

## Designpaper 8.1

---

### Übersicht

**Projekt:** Projekt Episko

*Inkrement:* X

*Arbeitspaket:* Y

**Autor:** Ben Oeckl

**Datum:** 02.03.2025

**Zuletzt geändert:**

*von:*

*am:*

**Version:** 1

**Prüfer:**

**Letzte Freigabe:**

*durch:*

*am:*

### Changelog

Datum	Verfasser	Kurzbeschreibung
02.03.2025	Ben Oeckl	Initiales Erstellen und Verfassen

### Distribution List

- Simon Blum [simon21.blum@gmail.com](mailto:simon21.blum@gmail.com)
  - Ben Oeckl [ben@oeckl.com](mailto:ben@oeckl.com)
  - Maximilian Rodler [maximilianreinerrodler@gmail.com](mailto:maximilianreinerrodler@gmail.com)
  - Paul Stöckle [paul.stoeckle@t-online.de](mailto:paul.stoeckle@t-online.de)
- 

## Designpaper

### Ziel

Das Arbeitspaket hat das Ziel, eine moderne und modulare Frontend-Lösung zur Erstellung, Bearbeitung und Löschung von Manifestdateien mit Metadaten zu realisieren. Durch die Implementierung der Create- und Edit-Seiten soll der Nutzer die Möglichkeit haben, Metadaten zu verwalten und den Pfad zu einer Manifestdatei anzugeben. Die Lösung wird unter Einsatz von TypeScript zur Definition der Metadatenmodelle sowie moderner UI-Komponenten (aus der shadcn-svelte Bibliothek) entwickelt.

## Umsetzung

- **Gemeinsame Komponente:**

Eine zentrale Komponente wird erstellt, die sämtliche Eingabefelder für die Manifestdatei umfasst. Diese Komponente arbeitet in zwei Modi, die mittels Tabs gesteuert werden:

  - **Manual Entry:** Der Nutzer gibt alle erforderlichen Daten direkt ein.
  - **From File:** Der Nutzer kann einen Dateipfad entweder manuell eingeben oder über einen Dateiauswahldialog setzen.
- **Seiten für Create und Edit:**
  - **Create Project:** Übergibt leere Standardwerte an die gemeinsame Komponente.
  - **Edit Project:** Lädt existierende Projektdaten und übergibt diese als Initialwerte an die gemeinsame Komponente.
- **TypeScript-Datenmodell:**

Die Metadaten werden über ein TypeScript-Interface definiert.
- **Die Logik wird in einen der nächsten Arbeitspaketen umgesetzt**