

UI5 Spreadsheet Importer

Marian Zeis - Unabhängiger SAP Entwickler

Ausgangslage

- Herausforderung: Notwendigkeit einer Excel-Upload-Funktion in einer UI5-Anwendung bei einem Kunden
- Sollte in andere UI5 Anwendungen integriert werden
- Von Anfang Planung als Open Source Projekt

Ziele bei der Entwicklung

- Einfache Integration in UI5-Anwendungen (wie "out-of-the-box"-Lösungen).
- Kompatibilität mit verschiedenen UI5/Fiori-Szenarien, SAP-Plattformen und OData-Versionen.
- Anpassungsfähigkeit und umfangreiche Customizing-Optionen.
- Gute Dokumentation und Tests.

Motivation: Effizienzsteigerung für Entwickler durch Wiederverwendbarkeit und einfache Integration.

UI5 Spreadsheet Importer

Grundkonzept ist eine UI5 Component

Apache-2.0 license

Landing Page: <https://spreadsheet-importer.com/>

Live Demo: <https://livedemo.spreadsheet-importer.com/>

Dokumentation: <https://docs.spreadsheet-importer.com/>

GitHub: <https://github.com/spreadsheetimporter/ui5-cc-spreadsheetimporter>

npm: <https://www.npmjs.com/package/ui5-cc-spreadsheetimporter>

Wie funktioniert es?



Orders V2 FE 120 ▾

DU

Standard ▾



Suchen



Bearbeitungsstatus:

Alle ▾

Angelegt von:



Start

Filter anpassen



Orders (6)

Spreadsheet Upload

Anlegen

Löschen



OrderNo	Customer	Währung	Date	
<input type="radio"/> 2	jane.doe@test.com		30.01.2019, 01:00:00	>
<input type="radio"/> 200	jane.doe2@test.com		29.01.2019, 01:00:00	>
<input type="radio"/> 201	jane.doe4@test.com		28.01.2019, 01:00:00	>
<input type="radio"/> 202	jane.doe4@test.com		28.01.2019, 01:00:00	>
<input type="radio"/> 203	jane.doe4@test.com		28.01.2019, 01:00:00	>
<input type="radio"/> 1	john.doe@test.com		31.01.2019, 01:00:00	>

Wie funktioniert es?



Orders V2 FE 120 ▾

DU

Standard ▾



Suchen



Bearbeitungsstatus:

Alle ▾

Angelegt von:



Start

Filter anpassen

Orders (6)

	OrderNo	Customer	Währung	Date	
<input type="radio"/>	2	jane.doe@test.com		30.01.2019, 01:00:00	>
<input type="radio"/>	200	jane.doe2@test.com		29.01.2019, 01:00:00	>
<input type="radio"/>	201	jane.doe4@test.com		28.01.2019, 01:00:00	>
<input type="radio"/>	202	jane.doe4@test.com		28.01.2019, 01:00:00	>
<input type="radio"/>	203	jane.doe4@test.com		28.01.2019, 01:00:00	>
<input type="radio"/>	1	john.doe@test.com		31.01.2019, 01:00:00	>

Spreadsheet Upload

Anlegen

Löschen



Wie funktioniert es?

```
</AssociationSet>
</EntityContainer>
<EntityType Name="Orders">
  <Key>
    <PropertyRef Name="ID"/>
    <PropertyRef Name="IsActiveEntity"/>
  </Key>
  <Property Name="ID" Type="Edm.Guid" Nullable="false"/>
  <Property Name="createdAt" Type="Edm.DateTimeOffset" Precision="7"/>
  <Property Name="createdBy" Type="Edm.String" MaxLength="255"/>
  <Property Name="modifiedAt" Type="Edm.DateTimeOffset" Precision="7"/>
  <Property Name="modifiedBy" Type="Edm.String" MaxLength="255"/>
  <Property Name="OrderNo" Type="Edm.String" MaxLength="22"/>
  <NavigationProperty Name="Items" Relationship="OrdersService.OrderItems_order" FromRole="Orders" ToRole="OrderItems"/>
  <NavigationProperty Name="Shipping" Relationship="OrdersService.ShippingDetails_order" FromRole="Orders" ToRole="ShippingDetails"/>
  <Property Name="buyer" Type="Edm.String" MaxLength="255"/>
  <NavigationProperty Name="currency" Relationship="OrdersService.Orders_currency" FromRole="Orders" ToRole="Currencies"/>
  <Property Name="currency_code" Type="Edm.String" MaxLength="3"/>
  <Property Name="spreadsheetRow" Type="Edm.Int32"/>
  <Property Name="IsActiveEntity" Type="Edm.Boolean" Nullable="false"/>
  <Property Name="HasActiveEntity" Type="Edm.Boolean" Nullable="false"/>
  <Property Name="HasDraftEntity" Type="Edm.Boolean" Nullable="false"/>
  <NavigationProperty Name="DraftAdministrativeData" Relationship="OrdersService.Orders_DraftAdministrativeData" FromRole="Orders" ToRole="DraftAdministrativeData"/>
  <Property Name="DraftAdministrativeData_DraftUUID" Type="Edm.Guid"/>
  <NavigationProperty Name="SiblingEntity" Relationship="OrdersService.Orders_SiblingEntity" FromRole="Orders" ToRole="Orders1"/>
</EntityType>
<EntityType Name="OrderItems">
  <Key>
    <PropertyRef Name="ID"/>
    <PropertyRef Name="IsActiveEntity"/>
  </Key>
  <Property Name="ID" Type="Edm.Guid" Nullable="false"/>
  <NavigationProperty Name="order" Relationship="OrdersService.OrderItems_order" FromRole="OrderItems" ToRole="Orders"/>
</EntityType>
```

	A	B	C	D	E
1	Order Number[OrderNo]	User ID[buyer]			
2		3 test@test.de			
3		4 test@test.de			
4					

Spreadsheet Upload

[Vorschau anzeigen](#)[Vorlage herunterladen](#)

0 Datensätze stehen zu hochladen bereit

31.01.2019, 01:00:00

upload

	A	B
1	Order Number[OrderNo]	User ID[buyer]
2		3 test@test.de
3		4 test@test.de
4		

extract (SheetJS)

Spreadsheet Upload

[Vorschau anzeigen](#)[Vorlage herunterladen](#)

2 Datensätze stehen zu hochladen bereit

validate

```
[
  {
    "OrderNo": "3",
    "buyer": "test@test.de"
  },
  {
    "OrderNo": "4",
    "buyer": "test@test.de"
  }
]
```

parse

```
{
  {
    "Order Number[OrderNo]": {
      "rawValue": 3,
      "sheetDataType": "n",
      "format": "General",
      "formattedValue": "3"
    },
    "User ID[buyer]": {
      "r": "<t>test@test.de</t>",
      "h": "test@test.de",
      "l": {
        "rawValue": "test@test.de",
        "sheetDataType": "s",
        "format": "General",
        "formattedValue": "test@test.de"
      }
    }
  },
  {
    "Order Number[OrderNo]": {
      "rawValue": 4,
      "sheetDataType": "n",
      "format": "General",
      "formattedValue": "4"
    },
    "User ID[buyer]": {
      "r": "<t>test@test.de</t>",
      "h": "test@test.de",
      "rawValue": "test@test.de",
      "sheetDataType": "s",
      "format": "General",
      "formattedValue": "test@test.de"
    }
  }
}
```

post

POST Orders HTTP/1.1

Content-Type: application/json

Content-ID: id-1725871991602-250

sap-contextid-accept: header

Accept: application/json

Accept-Language: de-DE

DataServiceVersion: 2.0

MaxDataServiceVersion: 2.0

X-Requested-With: XMLHttpRequest

Content-Length: 83

{"OrderNo": "3", "buyer": "test@test.de", "__metadata": {"type": "OrdersService.Orders"}}

--changeset_9add-7c10-0cd9

refresh

Orders (9)

OrderNo	Customer
<input type="radio"/> test string	test string
<input type="radio"/> 4	test@test.de
<input type="radio"/> 3	test@test.de
<input type="radio"/> 203	jane.doe4@test.com
<input type="radio"/> 202	jane.doe4@test.com

Ziele umgesetzt?

Eigentlich alle Ziele umgesetzt:

- Integration überall möglich
 - Sogar mehr als ursprünglich gedacht
- Deployment überall möglich
 - Hier schwieriger als gedacht
 - Konzept musste überarbeitet werden
 - Änderung und Bugs auf SAP Seite hat es schwieriger gemacht
- Umfassende Dokumentation und Beispiele
- E2E Tests umgesetzt
 - Wegen Komplexität weniger als gewünscht
- Zwischen 200-300 wöchentliche Downloads auf npmjs

Security

- Dateien werden nicht ins Backend geschickt
- Datei kann vor dem prozessieren gescannt werden (z.B. mit [SAP Malware Scanning Service](#))
- Alle Teile sind Open Source einsehbar
- Einzige Dependency ist “[SheetJS](#)” wird immer aktuell gehalten
- GitHub Dependabot alerts werden immer schnellstmöglich behoben
- [Provenance](#): Verfolgbarkeit der Entstehung und Änderungen von Code/Abhängigkeiten
- GitHub Settings
 - Kein Write Access
 - Bei Pull Requests werden die Actions nicht automatisch ausgeführt
 - Review vor Merge

Provenance

Built and signed on	Source Commit	github.com/spreadsheetimporter/ui5-cc-spreadsheetimp...
 GitHub Actions	Build File	.github/workflows/release-please.yml
View build summary	Public Ledger	Transparency log entry

Warum Open Source?

- Schnelle Fehlererkennung und Verbesserung durch die breite Nutzung
 - Über 50 Issues, die zu Verbesserungen geführt
- Zusammenarbeit und Feedback aus der Entwickler-Community.
 - Inklusive Hilfe von SAP Mitarbeiter
- Flexibilität, neue Ideen auszuprobieren und von Community-Beiträgen zu profitieren.
- Persönliche Weiterentwicklung durch das Open-Source-Projektmanagement.

Open Source ermöglichte eine schnellere und umfangreichere Verbreitung und Verbesserung der Komponente.

Erfahrungen und Lehren aus der Open-Source-Arbeit

- Umfangreiche Benutzung auf allen SAP-Plattformen und UI5-Szenarien durch externe Entwickler.
- Dies erfordert aber:
 - Feedback muss eingefordert werden
 - Niedrige Einstieg (einfache Installation, gute Dokumentation, einfach zu debuggen)
 - Dadurch aber auch viele Nutzung ohne Feedback
- Support ist notwendig und kostet Zeit
- Issue Erstellung mit Template:
 - <https://github.com/spreadsheetimporter/ui5-cc-spreadsheetimporter/issues/new/choose>

Erfahrungen und Lehren aus der Open-Source-Arbeit

- Marketing ist notwendig:
 - Landing Page statt nur README
 - Blog Artikel
 - Social Media (Twitter, LinkedIn)
 - <https://bestofui5.org/>
 - <https://bestofcapjs.org/>
 - <https://dotabap.org/>
- Prototyping neuer Funktionen: Nutzung des Projekts als "Spielplatz" für innovative Ideen in Zusammenarbeit mit SAP
 - Hilft wiederum Open Source
 - Vorträge z.B. UI5con 2024: [Seamless Integration: Crafting the Perfect UI5 Development Environment](#)

Erkenntnisse

- Gutes Open Source Projekt benötigt Zeit
- Erfahrung kann man mit der Zeit sammeln
- Mitarbeiter finden ist schwierig
- **Gute Community ist wichtig!**

Danke

Kontakt:

Marian Zeis

Email: marian@marianzeis.de

LinkedIn: <https://www.linkedin.com/in/marianzeis/>

GitHub: <https://github.com/marianfoo>

Präsentation:

<https://github.com/marianfoo/marianfoo/raw/main/DSAGOpenSourceUI5SpreadsheetImporter.pdf>

