

## Designpaper 6.3

---

### Übersicht

**Projekt:** Projekt Episko  
*Inkrement:* 6  
*Arbeitspaket:* 3  
**Autor:** Paul Stöckle  
**Datum:** 20.01.2025  
**Zuletzt geändert:** 20.01.2025  
*von:* Paul Stöckle  
*am:* 25.01.2025  
**Version:** 1  
**Prüfer:** Max Rodler  
**Letzte Freigabe:**  
*durch:* Max Rodler  
*am:* 27.01.2025

### Changelog

Datum	Verfasser	Kurzbeschreibung
20.01.2025	Paul Stöckle	Initiales Erstellen und Verfassen
25.01.2025	Paul Stöckle	Designänderungen

### 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

Im Rahmen des Arbeitspaketes soll die grundlegende Struktur der CLI-Komponente entwickelt werden. `### CLI` Es wird Clap zur Argumentübergabe verwendet. `### Commands`, Argumente und Funktionalitäten | Command | Funktion | |-----|-----|  
create [OPTIONS] | Erstellen einer neuen Manifestdatei. Daten können durch Flags mitgegeben werden. Alle anderen werden durch interaktiv eingegeben Daten ergänzt. | | create [OPTIONS mit -n] | Erstellen einer neuen Manifestdatei. Daten werden nur den Flags entnommen. Bei relevanten Daten, die

fehlen wird das Programm abgebrochen. | | remove <FILE> | Löschen der angegebenen Datei und Entfernung der Informationen aus dem System. | | add <FILE> | Hinzufügen der angegebenen Datei zum System. | | validate <FILE> | Validierung auf Gültigkeit der angegebenen Datei. |

### **Code-Isolation**

Die CLI wird in einem eigenen Paket entwickelt: “episko\_cli”. Die verschiedenen Komponenten werden auf mehrere Dateien aufgeteilt. Zur Argumentübergabe wird clap genutzt. Zur interaktiven Eingabe wird dialoguer benutzt.