	https://cdn-icons-pn...	25.00
0000009018	0000008018	0000005005	2025-09-27	BAR	https://cdn-icons-pn...	0.00
0000009021	0000008021	0000005005	2025-10-12	BAR	https://cdn-icons-pn...	0.00
0000009009	0000008009	0000005006	2025-10-30	KARTE	https://cdn-icons-pn...	5.00
0000009019	0000008019	0000005006	2025-10-23	KARTE	https://cdn-icons-pn...	20.00
0000009023	0000008023	0000005006	2025-10-25	KARTE	https://cdn-icons-pn...	0.00
0000009016	0000008016	0000005007	2025-10-28	BAR	https://cdn-icons-pn...	0.00
0000009022	0000008022	0000005007	2025-10-23	BAR	https://cdn-icons-pn...	0.00
0000009010	0000008010	0000005007	2025-10-02	KARTE	https://cdn-icons-pn...	0.00

Mitarbeiter (8)

Create Delete

Cri-D.

Mitarbeiter-ID	Name	Geburtsdatum	Rolle	Abteilung	Einstellungsdatum	Gehalt	Kontaktinformationen
5001	Kamal Badawi	Jan 15, 1993	KASSIERER/IN	VKF1	Oct 1, 2023	2,575.00	Seifgrundstr. 4, 61348 Bad Homburg
5002	Clara Schäfer	Sep 5, 1998	Schichtleiter/in	VKF1	Jan 3, 2016	3,375.00	Louisenstraße 32, 61348 Bad Homburg
5003	Sophie Braun	Sep 5, 1995	Kassierer/in	VKF1	Jan 18, 2016	2,800.00	Kaiser-Friedrich-Promenade 85, 61348 Bad Homburg
5004	Lukas Schneider	May 6, 2000	Kassierer/in	VKF2	Jan 1, 2019	1,950.00	Saalburgstraße 12, 61350 Bad Homburg
5005	Maximilian Hoffmann	May 6, 2001	Schichtleiter/in	VKF2	Jan 1, 2019	3,150.00	Friedrichstraße 7, 61348 Bad Homburg
5006	Anna Müller	May 6, 2005	Kassierer/in	VKF2	Jan 1, 2020	2,500.00	Gartenfeldstraße 22, 61348 Bad Homburg
5007	Lisa Müller	Oct 30, 1997	Kassierer/in	VKF2	Jan 1, 2022	1,980.00	Seifgrundstr. 1, 61348 Bad Homburg
5008	Harald Schmidt	Aug 6, 1982	KASSIERER/IN	VKF1	Oct 18, 2025	2,650.00	Am Rathaus 2, 61348 Bad Homburg



Nachdem der Mitarbeiter mit der MitarbeiterId 5002 gelöscht wurde, stellt sich der Tabelleninhalt wie folgt dar:

Data Preview										<a href="#">SQL Console</a>	<a href="#">Nur</a>
MitarbeiterId	Name	FotoURL	Geburtsdatum	Rolle	Abteilung	Einstellungsdatum	Gehalt	Kontaktinfo			
0000005001	Kamal Bada...	https://cdn.pi...	1993-01-15	KASSIERER...	VKF1	2023-10-01	2575.00	Seifgrundstr. 4, 61...			
0000005003	Sophie Braun	https://cdn.pi...	1995-09-05	Kassierer/in	VKF1	2016-01-18	2800.00	Kaiser-Friedrich-P...			
0000005004	Lukas Schn...	https://cdn.pi...	2000-05-06	Kassierer/in	VKF2	2019-01-01	1950.00	Saalburgstraße 12...			
0000005005	Maximilian ...	https://cdn.pi...	2001-05-06	Schichtleit...	VKF2	2019-01-01	3150.00	Friedrichstraße 7, ...			
0000005006	Anna Müller	https://cdn.pi...	2005-05-06	Kassierer/in	VKF2	2020-01-01	2500.00	Gartenfeldstraße ...			
0000005007	Lisa Müller	https://cdn.pi...	1997-10-30	Kassierer/in	VKF2	2022-01-01	1980.00	Seifgrundstr. 1, 61...			
0000005008	Harald Sch...		1982-08-06	KASSIERER...	VKF1	2025-10-18	2650.00	Am Rathaus 2, 61...			

Data Preview							<a href="#">SQL Console</a>	<a href="#">Nur</a>
VerkaufId	TicketId	MitarbeiterId	Verkaufsdatum	Zahlungsart	ZahlungsartURL	Rabatt		
0000009005	0000008005	0000005001	2025-10-14	BAR	https://cdn-icons-pn...	0.00		
0000009020	0000008020	0000005001	2025-10-15	BAR	https://cdn-icons-pn...	0.00		
0000009003	0000008003	0000005001	2025-10-12	KARTE	https://cdn-icons-pn...	10.00		
0000009015	0000008015	0000005001	2025-10-17	BAR	https://cdn-icons-pn...	0.00		
0000009011	0000008011	0000005001	2025-10-22	BAR	https://cdn-icons-pn...	0.00		
0000009013	0000008013	0000005001	2025-10-01	KARTE	https://cdn-icons-pn...	15.00		
0000009006	0000008006	0000005003	2025-10-12	BAR	https://cdn-icons-pn...	5.00		
0000009014	0000008014	0000005003	2025-10-13	KARTE	https://cdn-icons-pn...	15.00		
0000009001	0000008001	0000005003	2025-10-11	BAR	https://cdn-icons-pn...	0.00		
0000009012	0000008012	0000005004	2025-10-30	BAR	https://cdn-icons-pn...	0.00		
0000009007	0000008007	0000005004	2025-10-27	BAR	https://cdn-icons-pn...	0.00		
0000009017	0000008017	0000005004	2025-10-25	BAR	https://cdn-icons-pn...	5.00		
0000009002	0000008002	0000005005	2025-10-15	BAR	https://cdn-icons-pn...	25.00		
0000009018	0000008018	0000005005	2025-09-27	BAR	https://cdn-icons-pn...	0.00		
0000009021	0000008021	0000005005	2025-10-12	BAR	https://cdn-icons-pn...	0.00		
0000009019	0000008019	0000005006	2025-10-23	KARTE	https://cdn-icons-pn...	20.00		
0000009023	0000008023	0000005006	2025-10-25	KARTE	https://cdn-icons-pn...	0.00		
0000009009	0000008009	0000005006	2025-10-30	KARTE	https://cdn-icons-pn...	5.00		
0000009016	0000008016	0000005007	2025-10-28	BAR	https://cdn-icons-pn...	0.00		
0000009022	0000008022	0000005007	2025-10-23	BAR	https://cdn-icons-pn...	0.00		
0000009010	0000008010	0000005007	2025-10-02	KARTE	https://cdn-icons-pn...	0.00		

**Für Kino:** Beim Löschen des Kinos mit der Kinold 4005 werden aufgrund der relationalen Abhängigkeiten automatisch alle zugehörigen Datensätze in den Tabellen Raum, Vorstellung und Ticket gelöscht.

Bevor der Kino mit der Kinold 4005 gelöscht wird, stellt sich der Tabelleninhalt wie folgt dar:

Screenshot of the Data Preview window for the 'Kino' table (ZSS25\_R4C\_V\_KINO\_A1). The table contains 6 rows retrieved in 37 ms.

Kinold	Name	Adresse	Stadt	Plz	AnzahlRaeume	Eroeffnungsdatum	Kontaktinfo
0000004001	RAP4YOU	POTSDAMER ...	BAD HOM...	10785	5	2001-12-01	01514 1223334
0000004002	UCI DXA	AM WEHRHA...	FRIEDBERG	40211	4	2003-03-30	01710 8846622
0000004003	MATHÄSER ...	BAYERSTRAB...	MARBURG	80335	6	2004-12-08	01725 8636633
0000004004	CINEMINN...	DAMMTORST...	LIMBURG	20354	5	2006-09-15	01600 4446644
0000004005	FILM ONE	HOHENZOLL...	FRANKFURT	50672	3	2010-05-24	01500 2223355
0000004006	Kino Taunus	Am Bahnhof 12	Bad Homb...	61348	4	2025-10-18	01532 5577456

Screenshot of the Data Preview window for the 'Raum' table (ZSS25\_R4C\_V\_RAUM\_A1). The table contains 9 rows retrieved in 30 ms.

RaumId	Kinold	Name	StuehlAnzahlProReihe	Sitzplaetze	Ausstattung
0000006001	0000004001	SAAL 1	10	120	3D, DOLBY SURR...
0000006002	0000004001	SAAL 2	8	80	2D, DOLBY SURR...
0000006003	0000004002	HAUPTSAAL	13	150	3D, DOLBY ATMOS
0000006004	0000004003	SAAL A	10	100	2D, STANDARD
0000006005	0000004003	SAAL B	6	60	2D, STANDARD
0000006007	0000004004	SAAL 2	9	90	2D, STANDARD
0000006006	0000004004	SAAL 1	12	120	3D, DOBY SURRO...
0000006009	0000004005	KLEINER SA...	7	70	2D, STANDARD
0000006008	0000004005	GROBER SA...	15	200	3D, DOLBY ATMO...

Screenshot of the Data Preview window for the 'Vorstellung' table (ZSS25\_R4C\_V\_VORST\_A1). The table contains 8 rows retrieved in 45 ms.

VorstellungId	RaumId	Kinold	FilmId	Sprache	Datum	Startzeit	Endzeit	Untertitel	VorstellungFormat	Status
0000007001	0000006001	0000004001	0000003001	DE	2025-11-09	02:00:00 PM	06:00:00 PM	VORHANDEN	3D	ABGESAGT
0000007003	0000006002	0000004001	0000003002	DE	2025-11-02	03:00:00 PM	05:00:38 PM	N. V.	3D	AKTIV
0000007002	0000006001	0000004001	0000003008	DE	2025-10-23	05:00:00 PM	06:30:00 PM	VORHANDEN	IMAX	AKTIV
0000007004	0000006003	0000004002	0000003003	DE	2025-10-15	06:00:00 PM	09:00:00 PM	N. V.	IMAX	AKTIV
0000007005	0000006004	0000004003	0000003004	EN	2025-10-12	06:00:00 PM	09:00:00 PM	VORHANDEN	IMAX	AKTIV
0000007006	0000006005	0000004003	0000003005	DE	2025-10-18	06:00:00 PM	08:30:00 PM	VORHANDEN	2D	AKTIV
0000007007	0000006006	0000004004	0000003006	EN	2025-11-07	05:00:00 PM	09:30:00 PM	VORHANDEN	2D	AKTIV
0000007008	0000006008	0000004005	0000003007	DE	2025-10-25	04:00:00 PM	05:30:00 PM	VORHANDEN	3D	AKTIV

[A07] ZSS25\_R4C\_V\_KINO\_A1 [A07] ZSS25\_R4C\_V\_RAUM\_A1 [A07] ZSS25\_R4C\_V\_VORST\_A1 [A07] ZSS25\_R4C\_V\_TICKET\_A1 X  
► ZSS25\_R4C\_V\_TICKET\_A1

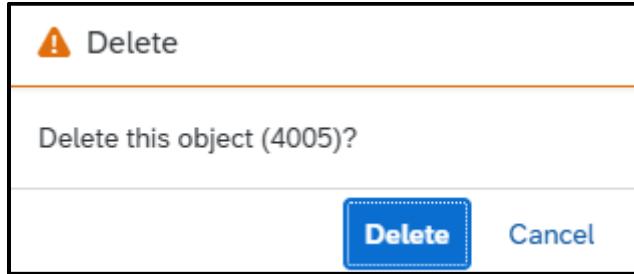
Data Preview find pattern 23 rows retrieved - 53 ms SQL Console Number of

TicketId	VorstellungId	BesucherId	RaumId	Kinold	Sitzplatznummer	Preis	TicketTyp	QrCode	Status
0000008012	0000007003	0000001004	0000006002	0000004001	40	8.00	KIND	HTTPS://GIT...	BEZAHLT
0000008001	0000007001	0000001001	0000006001	0000004001	13	15.00	KIND	HTTPS://GIT...	BEZAHLT
0000008013	0000007001	0000001001	0000006001	0000004001	20	20.00	STANDARD	HTTPS://GIT...	BEZAHLT
0000008009	0000007003	0000001001	0000006002	0000004001	3	40.00	VIP	HTTPS://GIT...	STORNIERT
0000008017	0000007003	0000001001	0000006002	0000004001	4	40.00	VIP	HTTPS://GIT...	BEZAHLT
0000008003	0000007002	0000001003	0000006001	0000004001	34	18.00	STANDARD	HTTPS://GIT...	BEZAHLT
0000008010	0000007001	0000001002	0000006001	0000004001	35	15.00	STANDARD	HTTPS://GIT...	BEZAHLT
0000008011	0000007002	0000001003	0000006001	0000004001	22	20.00	STANDARD	HTTPS://GIT...	BEZAHLT
0000008018	0000007003	0000001001	0000006002	0000004001	5	40.00	VIP	HTTPS://GIT...	BEZAHLT
0000008004	0000007003	0000001004	0000006002	0000004001	2	22.00	STANDARD	HTTPS://GIT...	OFFEN
0000008002	0000007001	0000001002	0000006001	0000004001	32	20.00	STANDARD	HTTPS://GIT...	BEZAHLT
0000008019	0000007002	0000001002	0000006001	0000004001	5	20.00	STANDARD	HTTPS://GIT...	BEZAHLT
0000008008	0000007003	0000001001	0000006002	0000004001	1	45.00	VIP	HTTPS://GIT...	BEZAHLT
0000008014	0000007004	0000001005	0000006003	0000004002	1	45.00	VIP	HTTPS://GIT...	STORNIERT
0000008005	0000007004	0000001005	0000006003	0000004002	2	35.00	VIP	HTTPS://GIT...	BEZAHLT
0000008006	0000007005	0000001006	0000006004	0000004003	1	35.00	VIP	HTTPS://GIT...	BEZAHLT
0000008007	0000007006	0000001007	0000006005	0000004003	5	15.00	KIND	HTTPS://GIT...	BEZAHLT
0000008015	0000007005	0000001006	0000006004	0000004003	2	25.00	VIP	HTTPS://GIT...	BEZAHLT
0000008016	0000007006	0000001007	0000006005	0000004003	3	35.00	VIP	HTTPS://GIT...	OFFEN
0000008023	0000007007	0000001007	0000006006	0000004004	3	25.00	VIP	HTTPS://GIT...	STORNIERT
0000008022	0000007007	0000001006	0000006006	0000004004	2	25.00	VIP	HTTPS://GIT...	BEZAHLT
0000008020	0000007008	0000001006	0000006008	0000004005	1	25.00	VIP	HTTPS://GIT...	STORNIERT
0000008021	0000007008	0000001007	0000006008	0000004005	5	35.00	VIP	HTTPS://GIT...	BEZAHLT

Kinos (6)

Create Delete Ctrl+D

Kino-ID	Name	Adresse	Stadt	PLZ	Anzahl Räume	Eröffnungsdatum	Kontaktinformationen
4001	RAP4YOU	POTSDAMER PLATZ 1	BAD HOMBURG	10785	5	Dec 1, 2001	01514 1223334
4002	UCI DXA	AM WEHRHAHN 45	FRIEDBERG	40211	4	Mar 30, 2003	01710 8846622
4003	MATHASER STARS	BAYERSTRÄÙE 3	MARBURG	80335	6	Dec 8, 2004	01725 8636633
4004	CINEMINNAXX	DAMMTORSTRÄÙE 7	LIMBURG	20354	5	Sep 15, 2006	01600 4446644
4005	FILM ONE	HOHENZOLLERNRING 35	FRANKFURT	50672	3	May 24, 2010	01500 2223355
4006	Kino Taunus	Am Bahnhof 12	Bad Homburg	61348	4	Oct 18, 2025	01532 5577456



Nachdem der Kino mit der Kinold 4005 gelöscht wurde, stellt sich der Tabelleninhalt wie folgt dar:

[A07] ZSS25\_R4C\_V\_KINO\_A1 X [A07] ZSS25\_R4C\_V\_RAUM\_A1 X [A07] ZSS25\_R4C\_V\_VORST\_A1 X [A07] ZSS25\_R4C\_V\_TICKET\_A1

ZSS25\_R4C\_V\_KINO\_A1 ▶

Data Preview

find pattern 5 rows retrieved - 4 ms

Kinold	Name	Adresse	Stadt	Plz	AnzahlRaeume	Eroeffnungsdatum	Kontaktinfo
0000004001	RAP4YOU	POTSDAMER ...	BAD HOM...	10785	5	2001-12-01	01514 122334
0000004002	UCI DXA	AM WEHRHA...	FRIEDBERG	40211	4	2003-03-30	01710 8846622
0000004003	MATHÄSER ...	BAYERSTRAB...	MARBURG	80335	6	2004-12-08	01725 8636633
0000004004	CINEMINN...	DAMMTORST...	LIMBURG	20354	5	2006-09-15	01600 4446644
0000004006	Kino Taunus	Am Bahnhof 12	Bad Homb...	61348	4	2025-10-18	01532 5577456

[A07] ZSS25\_R4C\_V\_KINO\_A1 X [A07] ZSS25\_R4C\_V\_RAUM\_A1 X [A07] ZSS25\_R4C\_V\_VORST\_A1 X [A07] ZSS25\_R4C\_V\_TICKET\_A1

ZSS25\_R4C\_V\_RAUM\_A1 Schließen

Data Preview

find pattern 7 rows retrieved - 3 ms

RaumId	Kinold	Name	StuehlAnzahlProReihe	Sitzplaetze	Ausstattung
0000006001	0000004001	SAAL 1	10	120	3D, DOLBY SURR...
0000006002	0000004001	SAAL 2	8	80	2D, DOLBY SURR...
0000006003	0000004002	HAUPTSAAL	13	150	3D, DOLBY ATMOS
0000006004	0000004003	SAAL A	10	100	2D, STANDARD
0000006005	0000004003	SAAL B	6	60	2D, STANDARD
0000006007	0000004004	SAAL 2	9	90	2D, STANDARD
0000006006	0000004004	SAAL 1	12	120	3D, DOBY SURRO...

[A07] ZSS25\_R4C\_V\_KINO\_A1 [A07] ZSS25\_R4C\_V\_RAUM\_A1 [A07] ZSS25\_R4C\_V\_VORST\_A1 X [A07] ZSS25\_R4C\_V\_TICKET\_A1

ZSS25\_R4C\_V\_VORST\_A1 ▶

Data Preview

find pattern 7 rows retrieved - 29 ms

VorstellungId	RaumId	Kinold	FilmId	Sprache	Datum	Startzeit	Endzeit	Untertitel	VorstellungFormat	Status
0000007002	0000006001	0000004001	0000003008	DE	2025-10-23	05:00:00 PM	06:30:00 PM	VORHANDEM	IMAX	AKTIV
0000007001	0000006001	0000004001	0000003001	DE	2025-11-09	02:00:00 PM	06:00:00 PM	VORHANDEM	3D	ABGESAGT
0000007003	0000006002	0000004001	0000003002	DE	2025-11-02	03:00:00 PM	05:00:38 PM	N. V.	3D	AKTIV
0000007004	0000006003	0000004002	0000003003	DE	2025-10-15	06:00:00 PM	09:00:00 PM	N. V.	IMAX	AKTIV
0000007006	0000006005	0000004003	0000003005	DE	2025-10-18	06:00:00 PM	08:30:00 PM	VORHANDEM	2D	AKTIV
0000007005	0000006004	0000004003	0000003004	EN	2025-10-12	06:00:00 PM	09:00:00 PM	VORHANDEM	IMAX	AKTIV
0000007007	0000006006	0000004004	0000003006	EN	2025-11-07	05:00:00 PM	09:30:00 PM	VORHANDEM	2D	AKTIV

TicketId	VorstellungId	BesucherId	RaumId	Kinold	Sitzplatznummer	Preis	TicketTyp	OrCode	Status
0000008013	0000007001	0000001001	0000006001	0000004001	20	20.00	STANDARD	HTTPS://GIT...	BEZAHLT
0000008004	0000007003	0000001004	0000006002	0000004001	2	22.00	STANDARD	HTTPS://GIT...	OFFEN
0000008018	0000007003	0000001001	0000006002	0000004001	5	40.00	VIP	HTTPS://GIT...	BEZAHLT
0000008011	0000007002	0000001003	0000006001	0000004001	22	20.00	STANDARD	HTTPS://GIT...	BEZAHLT
0000008017	0000007003	0000001001	0000006002	0000004001	4	40.00	VIP	HTTPS://GIT...	BEZAHLT
0000008010	0000007001	0000001002	0000006001	0000004001	35	15.00	STANDARD	HTTPS://GIT...	BEZAHLT
0000008019	0000007002	0000001002	0000006001	0000004001	5	20.00	STANDARD	HTTPS://GIT...	BEZAHLT
0000008012	0000007003	0000001004	0000006002	0000004001	40	8.00	KIND	HTTPS://GIT...	BEZAHLT
0000008009	0000007003	0000001001	0000006002	0000004001	3	40.00	VIP	HTTPS://GIT...	STORNIERT
0000008001	0000007001	0000001001	0000006001	0000004001	13	15.00	KIND	HTTPS://GIT...	BEZAHLT
0000008003	0000007002	0000001003	0000006001	0000004001	34	18.00	STANDARD	HTTPS://GIT...	BEZAHLT
0000008002	0000007001	0000001002	0000006001	0000004001	32	20.00	STANDARD	HTTPS://GIT...	BEZAHLT
0000008008	0000007003	0000001001	0000006002	0000004001	1	45.00	VIP	HTTPS://GIT...	BEZAHLT
0000008014	0000007004	0000001005	0000006003	0000004002	1	45.00	VIP	HTTPS://GIT...	STORNIERT
0000008005	0000007004	0000001005	0000006003	0000004002	2	35.00	VIP	HTTPS://GIT...	BEZAHLT
0000008006	0000007005	0000001006	0000006004	0000004003	1	35.00	VIP	HTTPS://GIT...	BEZAHLT
0000008016	0000007006	0000001007	0000006005	0000004003	3	35.00	VIP	HTTPS://GIT...	OFFEN
0000008007	0000007006	0000001007	0000006005	0000004003	5	15.00	KIND	HTTPS://GIT...	BEZAHLT
0000008015	0000007005	0000001006	0000006004	0000004003	2	25.00	VIP	HTTPS://GIT...	BEZAHLT
0000008023	0000007007	0000001007	0000006006	0000004004	3	25.00	VIP	HTTPS://GIT...	STORNIERT
0000008022	0000007007	0000001006	0000006006	0000004004	2	25.00	VIP	HTTPS://GIT...	BEZAHLT

## 8. Herausforderungen und Lösungsansätze

### 8.1. Durchführung von CRUD-Operationen in hierarchischen Datenmodellen

Wie in Abbildung 13 dargestellt, besteht zwischen den Tabellen Kino und Raum eine 1:n-Beziehung. Ebenso steht die Tabelle Raum in einer 1:n-Beziehung zur Tabelle Vorstellung. Das bedeutet: Ein Kino verfügt über mehrere Räume, und in jedem Raum finden mehrere Vorstellungen statt. Da Raum die Root-Tabelle von Vorstellung ist und Kino wiederum die Root-Tabelle von Raum, kann Kino somit als übergeordnete Root-Tabelle von Vorstellung betrachtet werden. Beim Ausführen von CRUD-Operationen (Create, Read, Update, Delete) sollen Änderungen in der Tabelle Kino auch Auswirkungen auf die abhängigen Datensätze in Vorstellung haben - und umgekehrt. Um diesen Zusammenhang korrekt abzubilden, wird in der Basis-View eine Join-Operation zwischen den Tabellen Raum und Vorstellung verwendet. Dadurch kann das Feld Kinold (Fremdschlüssel aus Raum) auch in die Tabelle Vorstellung übernommen werden. Darüber hinaus besteht zwischen Vorstellung und Ticket ebenfalls eine 1:n-Beziehung: Eine Vorstellung kann mehrere Tickets haben, und jedes Ticket gehört genau zu einer Vorstellung. Analog zu den vorherigen Beziehungen werden durch entsprechende Join-Operationen in der Basis-View auch die Felder RaumId und Kinold in die Tabelle Ticket übernommen, da Raum und Kino übergeordnete Root-Tabellen von Ticket sind. Diese Join-Beziehungen werden in der Basis-View (ZSS25\_R4C\_V\_VORST\_A1) und (ZSS25\_R4C\_V\_TICKET\_A1) wie folgt umgesetzt:

```

define view entity ZSS25_R4C_V_VORST_A1 as select from zss25_r4c_vorst as vorst
inner join zss25_r4c_raum as rm on vorst.raum_id = rm.raum_id
{
    key vorst.vorstellung_id as VorstellungId,
    vorst.raum_id as RaumId,
    rm.kino_id      as KinoId,
}

define view entity ZSS25_R4C_V_TICKET_A1 as select from zss25_r4c_ticket as tkt
inner join zss25_r4c_vorst as vor on vor.vorstellung_id = tkt.vorstellung_id
inner join zss25_r4c_raum as rm  on rm.raum_id = vor.raum_id
{
    key tkt.ticket_id as TicketId,
    tkt.vorstellung_id as VorstellungId,
    tkt.besucher_id as BesucherId,
    rm.raum_id as RaumId,
    rm.kino_id as KinoId,
}

```

In der Behavior Definition (ZSS25\_R4C\_R\_KINO\_A2) wird die folgende Implementierung verwendet, um die zuvor beschriebene Beziehung zwischen den Tabellen und die konsistenten CRUD-Operationen über mehrere Ebenen hinweg zu realisieren:

B [A07] ZSS25\_R4C\_R\_KINO\_A2 X

```
1 managed implementation in class zbp_ss25_r4s_r_kino_a2 unique;
2 strict ( 2 );
3 with draft;
4@ define behavior for ZSS25_R4C_R_KINO_A2 alias Kino
5 persistent table ZSS25_R4C_KINO
6 draft table ZSS25_R4C_DKINO
7 lock master total etag LastChangedAt
8 authorization master ( instance )
9 etag master LocalLastChangedAt
10 {
11     create;
12     update;
13     delete;
14
15     draft action Activate;
16     draft action Discard;
17     draft action Edit;
18     draft action Resume;
19     draft determine action Prepare;
20
21     field ( mandatory: create ) KINOID;
22     field ( mandatory: create ) Name;
23
24
25@ mapping for ZSS25_R4C_KINO corresponding
26 {
27     KinoId = kino_id;
28     Name = name;
29
30     Stadt = stadt;
31     Plz = plz;
32     AnzahlRaeume = anzahl_raeume;
33     Eroeffnungsdatum = eroeffnungsdatum;
34     Kontaktinfo = kontaktinfo;
35     CreatedAt = created_at;
36     CreatedBy = created_by;
37     LastChangedAt = last_changed_at;
38     LastChangedBy = last_changed_by;
39     LocalLastChangedAt = local_last_changed_at;
40 }
```

```

41    association _Raum { create; with draft; }
42
43
44
45 }
46
47
48 // ****
49 // ****
50 // ****
51 // ****
52 // ****
53@define behavior for ZSS25_R4C_R_RAUM_A2 alias Raum
54 persistent table ZSS25_R4C_RAUM
55 draft table ZSS25_R4C_DRAUM
56 lock dependent by _Kino
57 authorization dependent by _Kino
58 etag master LocalLastChangedAt
59 {
60     update;
61     delete;
62     field ( readonly ) KinoId;
63     field ( mandatory: create ) RaumId;
64
65@ mapping for ZSS25_R4C_RAUM corresponding
66 {
67
68     RaumId = raum_id;
69     KinoId = kino_id;
70     Name = name;
71     StuehlAnzahlProReihe = stuehl_anzahl_pro_reihe;
72     Sitzplaetze = sitzplaetze;
73     Ausstattung = ausstattung;
74     LocalLastChangedAt = local_last_changed_at;
75 }
76

```

```

77 validation checkRaum on save { create; update; delete; }
78 association _Kino { with draft; }
79 association _Vorstellung { create; with draft; }
80 }
81
82 // ****
83 // ****
84 // ****
85 // ****
86 // ****
87 // ****
88@ define behavior for ZSS25_R4C_R_VORST_A2 alias Vorstellung
89 persistent table ZSS25_R4C_VORST
90 draft table ZSS25_R4C_DVORST
91 lock dependent by _Kino
92 authorization dependent by _Kino
93 etag master LocalLastChangedAt
94 {
95     update;
96     delete;
97     field ( readonly ) KinId;
98     field ( readonly ) RaumId;
99     field ( mandatory: create ) VorstellungId;
100
101@ mapping for ZSS25_R4C_VORST corresponding
102 {
103
104     VorstellungId = vorstellung_id;
105     RaumId = raum_id;
106     FilmId = film_id;
107     Sprache = sprache;
108     Datum = datum;
109     Startzeit = startzeit;
110     Endzeit = endzeit;
111     Untertitel = untermittel;
112     VorstellungFormat = vorstellung_format;
113     Status = status;
114     LocalLastChangedAt    = local_last_changed_at;
115 }
```

```

116
117     validation checkVorstellung on save { create; update; delete; }
118     association _Raum { with draft; }
119     association _Kino { with draft; }
120     association _Ticket { create; with draft; }
121 }
122
123
124 // ****
125 // ****
126 // ****
127 // ****
128 // ****
129 // ****
130 // ****
131 // ****
132 // ****
133 // ****
134 @define behavior for ZSS25_R4C_R_TICKET_A2 alias Ticket
135 persistent table ZSS25_R4C_TICKET
136 draft table ZSS25_R4C_DTICK
137 lock dependent by _Kino
138 authorization dependent by _Kino
139 etag master LocalLastChangedAt
140 {
141     update;
142     delete;
143     field ( readonly ) KinoId;
144
145     field ( readonly ) RaumId;
146     field ( readonly ) VorstellungId;
147     field ( mandatory: create ) TicketId;
148
149 @ mapping for ZSS25_R4C_TICKET corresponding
150 }
```

```

150 {
151     BesucherId      = besucher_id;
152     TicketId        = ticket_id;
153     Sitzplatznummer = sitzplatznummer;
154     Preis            = preis;
155     TicketTyp       = ticket_typ;
156     QrCode          = qr_code;
157     Status           = status;
158     LocalLastChangedAt = local_last_changed_at;
159 }
160
161 validation checkTicket on save { create; update; delete; }
162 association _Kino { with draft; }
163 association _Raum { with draft; }
164 association _Vorstellung { with draft; }
165
166 }
```

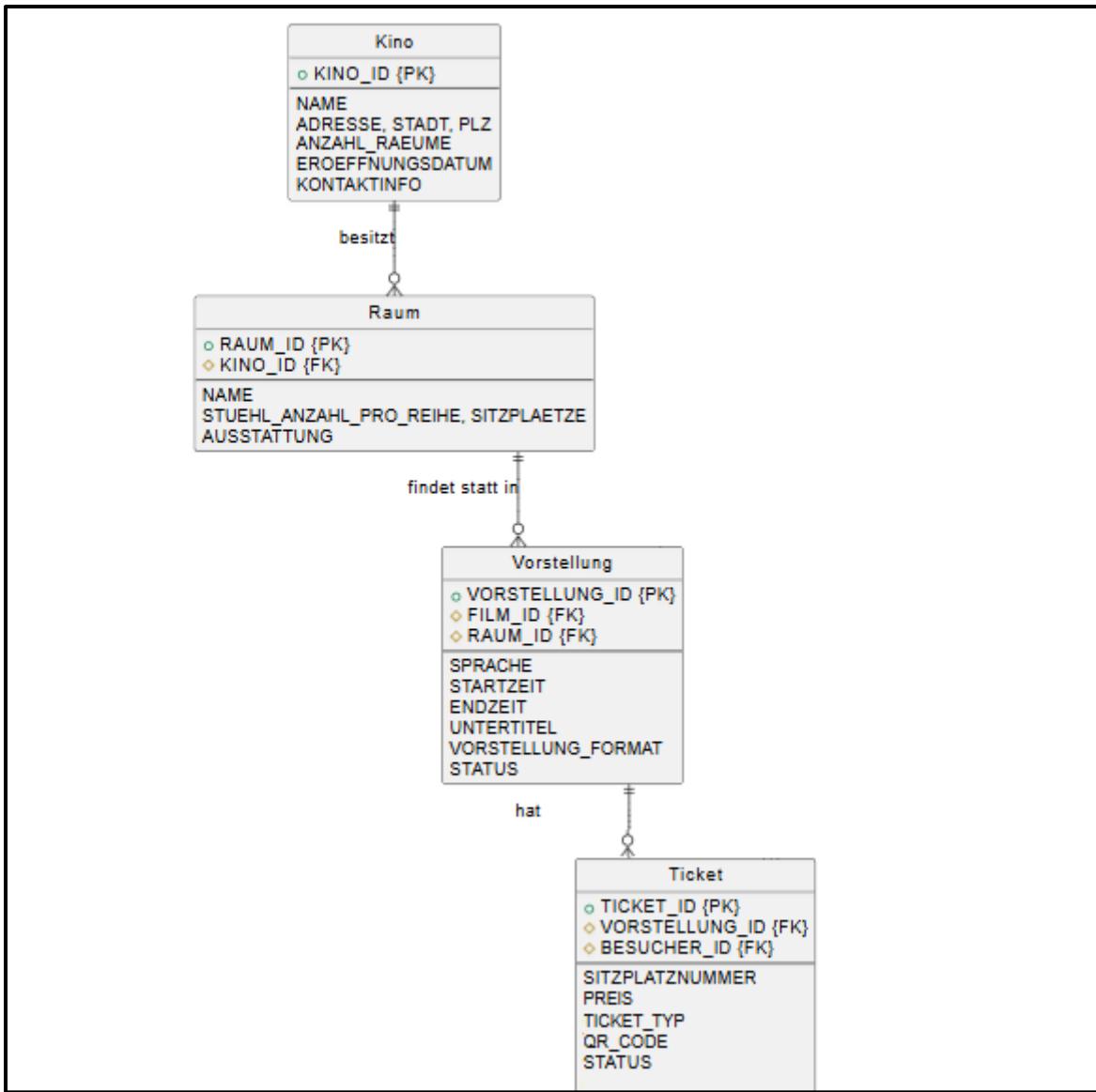


Abbildung 13: ER-Modell Kino - Raum - Vorstellung - Ticket

## 8.2. Prinzip der eindeutigen Root-Zuordnung in hierarchischen Datenmodellen

Im ABAP RESTful Application Programming Model (RAP) gilt die Regel, dass jede Child-Tabelle genau einer Root-Tabelle untergeordnet sein muss. Dies bedeutet jedoch nicht, dass eine Tabelle nur eine einzige Beziehung zu anderen Tabellen besitzen darf. Sie kann durchaus mehrere Beziehungen (z. B. Assoziationen oder Fremdschlüsselverknüpfungen) zu verschiedenen Entitäten aufweisen. Entscheidend ist, dass sie innerhalb der Datenhierarchie des RAP-Modells nur eine übergeordnete Root-Beziehung hat, über die ihre CRUD-Operationen (Create, Read, Update, Delete) ausgeführt und verwaltet werden. Diese eindeutige Root-Zuordnung ist notwendig, damit das Framework beim Persistieren von Änderungen (z. B. über save oder modify) eindeutig bestimmen kann, über welche Root-Instanz die Operation propagiert werden

soll. Wird eine Tabelle mehreren Root-Entitäten gleichzeitig als Child zugeordnet, führt dies zu Ambiguitäten und Fehlern im Laufzeitverhalten. Um dies zu vermeiden, sollte bei der Modellierung von Reusable Views sichergestellt werden, dass jede View nur eine Root-Beziehung abbildet. Falls eine Tabelle in unterschiedlichen Kontexten wiederverwendet werden soll, empfiehlt sich die Erstellung mehrerer spezialisierter Reusable Views, die jeweils klar einer Root-Tabelle zugeordnet sind.

Im folgenden Codeabschnitt wird das Prinzip exemplarisch dargestellt. Die Entität Vorstellung fungiert dabei als Root-Entität der Entität Ticket. Zwischen Ticket und Besucher besteht eine 1:1-Beziehung, während zwischen Vorstellung und Ticket eine 1:n-Beziehung existiert. Das bedeutet, dass eine Vorstellung mehrere Tickets umfassen kann, jedes Ticket jedoch genau einer Vorstellung und genau einem Besucher zugeordnet ist. Zudem besitzt Ticket Assoziationen zu den Entitäten Kino und Raum.

```

1 @AbapCatalog.viewEnhancementCategory: [#NONE]
2 @AccessControl.authorizationCheck: #NOT_REQUIRED
3 @EndUserText.label: 'Reusable View für Ticket'
4 @Metadata.ignorePropagatedAnnotations: true
5 @ObjectModel.usageType:{
6     serviceQuality: #X,
7     sizeCategory: #S,
8     dataClass: #MIXED
9 }
10 define view entity ZSS25_R4C_R_TICKET_A2
11   as select from ZSS25_R4C_V_TICKET_A1
12
13
14 association [1..1] to ZSS25_R4C_R_BESUCH_A2 as _Besucher
15   on $projection.BesucherId = _Besucher.BesucherId
16
17 association to parent ZSS25_R4C_R_VORST_A2 as _Vorstellung
18   on $projection.VorstellungId = _Vorstellung.VorstellungId
19
20 association to ZSS25_R4C_R_KINO_A2 as _Kino
21   on $projection.KinoId = _Kino.KinoId
22
23 association to ZSS25_R4C_R_RAUM_A2 as _Raum
24   on $projection.RaumId = _Raum.RaumId
25
26 {
27   key TicketId,
28   VorstellungId,
29   BesucherId,
30   RaumId,
31   KinoId,
```

## 8.3. Einschränkungen bei der Unterstützung von Datentypen im Service Binding

Bei der Nutzung des Service Bindings im ABAP RESTful Application Programming Model (RAP) ist zu beachten, dass nicht alle Datentypen vollständig unterstützt werden. Ein typisches Beispiel tritt auf, wenn innerhalb einer CDS-View eine berechnete Kennzahl definiert wird, deren Ergebnis in der zugehörigen Draft-Tabelle als Datentyp

decfloat34 gespeichert ist. Der Datentyp decfloat34 wird vom Service Binding nicht unterstützt, was beim Aktivieren oder Ausführen des Services zu einem Fehler führt. Die Lösung besteht darin, den betreffenden Ausdruck mithilfe der Funktion CAST in einen kompatiblen Datentyp (z. B. abap.dec(16,2) oder abap.curr(15,2)) umzuwandeln. Dadurch wird sichergestellt, dass der Wert korrekt verarbeitet und vom Service Binding ohne Fehler interpretiert werden kann.

### Falsches Beispiel:

```
// !!!!!!!!!!!!!!!!!!!!!!!  
// ***** FALSCH *****  
// RaumAnzahlReihen  
@EndUserText.label: 'RaumAnzahlReihen'  
ceil( rm.Sitzplaetze / rm.StuehlAnzahlProReihe ) as RaumAnzahlReihen  
  
{@AbapCatalog.decfloat.outputStyle : #NORMAL  
raumanzahlreihen      : abap.decfloat34;}
```



Application could not be started due to  
technical issues.

/sap/opu/odata4/sap/zss25\_r4c\_sb\_kino\_a5/srvd/sap/zss25\_r4c\_sd  
\_kino\_a4/0001/\$metadata: Unsupported OData version 4.01

### Richtiges Beispiel:

```
// !!!!!!!!!!!!!!!!!!!!!!!!
// ***** RICHTIG *****
// RaumAnzahlReihen
@EndUserText.label: 'RaumAnzahlReihen'
cast( ceil( rm.Sitzplaetze / rm.StuehlAnzahlProReihe ) as abap.int4 ) as RaumAnzahlReihen

raumanzahlreihen      : abap.int4;
```

```
// Zuzahlender Betrag
cast( tkt.Preis * (1 - (vkf.rabatt / 100)) as abap.dec(15,2) ) as Betrag,
```

```
// BesucherSitzReihe  
@EndUserText.label: 'BesucherSitzReihe'  
cast( ceil( tkt.Sitzplatznummer / rm.StuehlAnzahlProReihe ) as abap.int4 ) as BesucherSitzReihe,
```

## 9. GitHub-Integration mit abapGit

### 9.1. SSL-Zertifikat im SAP-System konfigurieren

Damit die Verbindung zu GitHub funktioniert, muss das entsprechende SSL-Zertifikat im SAP-System hinterlegt werden. Andernfalls kann folgender Fehler auftreten:

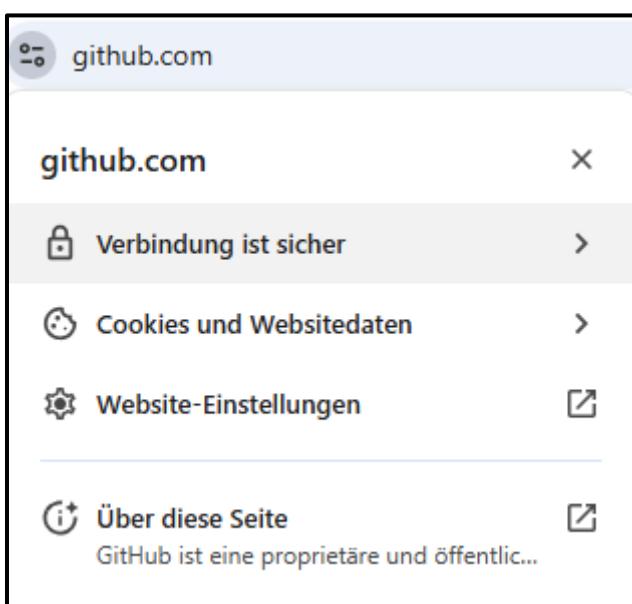
**!** HTTP error 421 occurred: SSL handshake with github.com:443 failed: SSSLERR\_PEER\_CERT\_UNTRUSTED (-102) Peer's X.509 certificate (chain) validation failed (missing trust?)  
SapSSLSessionStartNB() == SSSLE

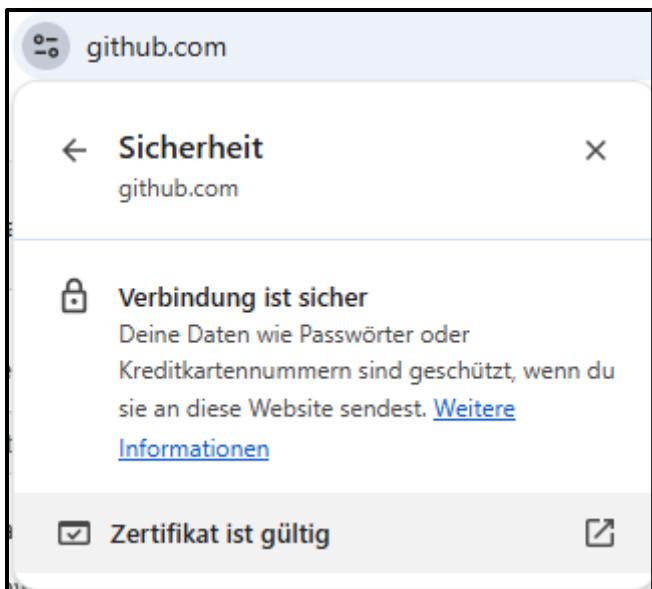


#### Schritt-für-Schritt-Anleitung (Installation):

##### 1. Zertifikat herunterladen:

- Seite aufrufen: <https://github.com/>
- Auf das Schlosssymbol in der Adresszeile klicken
- Verbindung ist sicher auswählen
- Zertifikat ist gültig → Details → Exportieren
- Zielordner (z. B. Downloads) auswählen
- Datei speichern





## Zertifikats-Viewer: github.com

X

Allgemein

Details

### Zertifikathierarchie

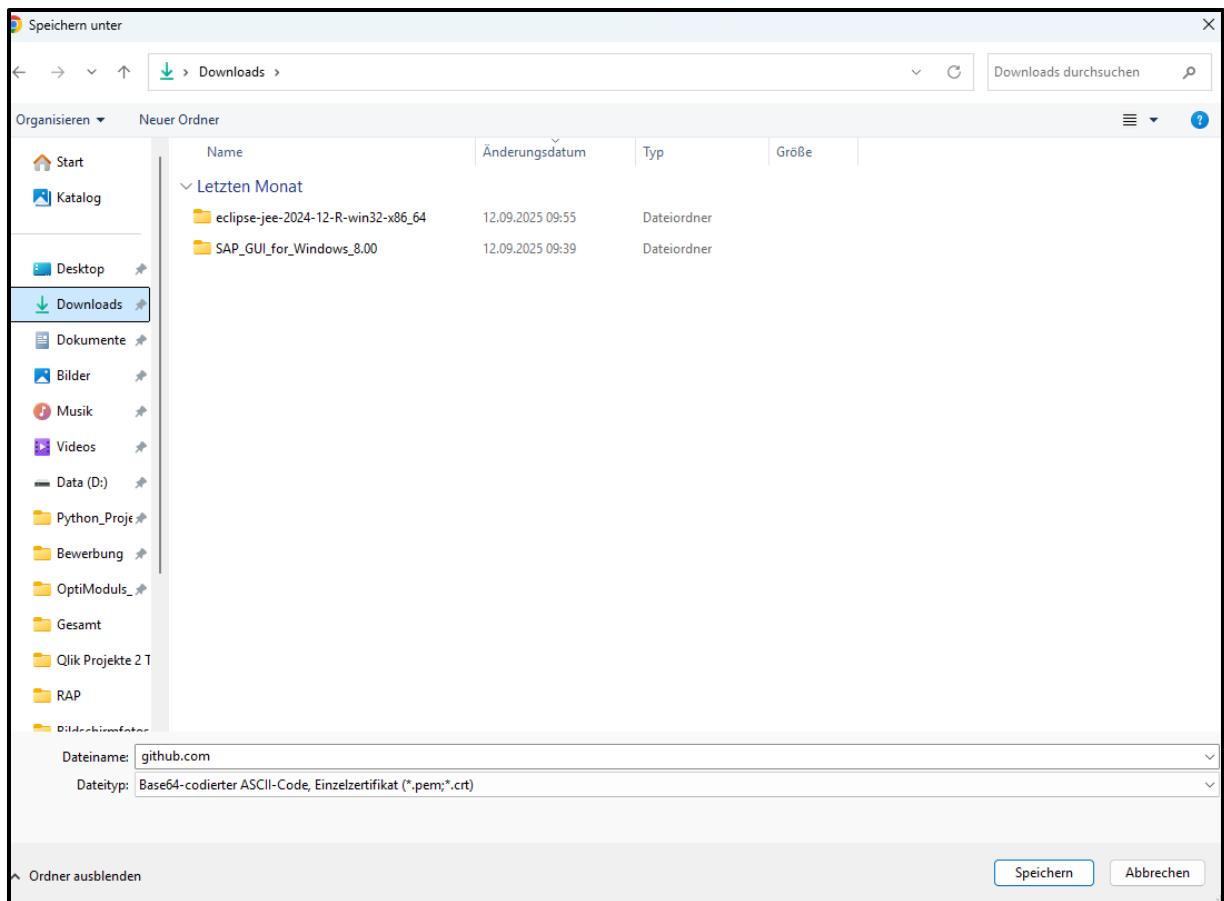
- ▼ USERTrust ECC Certification Authority
  - ▼ Sectigo ECC Domain Validation Secure Server CA
    - github.com

### Zertifikatfelder

- ▼ github.com
  - ▼ Zertifikat
    - Version
    - Seriенnummer
    - Algorithmus für Zertifikatsignatur
    - Aussteller
  - ▼ Gültigkeit
    - Nicht vor

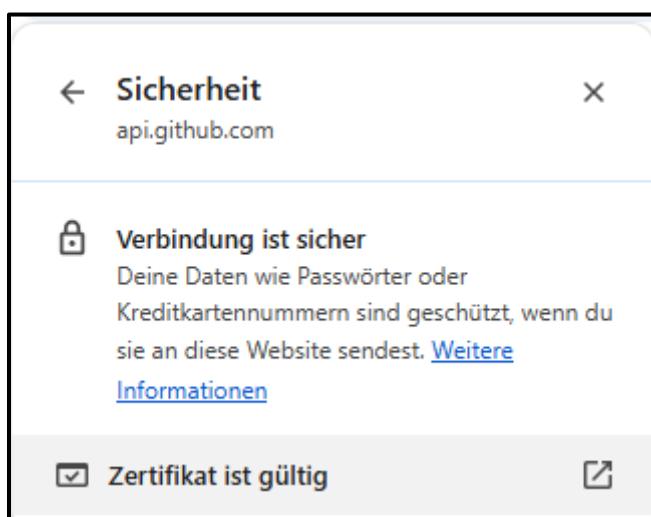
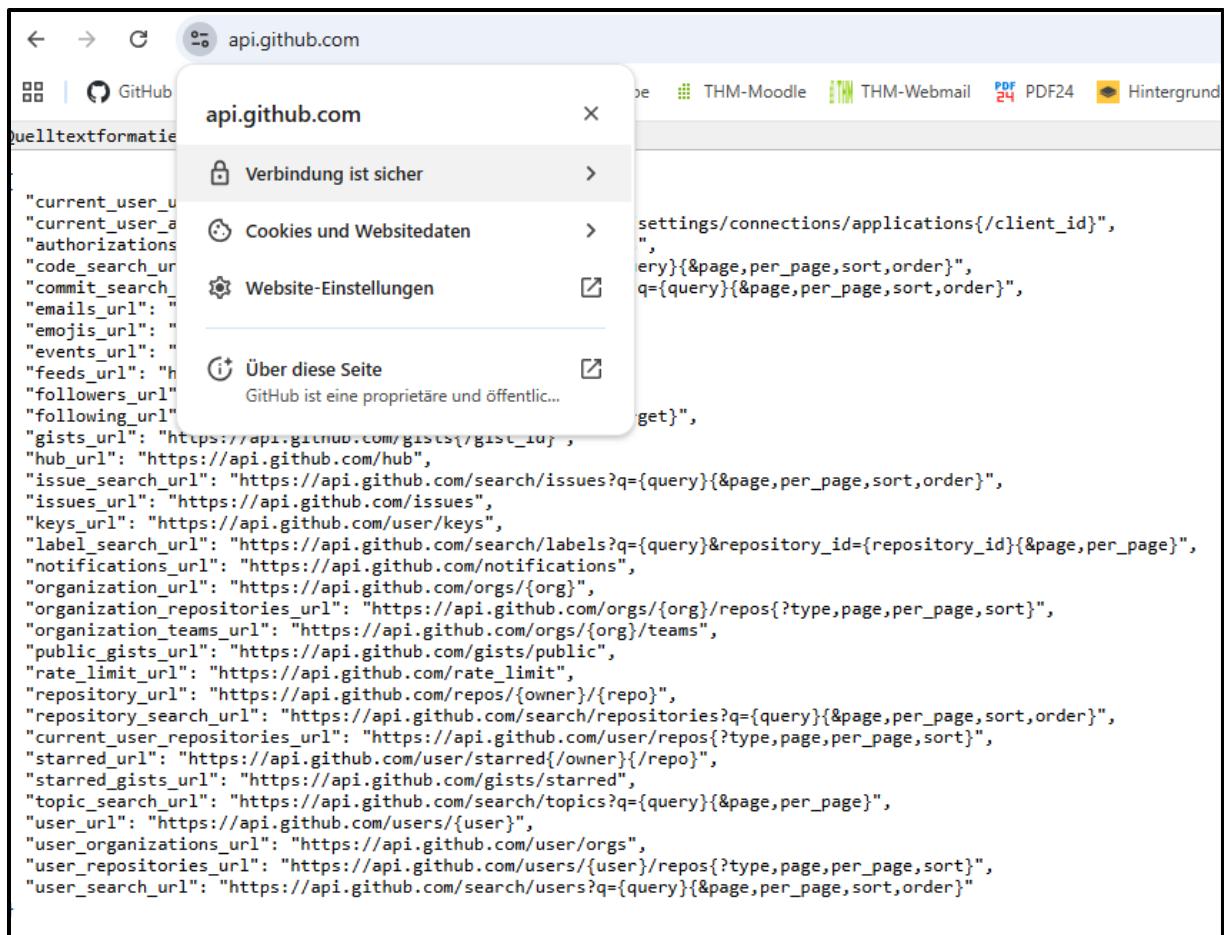
### Feldwert

Exportieren...



## 2. Zertifikat herunterladen:

- Seite aufrufen: <https://api.github.com/>
- Auf das Schlosssymbol in der Adresszeile klicken
- Verbindung ist sicher auswählen
- Zertifikat ist gültig → Details → Exportieren
- Zielordner (z. B. Downloads) auswählen
- Datei speichern



Zertifikats-Viewer: \*.github.com X

Allgemein Details

Zertifikathierarchie

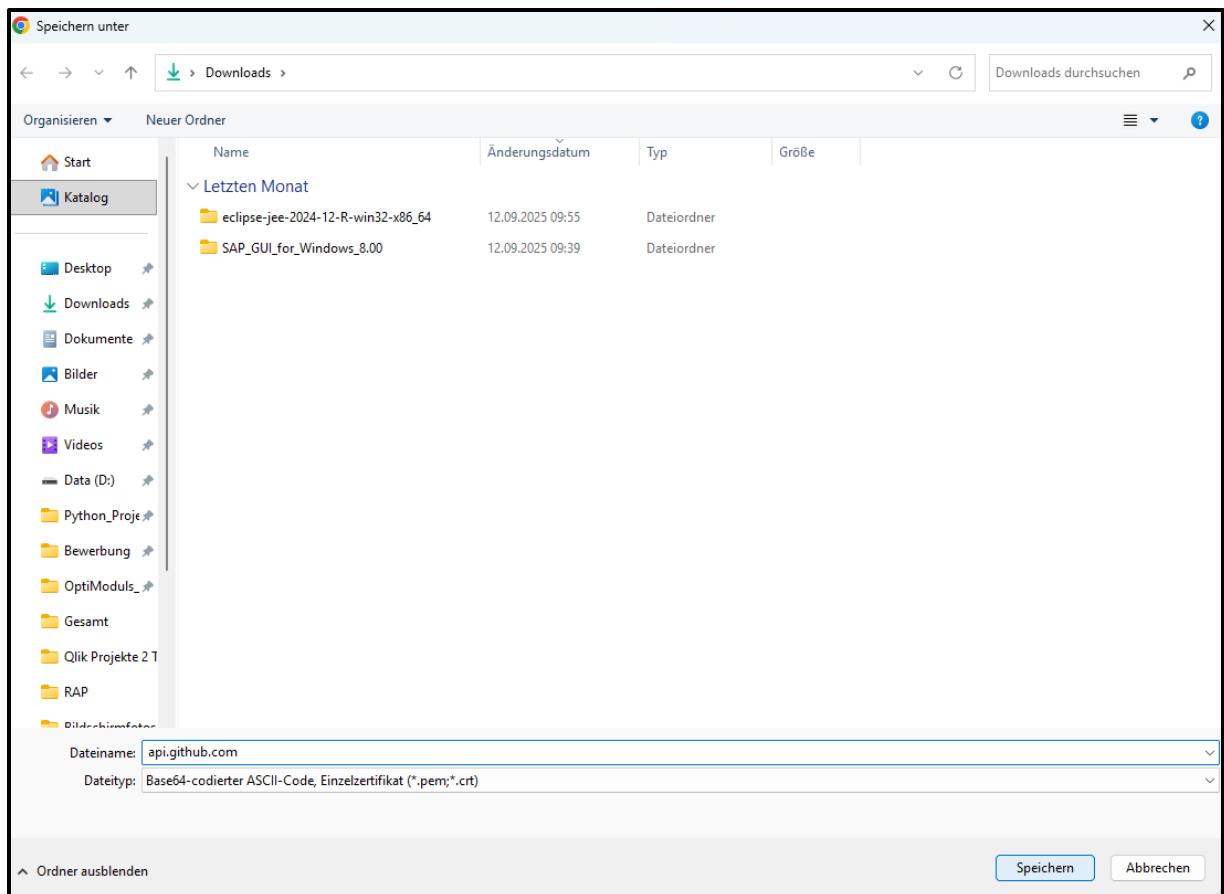
- ▼ USERTrust ECC Certification Authority
  - ▼ Sectigo ECC Domain Validation Secure Server CA
    - \*.github.com

Zertifikatfelder

- ▼ \*.github.com
  - ▼ Zertifikat
    - Version
    - Seriенnummer
    - Algorithmus für Zertifikatsignatur
    - Aussteller
  - ▼ Gültigkeit
    - Nicht vor

Feldwert

Exportieren...



### 3. Zertifikate im SAP-System hochladen

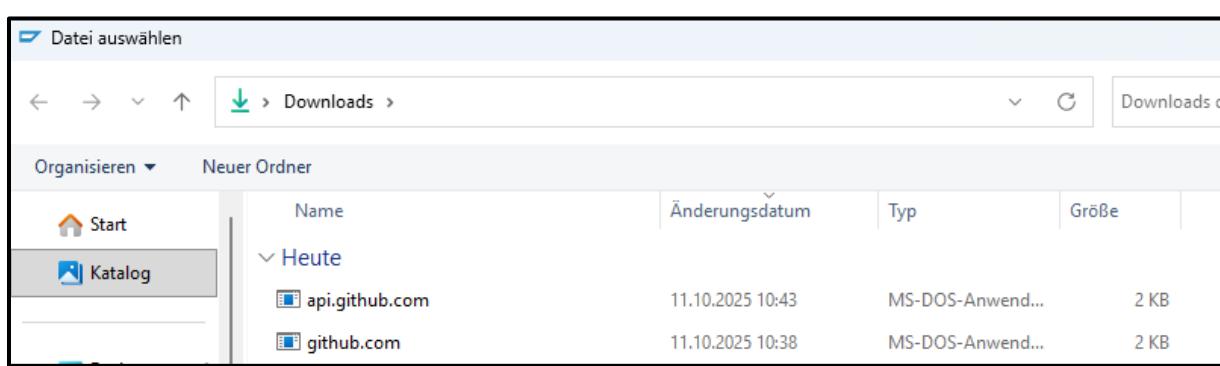
- SAP Logon öffnen
- Transaktion /nSTRUST eingeben
- SSL Client (Anonymous) auswählen
- Zertifikat importieren
- Die zuvor heruntergeladenen Zertifikate einzeln hochladen
- In den Bearbeitungsmodus (Edit Mode) wechseln
- Zertifikate in die Zertifikatsliste aufnehmen
- Sichern
- Überprüfen, ob die Zertifikate nun in der Zertifikatsliste angezeigt werden

The screenshot shows the SAP Fiori Launchpad interface. The top navigation bar includes the SAP logo, a search bar with the text '/instrust', and various icons for navigation and settings. Below the navigation bar is a sidebar titled 'Favoriten' containing links to Data Browser, ABAP Dictionary Pflege, Object Navigator, ABAP Class Builder, and SAP Fiori Launchpad. The main content area displays a hierarchical list of security profiles. The 'SSL-Client SSL-Client (Anonym)' profile is expanded, showing two sub-items: 'a07z\_A07\_07' and 'a07z\_A07\_08'. Other collapsed profiles include System-PSE, SSL-Server Standard, SSL-Client BCM, SSL-Client SSL-Client (Standard), and several WS-Security profiles.

- > System-PSE
- ✗ SNC SAPCryptolib
- > SSL-Server Standard
- < SSL-Client SSL-Client (Anonym)
  - a07z\_A07\_07
  - a07z\_A07\_08
- ✗ SSL-Client BCM
- > SSL-Client SSL-Client (Standard)
- ✗ SSL-Client PAYPAL
- ✗ WS-Security Standard
- ✗ WS-Security Other System Encryption C
- ✗ WS-Security WS Security Keys
- ✗ SMIME Standard
- Datei
- SSF Anmeldeticket

Zertifikat

Inhaber:	<input type="text"/>		
Inhaber (alt.):	<input type="text"/>		
Aussteller:	<input type="text"/>		
Seriennummer (hex.):	<input type="text"/>		
Seriennummer (dez.):	<input type="text"/>		
Gültig von:	<input type="text"/> / <input type="text"/>	bis:	<input type="text"/> / <input type="text"/>
Algorithmus:	<input type="text"/>		
Schlüsselstärke:	<input type="text"/>		
Signaturalgorithmus:	<input type="text"/>		
Prüfsumme (MD5):	<input type="text"/>		
Prüfsumme (SHA1):	<input type="text"/>		



Zertifikat

Inhaber:	CN=github.com
Inhaber (alt.):	dNSName=www.github.com, dNSName=github.com
Aussteller:	CN=Sectigo ECC Domain Validation Secure Server CA, O=Sectigo Limited, L=Salford, SP=Greater Manchester, C=GB
Seriennummer (hex.):	AB:66:86:B5:62:7B:E8:05:96:82:13:30:12:86:49:F5
Seriennummer (dez.):	227830333772454795620750445496253172213
Gültig von:	02/05/2025
	00:00:00
bis:	02/05/2026
	23:59:59
Algorithmus:	ECDSA
Schlüsselstärke:	P-256
Signaturalgorithmus:	ECDSA+SHA256
Prüfsumme (MD5):	6F:23:AF:B5:8B:3C:5C:0E:2A:D3:02:A3:AB:F0:D9:BE
Prüfsumme (SHA1):	E4:33:71:DD:D6:91:4A:75:B6:1F:9E:4F:74:6D:9B:F0:DD:26:FC:3A

In Zert.-Liste aufnehmen



Zertifikat

Inhaber:	CN=github.com		
Inhaber (alt.):	dNSName=www.github.com, dNSName=github.com		
Aussteller:	CN=Sectigo ECC Domain Validation Secure Server CA, O=Sectigo Limited, L=Salford, SP=Greater Manchester, C=GB		
Seriennummer (hex.):	AB:66:86:B5:62:7B:E8:05:96:82:13:30:12:86:49:F5		
Seriennummer (dez.):	227830333772454795620750445496253172213		
Gültig von:	02/05/2025	00:00:00	bis: 02/05/2026 23:59:59
Algorithmus:	ECDSA		
Schlüsselstärke:	P-256		
Signaturalgorithmus:	ECDSA+SHA256		
Prüfsumme (MD5):	6F:23:AF:B5:8B:3C:5C:0E:2A:D3:02:A3:AB:F0:D9:BE		
Prüfsumme (SHA1):	E4:33:71:DD:D6:91:4A:75:B6:1F:9E:4F:74:6D:9B:F0:DD:26:FC:3A		
<a href="#"></a> <a href="#"></a> <a href="#"></a> <a href="#"></a> <a href="#">In Zert.-Liste aufnehmen</a>			

Zertifikatsliste

Inhaber	Gültig ab	Gültig bis
CN=Certificate Authority, O=CS.UCC.MD	08/21/2025	07/17/2045
EMAIL=hostmaster@ucc.md, CN=ucc.md, OU=UCP1 2025, O=SAP UCC Magdeburg, L=Magdeburg, SP=S...	08/21/2025	08/16/2045
CN=github.com	02/05/2025	02/05/2026
CN=*.github.com	02/05/2025	02/05/2026

## 9.2. Erstellung eines Personal Access Tokens in GitHub

Um abapGit und GitHub miteinander zu verknüpfen, wird für die Anmeldung ein Personal Access Token benötigt.

### Schritt-für-Schritt-Anleitung:

- GitHub öffnen
- Auf das Profilbild klicken
- Settings auswählen
- Developer Settings öffnen
- Personal Access Tokens → Tokens (classic) wählen
- Auf Generate new token klicken
- Eine Beschreibung eingeben
- repo aktivieren (ankreuzen)

- Auf Generate token klicken
- Den Token kopieren und sicher aufbewahren

New personal access token (classic)

Personal access tokens (classic) function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to [authenticate to the API over Basic Authentication](#).

**Note**

AbapGit

What's this token for?

**Expiration**

90 days (Jan 09, 2026) ▾  
The token will expire on the selected date

**Select scopes**

Scopes define the access for personal tokens. [Read more about OAuth scopes.](#)

<input checked="" type="checkbox"/> repo	Full control of private repositories
<input type="checkbox"/> repo:status	Access commit status
<input type="checkbox"/> repo_deployment	Access deployment status
<input type="checkbox"/> public_repo	Access public repositories
<input type="checkbox"/> repo:invite	Access repository invitations
<input type="checkbox"/> security_events	Read and write security events

## 9.3. Vorgehensweise zur Installation von abapGit im SAP-System

Mit abapGit können ABAP-Entwicklungsobjekte einfach mit GitHub-Repositories synchronisiert werden. Bevor Projekte aus GitHub in ein SAP-System importiert oder exportiert werden können, muss abapGit im System installiert und aktiviert werden. Die folgende Anleitung beschreibt den gesamten Installationsprozess Schritt für Schritt.

### Schritt-für-Schritt-Anleitung:

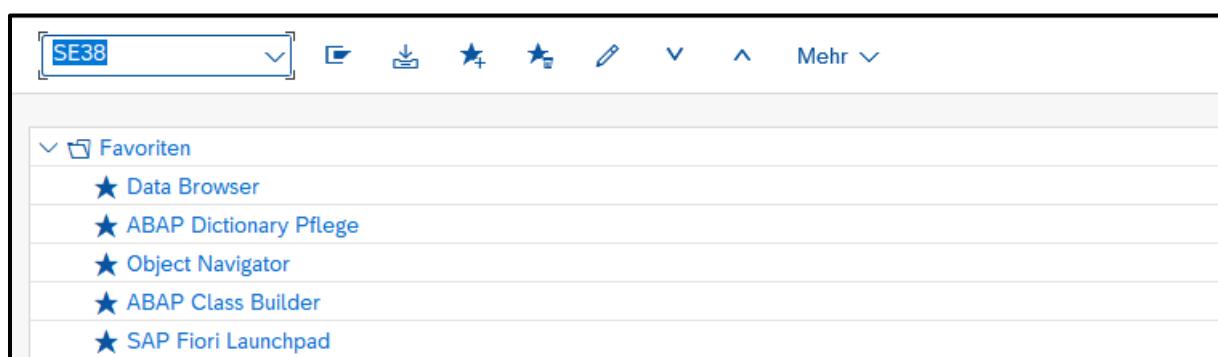
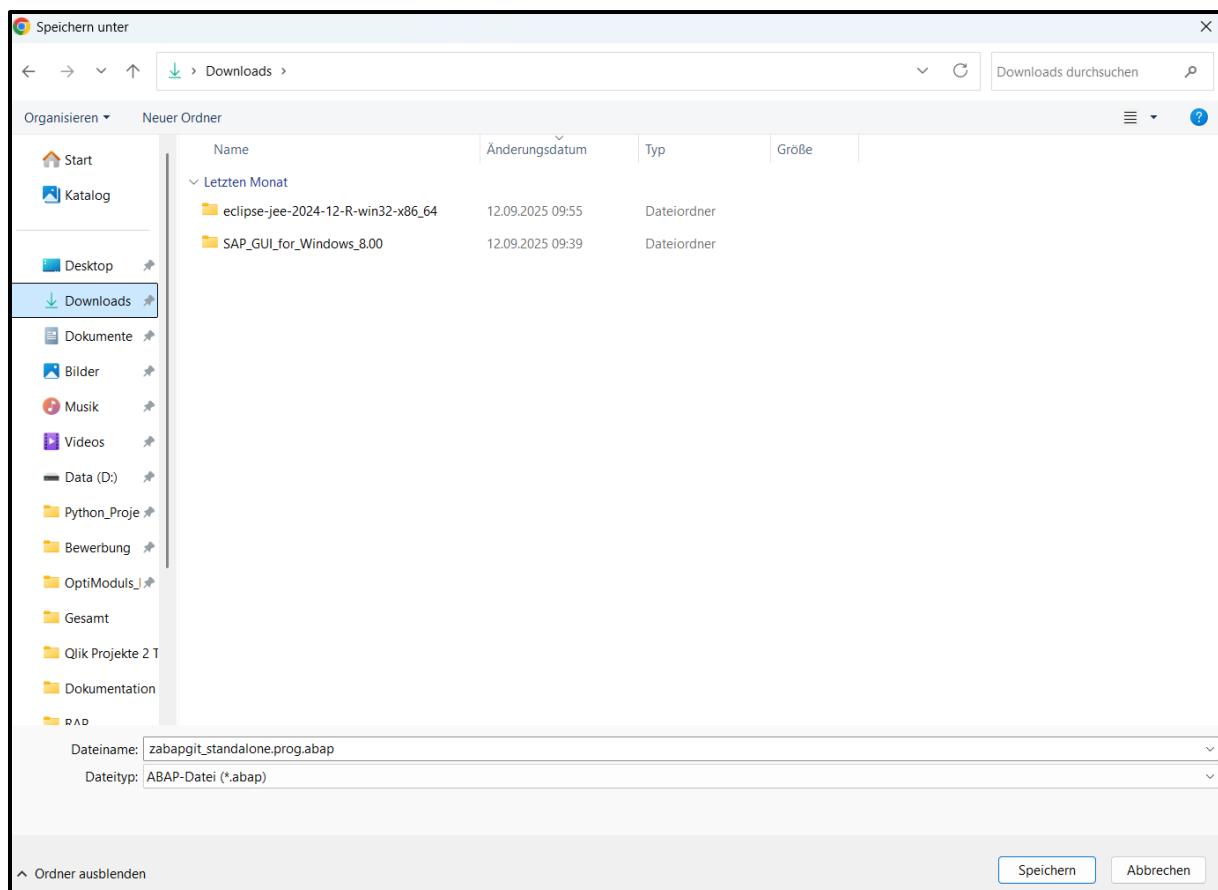
- Den folgenden Link öffnen: <https://docs.abapgit.org/user-guide/getting-started/install.html>
- Rechtsklick auf ABAP code → Link speichern unter ...
- Einen Download-Ordner auswählen und die Datei speichern

- Im SAP Logon die Transaktion SE38 öffnen
- Ein neues Programm mit dem Namen ZABAPGIT anlegen
- Eine Beschreibung eingeben
- Typ: Ausführbares Programm auswählen
- Programm sichern und Lokales Objekt auswählen
- Das Programm öffnet sich nun automatisch
- Menü: Hilfsmittel → Weitere Hilfsmittel → Upload/Download → Upload
- Die zuvor gespeicherte abapGit-Datei hochladen
- Nachdem der Quellcode in das SAP-System geladen wurde, das Programm aktivieren
- Das Programm ausführen (Taste F8)
- Im abapGit-Startbildschirm „New Online“ auswählen
- Die gewünschte GitHub Repository-URL eingeben, z. B. <https://github.com/kamal-badawi/SAP-RAP-RAP4CINEMA.git>
- Ein Package auswählen, z. B. \$SS25\_RAP4CINEMA
- Auf Create Online Repo klicken
- Bei der Anmeldung den GitHub-Benutzernamen und den zuvor erstellten Personal Access Token eingeben

## Install Standalone Version

1. Download the [ABAP code](#) ↗ (right click -> save-as) to a file.
  2. Via [SE38](#), [SE80](#), or [ADT](#) → [ZABAPGIT\\_FULL](#). Note: Depending on your system configuration, you may need to change the version.
  3. In source code change mode → [File](#) → [Upload/Download](#) → [Upload](#)
  4. Activate
- Typically, abapGit will only be installed in a local system, so it can be installed in a local \$ABAPGIT package (e.g. `$ABAPGIT`).
- 
- ZABAPGIT\_STANDALONE (formerly ABAPGIT)  
if you plan to install the developer  
file using Utilities -> More Utilities ->

Now you can use abapGit by executing the report in transaction [SE38](#).





## ABAP: Programmeigenschaften ZABAPGIT ändern



Titel: Git mit ABAP

Originalsprache: DE

Erstellt: LEARN-195

10/11/2025

Letzte Änderung:

Status: neu(überarbeitet)

## Attribute

Typ: Ausführbares Programm

Status: Unklassifiziert

Berechtigungsgruppe:

Anwendung:

Name der LDB:

Selektionsbild:

ABAP-Sprachversion: Standard-ABAP

 Festpunktarithmetik Editorsperre Start über Variante

Sichern

**☰ Objektkatalogeintrag anlegen**

Objekt: R3TR PROG ZABAPGIT

**Attribute**

Paket:	
Verantwortlicher:	<u>LEARN-195</u>
Originalsystem:	A07
Originalsprache:	DE Deutsch
Anlegedatum:	

Lokales Objekt Sperrübersicht X

SAP ABAP Editor: Report ZABAPGIT

Report: ZABAPGIT

```

REPORT ZABAPGIT.

```

Hilfsmittel   Umfeld   System   Hilfe

- Einstellungen
- Objektliste anzeigen (Strg+Umsch+F5)
- Arbeitsvorrat
- Navigationsfenster anzeigen (Strg+Umsch+F4)
- Navigationsindex aktualisieren
- Hilfe zu ... (Strg+F8)
- Breakpoints
- Block/Ablage
- Weitere Hilfsmittel >
- VerwendungsNachweis (Strg+Umsch+F3)
- Versionen

Split-Screen-Editor

Laufzeitanalyse

Muster bearbeiten >

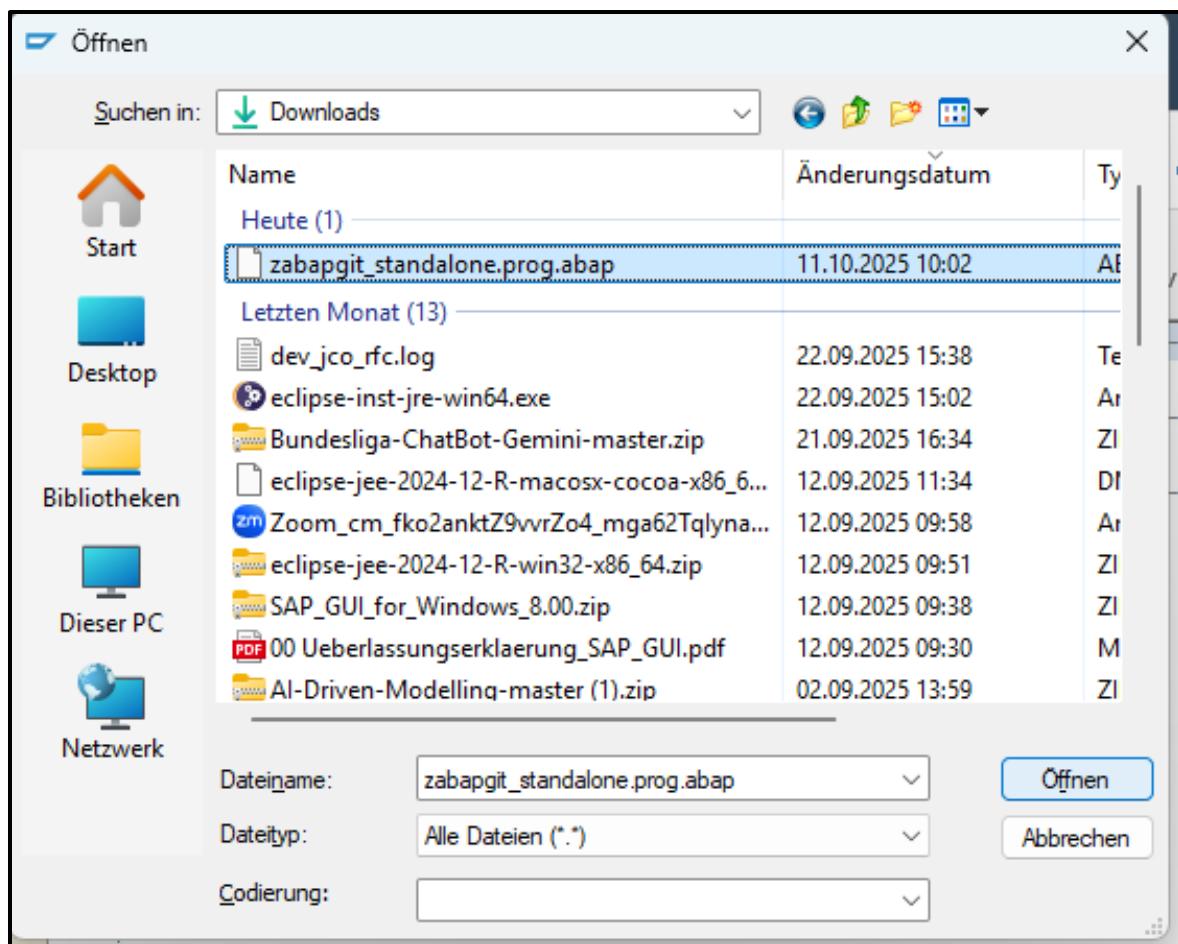
Upload/Download > Upload...

Rahmenprogramme

Zeilenlänge 72 erzwingen

Generierte Tests löschen

Code Completion >



Muster

Aktivieren (Strg+F3)      inaktiv(überarbeitet)

Report: ZABAPGIT

```

1 REPORT zabapgit_standalone LINE-SIZE 100.
2
3 * See http://www.abapgit.org
4
5  *****
6 * The MIT License (MIT)
7 *
8 * Copyright (c) 2014 abapGit Contributors
9 *
10 * Permission is hereby granted, free of charge, to any person obtaining a copy
11 * of this software and associated documentation files (the "Software"), to deal
12 * in the Software without restriction, including without limitation the rights
13 * to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
14 * copies of the Software, and to permit persons to whom the Software is
15 * furnished to do so, subject to the following conditions:
16 *
17 * The above copyright notice and this permission notice shall be included in all
18 * copies or substantial portions of the Software.
19 *
20 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
21 * IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
22 * FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
23 * AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
24 * LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
25 * OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
26 * SOFTWARE.
27 *****
28
29 INTERFACE zif_abapgit_version DEFERRED.
30 INTERFACE zif_abapgit_definitions DEFERRED.
31 INTERFACE zif_abapgit_xml_output DEFERRED.
32 INTERFACE zif_abapgit_xml_input DEFERRED.
33 INTERFACE zif_abapgit_log DEFERRED.
34 INTERFACE zif_abapgit_popups DEFERRED.
35 INTERFACE zif_abapgit_frontend_services DEFERRED.
36 INTERFACE zif_abapgit_services_repo DEFERRED.
37 INTERFACE zif_abapgit_services_git DEFERRED.
38 INTERFACE zif_abapgit_progress DEFERRED.
39 INTERFACE zif_abapgit_gui_diff_extra DEFERRED.
40 INTERFACE zif_abapgit_gui_diff DEFERRED.
41 INTERFACE zif_abapgit_html_table DEFERRED.
42 INTERFACE zif_abapgit_html_popup DEFERRED.
43 INTERFACE zif_abapgit_html_form DEFERRED.

```

```

1 REPORT zabapgit_standalone LINE-SIZE 100.
2
3 * See http://www.abapgit.org
4
5 *-----*
6 * The MIT License (MIT)
7 *
8 * Copyright (c) 2014 abapGit Contributors
9
10 * Permission is hereby granted, free of charge, to any person obtaining a copy
11 * of this software and associated documentation files (the "Software"), to deal
12 * in the Software without restriction, including without limitation the rights
13 * to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
14 * copies of the Software, and to permit persons to whom the Software is
15 * furnished to do so, subject to the following conditions:
16 *
17 * The above copyright notice and this permission notice shall be included in all
18 * copies or substantial portions of the Software.
19 *
20 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
21 * IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
22 * FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
23 * AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
24 * LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
25 * OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE

```

**Tutorial**

---

#### Online repositories

- To clone a remote repository (e.g. from github) click New Online from the top menu. This will link a remote repository with a package on your system.
- Use the pull button to retrieve and activate the remote objects.
- If the remote repository is updated, you will see the changes and can pull to apply the updates.

#### Offline repositories

- To add a package as an offline repository, click New Offline from the top menu.
- abapGit will start tracking changes for the package without linking it to an online git repository.
- You can link the package later or just export the package content as a ZIP file.

#### Repository list and favorites

- To favorite a repository, use the icon in the repository list.
- To go to a repository, click on the repository name.
- To go back to your favorites, use the Repository List
- Explore to find projects using abapGit

Git Repository URL \*

Package \*

 ...

Branch

 ...

Folder Logic

Prefix Full Mixed

Display Name

Labels (comma-separated, allowed chars: "-\_. a-zA-Z0-9")

 ...

Ignore Subpackages

Serialize Main Language Only

? Create Online Repo Create Package Back

☰ Angabe der Anmeldedaten

Bitte geben Sie Ihren Benutzernamen und Ihr Kennwort an

Ressource:

Benutzername:

Kennwort:

OK Abbruch

The screenshot shows the abapGit interface for a GitHub repository named SAP-RAP-RAP4CINEMA. The repository URL is github.com/kamal-badawi/SAP-RAP-RAP4CINEMA. The main tab is 'main'. The 'Stage' tab is selected, showing a list of changes:

Type	Name	Changed by	Status
non-code and meta files	/.abapgit.xml		diff A
DEV-C	\$SS25_RAP4CINEMA /src/package.devc.xml	LEARN-195	diff A

## 9.4. Code mit abapGit auf GitHub pushen

Nachdem abapGit erfolgreich installiert und mit einem GitHub-Repository verknüpft wurde, können ABAP-Entwicklungsobjekte direkt aus dem SAP-System in das GitHub-Repository übertragen werden. Dieser Vorgang wird als Pushen bezeichnet. Damit wird der aktuelle Entwicklungsstand im Repository gesichert und für andere Entwickler zugänglich gemacht.

### Schritt-für-Schritt-Anleitung:

- Im SAP-System die Transaktion SE38 öffnen und das Programm ZABAPGIT ausführen.
- Das gewünschte Repository auswählen (z. B. \$SS25\_RAP4CINEMA).
- Mit Pull sicherstellen, dass der lokale Stand aktuell ist.
- Im abapGit-Hauptbildschirm auf Stage klicken, um Änderungen zu überprüfen.
- Alle geänderten Objekte kontrollieren und ggf. abwählen, falls sie nicht gepusht werden sollen.
- Im Feld Commit Message eine aussagekräftige Beschreibung der Änderungen eingeben (z. B. Metadata Extension für Kino wurde erstellt).
- Auf Commit klicken, um die Änderungen lokal zu speichern.
- Anschließend auf Push klicken, um die Änderungen auf das GitHub-Repository hochzuladen.
- Bei Aufforderung den GitHub-Benutzernamen und den Personal Access Token eingeben.
- Nach erfolgreichem Push sollte die Änderung im GitHub-Repository sichtbar sein.

The screenshot shows the abapGit interface for a GitHub repository named SAP-RAP-RAP4CINEMA. The repository URL is github.com/kamal-badawi/SAP-RAP-RAP4CINEMA. The main tab is 'main'. The 'Stage' tab is selected, showing a list of changes:

Type	Name	Changed by	Status
non-code and meta files	/.abapgit.xml		diff A
DEV-C	\$SS25_RAP4CINEMA /src/package.devc.xml	LEARN-195	diff A

SAP-RAP-RAP4CINEMA		github.com/kamal-badawi/SAP-RAP-RAP4CINEMA	main	SSS25_RAP4CINEMA
Add All and Commit (151)		Filter Objects		
Type	File	Changed by (me)	Transport	add ↓ reset ↓
A	./abapgit.xml		?	add
A	BDEF /src/zss25_r4c_c_besuch_a3.bdef.asbdef	LEARN-195	?	add

Stage Summary: + 151 files (See Details Below)

Comment \*

Initial Commit

Body

Add an optional description...

Committer Name \*

Learn-195 Learn-195

Committer Email \*

Kamal.badawi@gmx.de

Author Name

Kamal Badawi

Author Email

Kamal.badawi@gmx.de

New Branch Name

Optionally, enter a new branch name for this commit

[?](#) [Commit](#) [Back](#)

## 10. Fazit

Mit RAP4Cinema wurde ein vollständig modularisiertes und cloud-fähiges Kino-Management-System auf Basis des SAP RESTful Application Programming Model (RAP) erfolgreich umgesetzt. Die Arbeit begann mit einer sorgfältigen Analyse der Systemarchitektur, wobei Entity-Relationship-Modell, Klassendiagramme, Datenflussdiagramme, Use-Case- und Zustandsdiagramme die Grundlage für das Design der Anwendung bildeten. Diese Modellierung ermöglichte ein klares Verständnis der Datenstrukturen, der Beziehungen zwischen Entitäten und der zentralen Anwendungsfälle. Aufbauend darauf erfolgte die technische Umsetzung in SAP RAP. Die grundlegenden Packages, Datenbanktabellen und Dateneingaben über SAP Logon (SE16N) wurden eingerichtet, gefolgt von der Implementierung der verschiedenen View-Ebenen - Basis View, Interface View (Reusable View) und Projection View (Consumption View) - sowie der Service Definition und Service Bindings als OData-Service. Metadata Extensions und Behavior Definitions ermöglichen die Anpassung der Benutzeroberfläche und die Abbildung der Geschäftslogik, während Draft Database Tables Entwurfs- und Zwischenspeicherfunktionen unterstützten. Erweiterungen wie Pflichtfelder, QuickInfo-Annotationen, Datumsbereichsunterstützung, farbige Kennzeichnungen, Integration von Bildern und Symbolen, Drop-Downs, Kennzahlen im Kopfbereich der Object Page und UI-Gruppierungen verbesserten die Funktionalität und Benutzerfreundlichkeit der Anwendung. Auswertungen zur Verkaufsleistung der Mitarbeiter sowie Treue- und Top-Kundenanalysen rundeten die fachlichen Funktionalitäten ab. Die Integration von SAP Fiori stellte eine moderne, nutzerfreundliche Oberfläche bereit, und umfassende CRUD-Tests bestätigten die Stabilität und Datenintegrität der Anwendung. Die

Umsetzung komplexer hierarchischer Datenmodelle, die eindeutige Root-Zuordnung und die Datentypunterstützung im Service Binding zeigten praxisnahe Herausforderungen, die zu einem tieferen Verständnis der RAP-Architektur führten. Die GitHub-Integration über abapGit erleichterte Versionskontrolle und kollaborative Entwicklung, einschließlich SSL-Konfiguration, Personal Access Tokens und Code-Push auf GitHub. Insgesamt demonstriert RAP4Cinema das große Potenzial von SAP RAP für die Entwicklung moderner, skalierbarer Geschäftsanwendungen. Die Anwendung liefert nicht nur ein funktionsfähiges Kinosystem, sondern vermittelt praxisnahe Erfahrungen in Architektur, Entwicklung und Projektumsetzung. RAP4Cinema bietet eine solide Basis für zukünftige Erweiterungen, wie analytische Auswertungen, Integrationen externer Services oder die Bereitstellung als vollständige Cloud-Anwendung auf der SAP Business Technology Platform (BTP).

## GitHub

SAP RAP – Fiori & Dokumentation - <https://github.com/kamal-badawi/SAP-RAP-RAP4CINEMA-Documentation>

SAP RAP RAP4CINEMA - <https://github.com/kamal-badawi/SAP-RAP-RAP4CINEMA>

## Literaturverzeichnis

abapGit (Offizielle Installationsanleitung) <https://docs.abapgit.org/user-guide/getting-started/install.html>

abapGit Installation <https://www.youtube.com/watch?v=ZcFSfYyYoN8>

abapGit mit GitHub verknüpfen <https://www.youtube.com/watch?v=9JEU1G76KFs>

Foto für Besuchergruppe (Erwachsen) <https://cdn-icons-png.flaticon.com/128/2179/2179249.png>

Foto für Besuchergruppe (Kinder) <https://cdn-icons-png.flaticon.com/128/3886/3886660.png>

Foto für Besuchergruppe (Senior) <https://cdn-icons-png.flaticon.com/128/5798/5798292.png>

Foto für Mitarbeiter (Mitarbeiter-ID: 5001) [https://cdn.pixabay.com/photo/2012/04/25/00/01/person-41252\\_1280.png](https://cdn.pixabay.com/photo/2012/04/25/00/01/person-41252_1280.png)

Foto für Mitarbeiter (Mitarbeiter-ID: 5002) [https://cdn.pixabay.com/photo/2016/03/31/20/03/face-1295475\\_1280.png](https://cdn.pixabay.com/photo/2016/03/31/20/03/face-1295475_1280.png)

Foto für Mitarbeiter (Mitarbeiter-ID: 5003) [https://cdn.pixabay.com/photo/2016/03/31/19/19/comic-1294904\\_1280.png](https://cdn.pixabay.com/photo/2016/03/31/19/19/comic-1294904_1280.png)

Foto für Mitarbeiter (Mitarbeiter-ID: 5004) [https://cdn.pixabay.com/photo/2016/04/01/09/28/face-1299390\\_1280.png](https://cdn.pixabay.com/photo/2016/04/01/09/28/face-1299390_1280.png)

Foto für Mitarbeiter (Mitarbeiter-ID: 5005) [https://cdn.pixabay.com/photo/2016/03/31/19/19/comic-1294905\\_1280.png](https://cdn.pixabay.com/photo/2016/03/31/19/19/comic-1294905_1280.png)

Foto für Mitarbeiter (Mitarbeiter-ID: 5006) [https://cdn.pixabay.com/photo/2017/09/13/04/32/girl-2744387\\_1280.png](https://cdn.pixabay.com/photo/2017/09/13/04/32/girl-2744387_1280.png)

Foto für Mitarbeiter (Mitarbeiter-ID: 5007) [https://cdn.pixabay.com/photo/2017/09/12/19/31/girl-2743378\\_1280.png](https://cdn.pixabay.com/photo/2017/09/12/19/31/girl-2743378_1280.png)

Foto für Ticketart (Kinder) <https://cdn-icons-png.flaticon.com/128/1702/1702342.png>

Foto für Ticketart (Standard) <https://cdn-icons-png.flaticon.com/128/554/554744.png>

Foto für Ticketart (VIP) <https://cdn-icons-png.flaticon.com/128/6701/6701785.png>

Foto für Ticketgültigkeit (Gültig) <https://cdn-icons-png.flaticon.com/128/3904/3904485.png>

Foto für Ticketgültigkeit (Ungültig) <https://cdn-icons-png.flaticon.com/128/6627/6627066.png>

Foto für Zahlungsart (Bar) <https://cdn-icons-png.flaticon.com/128/2751/2751615.png>

Foto für Zahlungsart (Karte) <https://cdn-icons-png.flaticon.com/128/311/311147.png>

Foto für Zahlungsstatus (Bezahlt) <https://cdn-icons-png.flaticon.com/128/4315/4315445.png>

Foto für Zahlungsstatus (Offen) <https://cdn-icons-png.flaticon.com/128/694/694572.png>

- Foto für Zahlungsstatus (Storniert) <https://cdn-icons-png.flaticon.com/128/7699/7699001.png>
- Foto für Zeitkategorie (Heute) <https://cdn-icons-png.flaticon.com/128/11678/11678959.png>
- Foto für Zeitkategorie (Vergangenheit) <https://cdn-icons-png.flaticon.com/128/10755/10755206.png>
- Foto für Zeitkategorie (Zukunft) <https://cdn-icons-png.flaticon.com/128/10755/10755084.png>
- Sehr hilfreiche Playlist zur RAP-Entwicklung (a) [https://www.youtube.com/watch?v=yE7LIWY-x-c&list=PLxcJ56QBR\\_Mr8E5zyvJGPhwa\\_cdTq060t](https://www.youtube.com/watch?v=yE7LIWY-x-c&list=PLxcJ56QBR_Mr8E5zyvJGPhwa_cdTq060t)
- Sehr hilfreiche Playlist zur RAP-Entwicklung (b) [https://www.youtube.com/watch?v=iXYID\\_xtQ68&list=PLqz8SLrkjv2iwPtPRJ4V6zJRfgNrgZKIQ](https://www.youtube.com/watch?v=iXYID_xtQ68&list=PLqz8SLrkjv2iwPtPRJ4V6zJRfgNrgZKIQ)
- Sehr hilfreiche Playlist zur RAP-Entwicklung (c) [https://www.youtube.com/watch?v=rHTEP5wQJSE&list=PLI2R5hV2X4PwOLMc7-98qeVVTDzor\\_Cj-](https://www.youtube.com/watch?v=rHTEP5wQJSE&list=PLI2R5hV2X4PwOLMc7-98qeVVTDzor_Cj-)