

Projektpräsentation

Karmaknit System Release Editor

Webbasierter Editor für Systemfreigaben

Max Schmitt, Projektleiter · Karmaknit GmbH, Frankfurt am Main

Ausgangssituation: Freigaben in Textdokumenten

Bisher

- Freigabedokumente in Textverarbeitung
- Freitext, nicht maschinell auswertbar
- Verbindung System ↔ Komponenten nur im Dokument



Künftig

- Strukturierte, digitale Freigabedaten
- Erfassung und Anzeige über Web-Oberfläche
- Komponenten sind verknüpft und wiederverwendbar



Nutzen: Elektronische Datenverarbeitung wird möglich, z. B. die Sicherheitsüberprüfung der enthaltenen Komponenten, wie sie der EU Cyber Resilience Act vorsieht.

Projektziele und Abgrenzung

Im Projekt



Datenmodell für System- und Komponentenfreigaben



Speicherung als JSON-Dateien



CRUD-Oberfläche für beide Freigabearten



Dokumentartige Ansicht mit Komponentenlinks

Nicht Teil des Projekts



Benutzer- und Rechteverwaltung



Datenbankanbindung



Automatisierte Unit-Tests



Automatische Sicherheitsprüfung

Vorgehen: fünf Phasen, 80 Arbeitsstunden

1

Konzeption

MVVM, drei Projekte

2

Datenmodell

Klassen, JSON-Aufbau

3

Implementierung

ViewModels, Views,
CRUD

4

Test

Beispieldaten im
Browser

5

Dokumentation

Markdown und Word

Ressourcen: Projektleiter Max Schmitt, Entwicklungsrechner, .NET SDK. Ausschließlich quelloffene Bibliotheken (MIT), daher keine Lizenzkosten.

Architektur: MVVM in drei Projekten



DataModels

`Karmaknit.SystemReleaseEditor.
DataModels`

Datenklassen, JSON-Persistenz,
Beispieldaten

ViewModels

`Karmaknit.SystemReleaseEditor.
ViewModels`

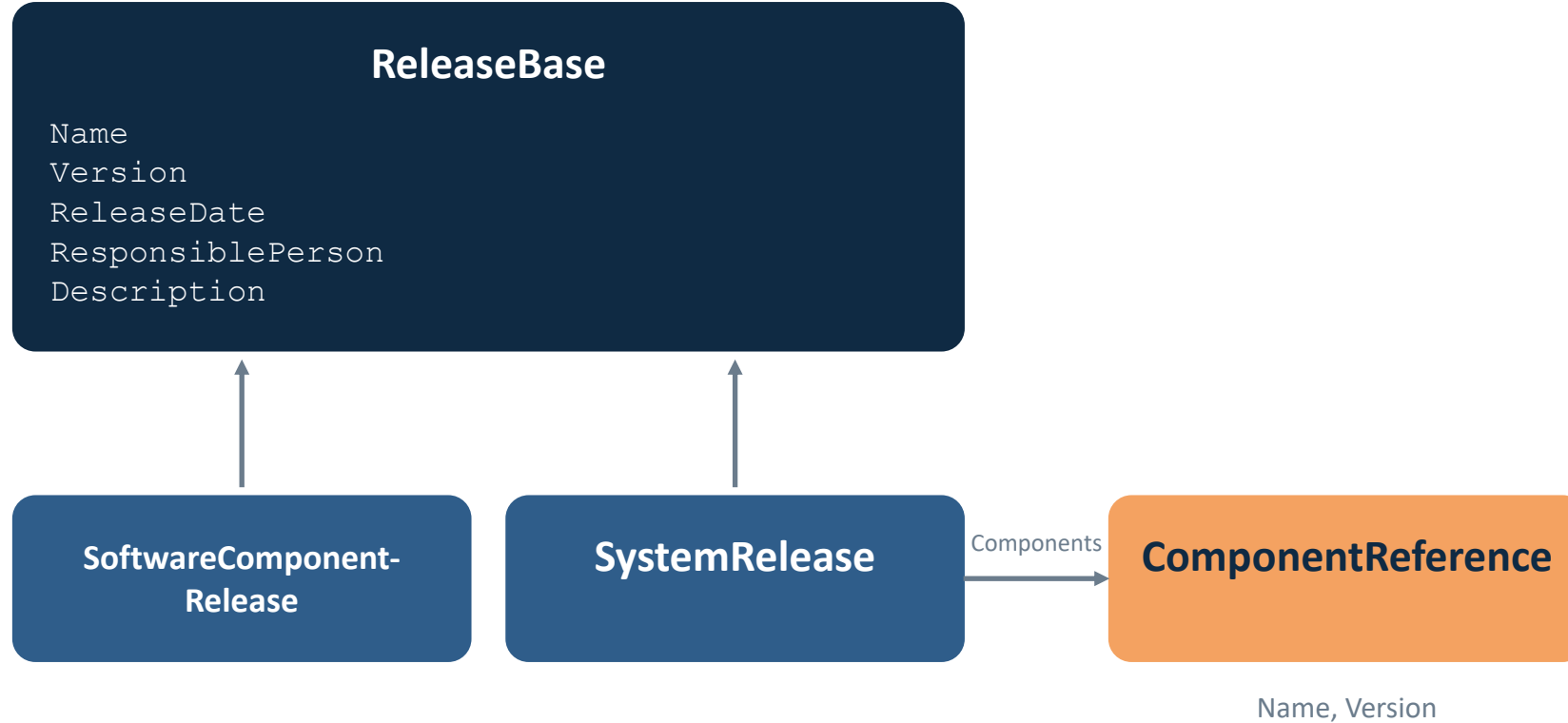
Darstellungslogik, Validierung, CRUD-
Operationen

UI

`Karmaknit.SystemReleaseEditor.
UI`

Blazor-Server-Anwendung,
Bootstrap, Routing

Datenmodell: Referenzen statt Kopien



Ein Prinzip

- Systemfreigabe enthält nur Verweise (Name + Version)
- Eine Komponente kann in mehreren Systemen stehen
- Korrektur wirkt überall, keine doppelte Pflege

Persistenz: eine lesbare JSON-Datei je Version

```
C:\KarmaknitSystemReleases\  
System_Shop-Suite_1.0.0.json  
System_Shop-Suite_1.1.0.json  
Component_Auth-Service_1.3.0.json  
Component_Billing-Engine_2.0.1.json
```

```
{  
  "Components": [  
    { "Name": "Auth Service",  
      "Version": "1.3.0" }  
  ],  
  "Name": "Shop Suite",  
  "Version": "1.1.0",  
  "ReleaseDate": "2025-06-20T00:00:00",  
  "ResponsiblePerson": "Max Schmitt"  
}
```

- Name + Version bilden den Schlüssel und den Dateinamen
- Umbenennen: Datei wird verschoben, Verweise werden angepasst
- Löschen einer Komponente entfernt Verweise in Systemen
- Zugriffe synchronisiert, Beispieldaten beim ersten Start

ViewModels mit CommunityToolkit.Mvvm

System- und ComponentReleaseListViewModel

Listen laden (Read), löschen (Delete)

SystemReleaseEditViewModel

Create, Update, Komponentenauswahl

ComponentReleaseEditViewModel

Create, Update, Verwender anzeigen

SystemReleaseDocumentViewModel

aufgelöste, schreibgeschützte Ansicht

```
public partial class
    ComponentReleaseEditViewModel
    : ObservableObject
{
    [ObservableProperty]
    private string _name;

    public bool Save()
    // validates and stores
}
```

Der Source Generator erzeugt die Properties samt Änderungsbenachrichtigung. Die Edit-ViewModels prüfen Pflichtfelder und Eindeutigkeit von Name + Version.

Oberfläche: CRUD für beide Freigabearten

Karmaknit System Release Editor

System releases

Component releases

System releases

New system release

Name	Version	Release date	Responsible	Components	
Shop Suite	1.0.0	2025-04-30	Max Schmitt	3	ViewEditDelete
Shop Suite	1.1.0	2025-06-20	Max Schmitt	4	ViewEditDelete

Liste der Systemfreigaben

- Razor-Komponenten mit Bootstrap
- Views binden direkt an ViewModel-Properties
- Löschen mit Rückfrage, Hinweis auf Verwendung

/systems

Liste, Ansehen, Bearbeiten, Löschen

/systems/edit

Anlegen und Ändern mit Komponentenauswahl

/components

Komponentenfreigaben verwalten

/systems/document

Dokumentartige Ansicht

Dokumentansicht und Bearbeitung

Karmaknit System Release Editor

System releases

Component releases

System release: Shop Suite

Version

1.1.0

Release date

2025-06-20

Responsible person

Max Schmitt

Description

Adds reporting and updated authentication.

Software components

1. [Auth Service 1.3.0](#) - released 2025-06-02 by Anna Berger

2. [Billing Engine 2.0.1](#) - released 2025-03-10 by Jonas Weber

3. [Customer Portal 4.1.0](#) - released 2025-04-22 by Lea Fischer

4. [Reporting Module 0.9.5](#) - released 2025-05-05 by Jonas Weber

Edit

Back to list

Dokumentansicht: Komponenten als Links

Karmaknit System Release Editor

System releases

Component releases

Edit system release

Name

Shop Suite

Version

1.1.0

Release date

20.06.2025

Responsible person

Max Schmitt

Description

Adds reporting and updated authentication.

Software components

☐ Auth Service 1.2.0

☒ Auth Service 1.3.0

☒ Billing Engine 2.0.1

☒ Customer Portal 4.1.0

☒ Reporting Module 0.9.5

Save

Cancel

Bearbeiten: Komponenten per Checkbox wählen

Wesentliche Designentscheidungen



MVVM

Trennung von Darstellung, Logik und Daten



CommunityToolkit.Mvvm

Weniger Code für Änderungsbenachrichtigung



Referenzen statt Kopien

Keine Redundanz, keine Inkonsistenz



Name + Version als Schlüssel

Lesbare Dateinamen, keine technische ID



Repository in DataModels

Nur drei Projekte vorgegeben



Synchronisierter Zugriff

Mehrere Sitzungen schreiben parallel

Qualitätssicherung und Soll-Ist-Vergleich

Anforderungen erfüllt

- Blazor Server 8, C#, Bootstrap
- Drei Projekte, Namespaces, src/ mit .sln
- JSON.NET, eine Datei je Version
- Komponenten ohne Kopie wiederverwendbar
- CRUD für beide Freigabearten
- Dokumentansicht mit Komponentenlinks

Test mit Beispieldaten

Keine Unit-Tests (Vorgabe), manueller Test im Browser:



Listen und Dokumentansicht inkl. Links



Bearbeiten mit Komponentenauswahl



Anlegen: korrekte JSON-Datei erzeugt



Löschen und Umbenennen noch nicht separat getestet

Fazit und Ausblick

Fazit

Freigabedaten sind digital erfasst, verknüpft und als Dokument darstellbar. Referenzen und lesbare Dateinamen haben sich als einfache, nachvollziehbare Lösung bewährt.

Ausblick

- Automatisierte Tests
- Export, z. B. SBOM oder PDF
- Benutzer- und Rechteverwaltung
- Suche bei der Komponentenauswahl

Vielen Dank! Fragen?