

Projektpräsentation



IHK-Abschlussprüfung

Fachinformatiker Anwendungsentwicklung

Kacper Sobczak

Prüflingsnummer 104-05258

Entwicklung eines Contentful Plugins zur Überprüfung der Validität von Einträgen von Contentfeldern

Gliederung



- Unternehmen
- Analyse
- Planung
- Implementierung
- Test
- Fazit & Ausblick



Unternehmen



Gastronomie



Hotellerie



Textil



Online-Marktplatz

LUSINI



Hauptsitz im Wertigen

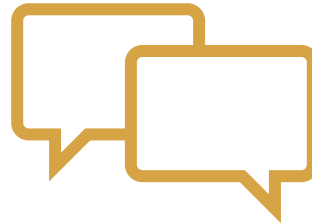


- Gründung 1987
- Webentwicklung
- 15 Entwickler
- E-Commerce Shop
- Ominchannel
- International





15 Niederlassungen





Analyse

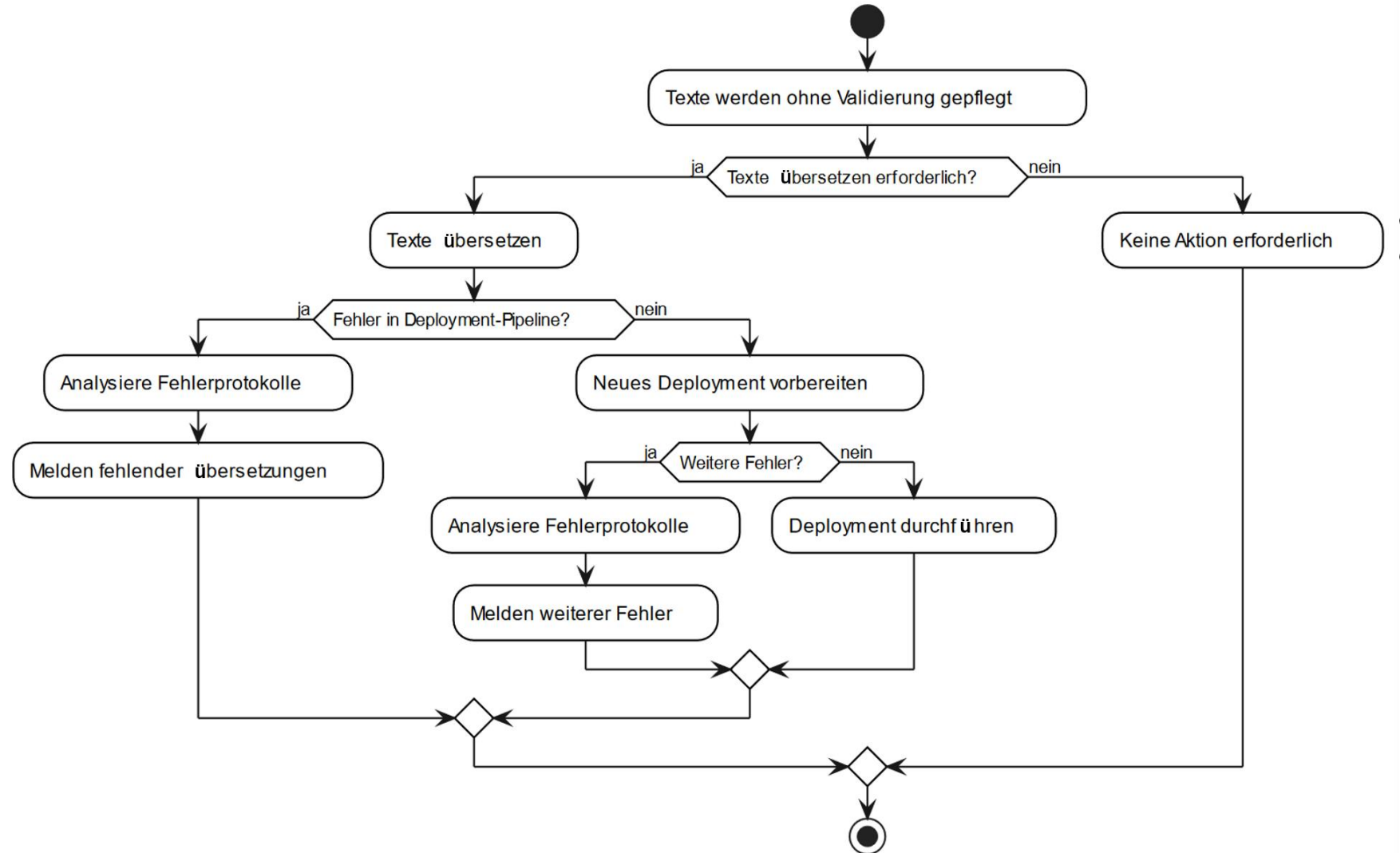
Projektkosten



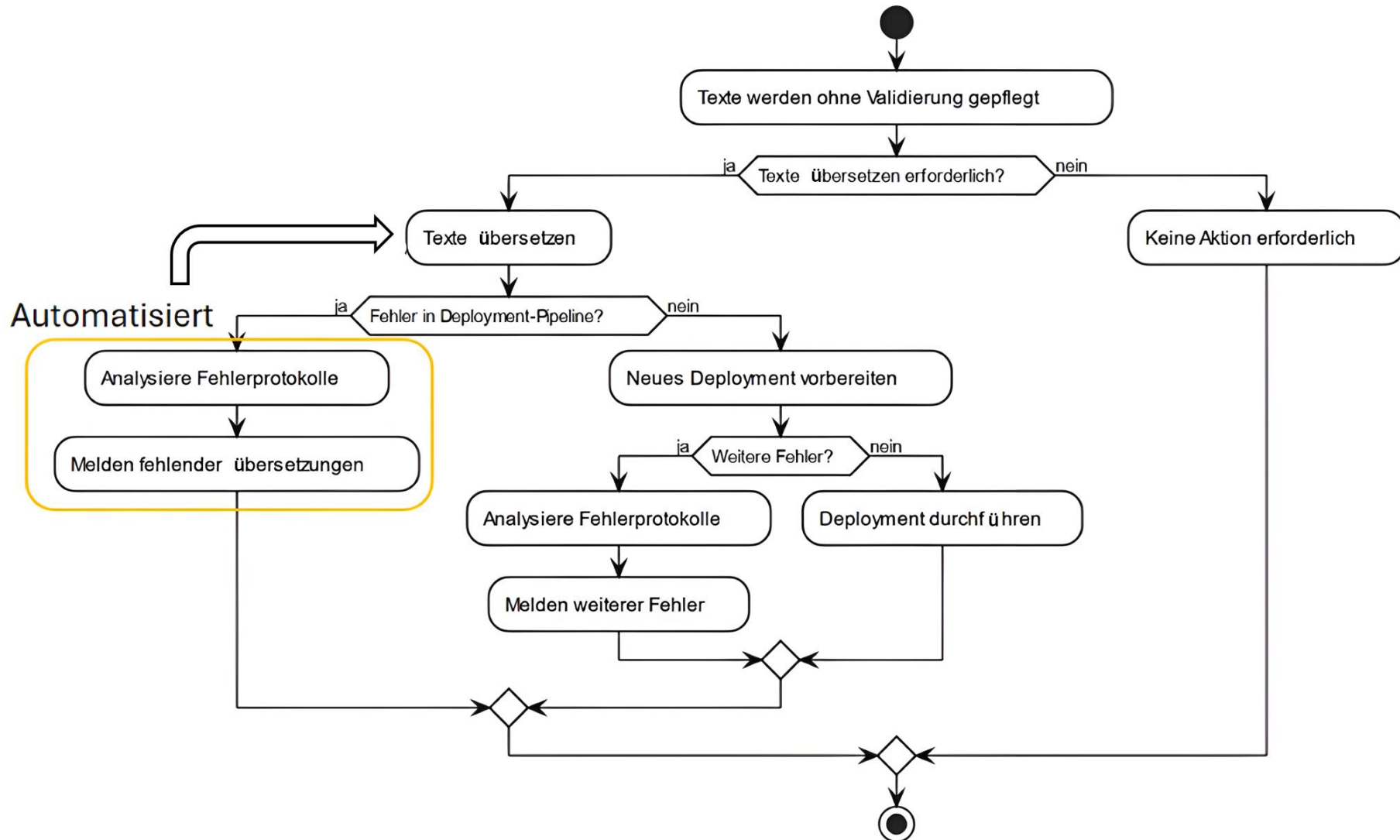
Vorgang	Mitarbeiter	Zeit	Stundensatz	Gesamt
Entwicklungskosten	1 x Auszubildener	80 h	10 €	800,00 €
Fachgespräch	2 x Mitarbeiter	5 h	30 €	150,00 €
Code-Review	1 x Mitarbeiter	1 h	30 €	30,00 €
Abnahme	1 x Mitarbeiter	0,5 h	30 €	15,00 €

Σ 995,00 €

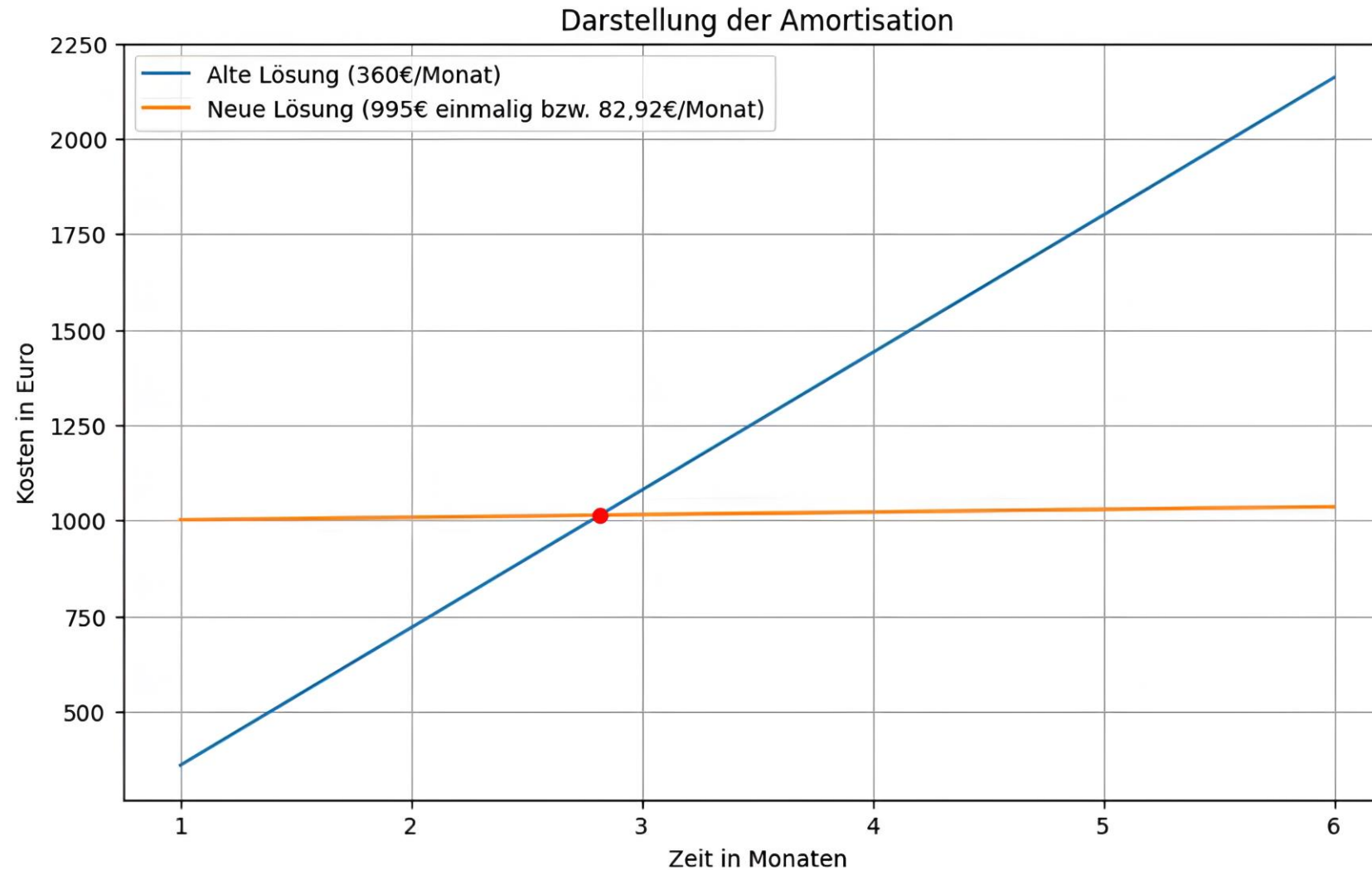
Ist-Zustand



Soll-Zustand



Armortisation

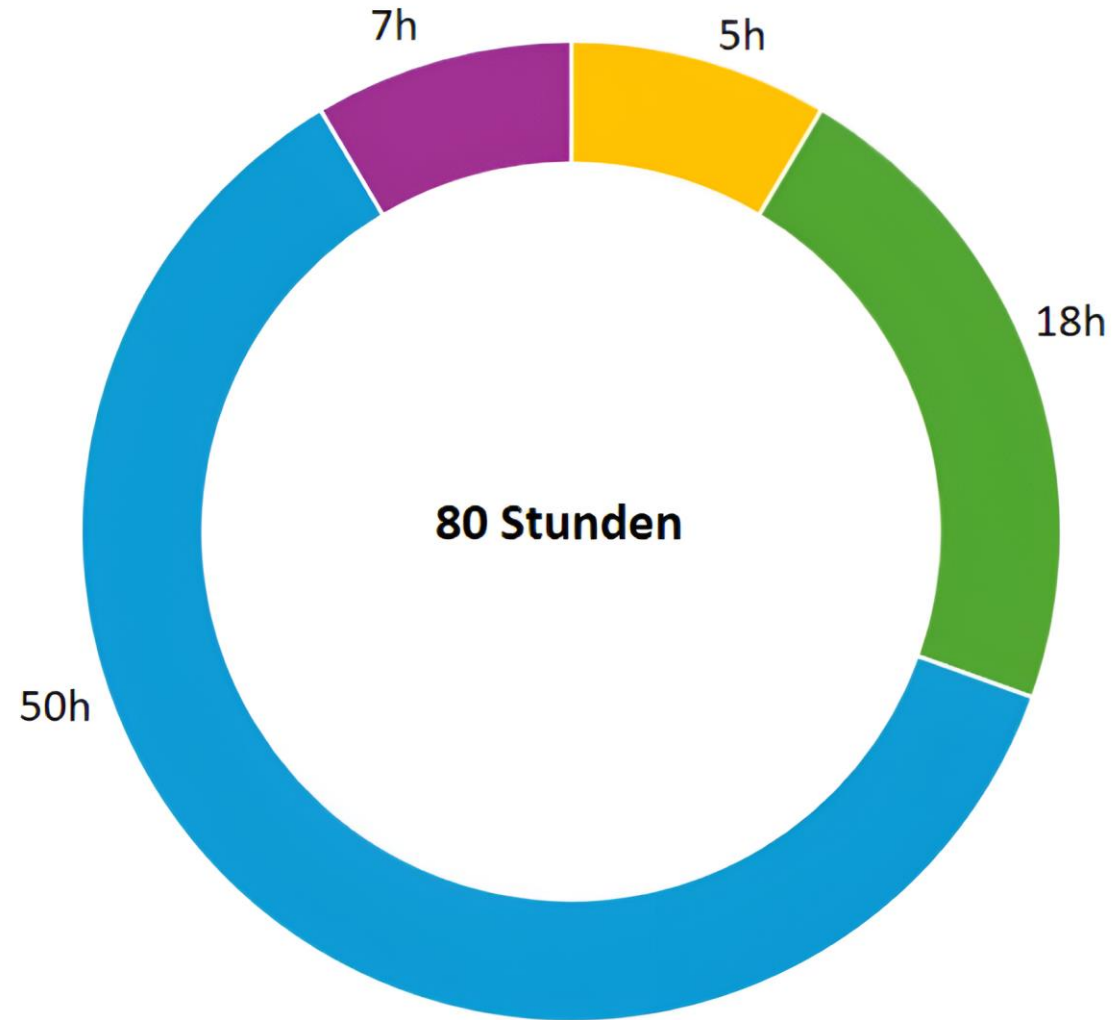


Planung

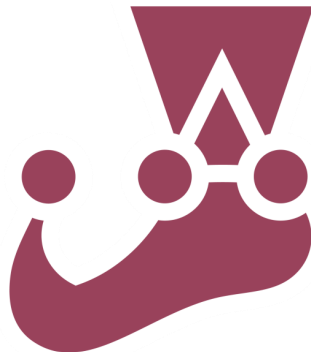
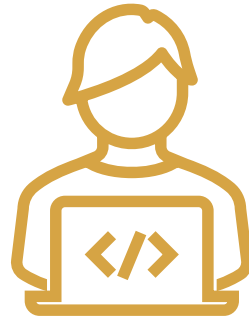
Zeitplanung

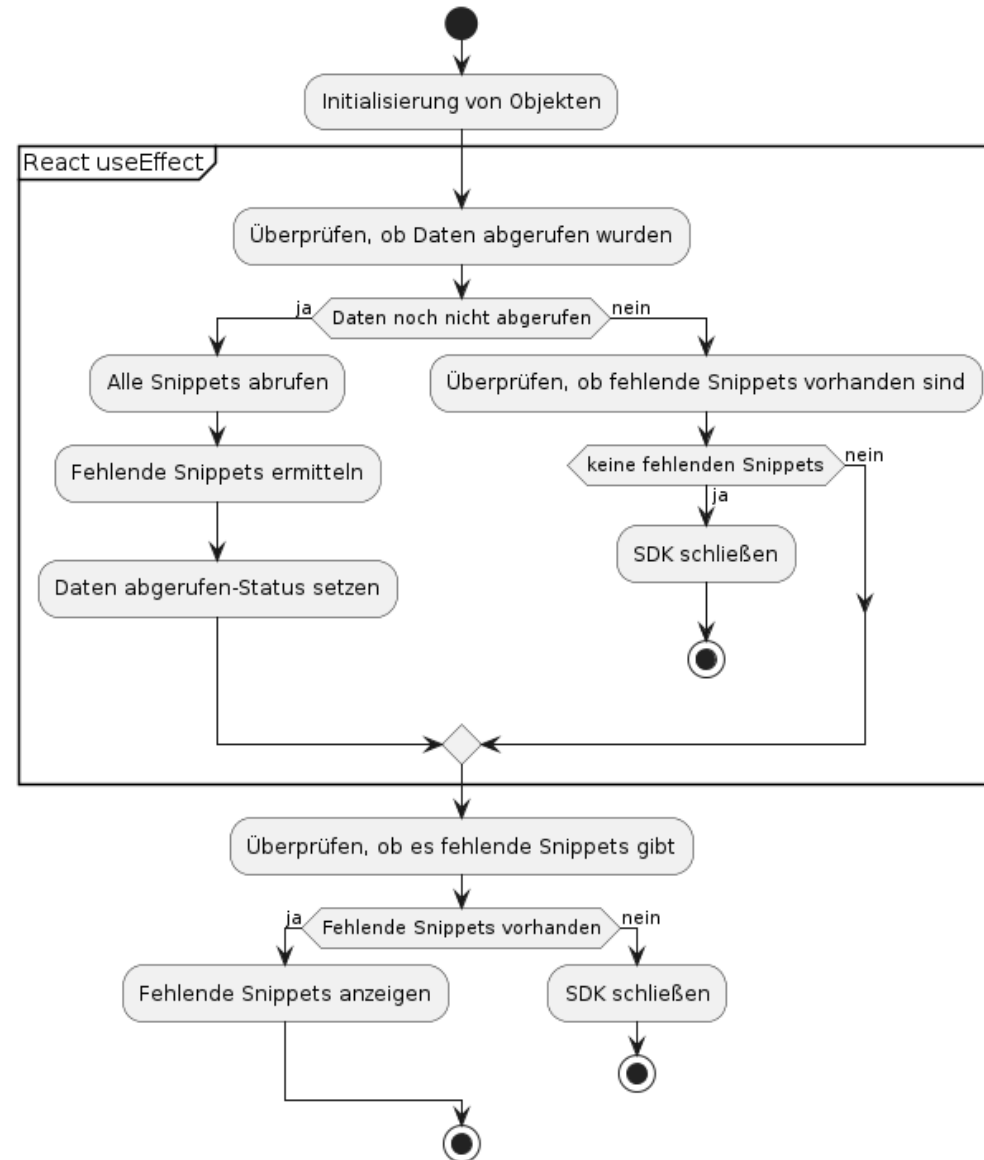


- Informationbeschaffung
- Planung
- Durchführung
- Kontrolle



Entwurf





Implementierung

JSON-Preview

```
{
  "name": "Snippets",
  "description": "",
  "displayField": "fieldName",
  "fields": [
```

```
{
  "id": "fieldName",
  "name": "Field Name",
  "type": "Symbol",
  "localized": false,
  "required": true,
  "validations": [],
  "disabled": false,
  "omitted": false
},
{
  "id": "value",
  "name": "value",
  "type": "Text",
  "localized": true,
  "required": true,
  "validations": [],
  "disabled": false,
  "omitted": false
},
{
  "id": "context",
  "name": "Context",
  "type": "Text",
  "localized": true,
  "required": false,
  "validations": [],
  "disabled": false,
  "omitted": false
},
{
  "id": "description",
  "name": "Description",
  "type": "Text",
  "localized": false,
  "required": false,
  "validations": [],
  "disabled": false,
  "omitted": false
},
```

```
{
  "id": "deprecated",
  "name": "deprecated",
  "type": "Boolean",
  "localized": false,
  "required": false,
  "validations": [],
  "disabled": false,
  "omitted": false
},
{
  "sys": {
    "space": {
      "sys": {
        "type": "Link",
        "linkType": "Space",
        "id": "aza65graowyr"
      }
    }
  },
  "id": "snippets",
  "type": "ContentType",
  "createdAt": "2021-04-12T10:08:01.221Z",
  "updatedAt": "2023-02-08T15:32:23.350Z",
  "environment": {
    "sys": {
      "id": "master",
      "type": "Link",
      "linkType": "Environment"
    }
  },
  "publishedVersion": 41,
  "publishedAt": "2023-02-08T15:32:23.350Z",
  "firstPublishedAt": "2021-04-12T10:08:01.671Z",
  "createdBy": {
    "sys": {
```

SDK - Setup

```
src > index.tsx > init() callback
1  import React from 'react';
2  import { render } from 'react-dom';
3  import { createClient } from 'contentful-management';
4
5  import {
6    AppExtensionSDK,
7    FieldExtensionSDK,
8    SidebarExtensionSDK,
9    DialogExtensionSDK,
10   EditorExtensionSDK,
11   PageExtensionSDK,
12   init,
13   locations,
14 } from '@contentful/app-sdk';
15 import type { KnownSDK } from '@contentful/app-sdk';
16 import '@contentful/forma-36-react-components/dist/styles.css';
17 import '@contentful/forma-36-fcss/dist/styles.css';
18 import '@contentful/forma-36-tokens/dist/css/index.css';
19 import './index.css';
20
21 import ConfigScreen from './components/ConfigScreen';
22 import EntryEditor from './components/EntryEditor';
23 import Page from './components/Page';
24 import Sidebar from './components/Sidebar';
25 import Field from './components/Field';
26 import Dialog from './components/Dialog';
27 import LocalhostWarning from './components/LocalhostWarning';
```

SDK - Setup

```
29 if (process.env.NODE_ENV === 'development' && window.self === window.top) {
30   // You can remove this if block before deploying your app
31   const root = document.getElementById('root');
32
33   render(<LocalhostWarning />, root);
34 } else {
35   init((sdk: KnownSDK) => {
36     const root = document.getElementById('root');
37
38     // Creating a CMA client allows you to use the contentful-management library
39     // within your app. See the contentful-management documentation at https://contentful.github.io/contentful-management.js/contentful-management/latest/
40     // to learn what is possible.
41     const cma = createClient(
42       { apiAdapter: sdk.cmaAdapter },
43       {
44         type: 'plain',
45         defaults: {
46           environmentId: sdk.ids.environment,
47           spaceId: sdk.ids.space,
48         },
49       }
50     );
```

SDK - Setup

```
52 // All possible locations for your app
53 // Feel free to remove unused locations
54 // Dont forget to delete the file too :)
55 const ComponentLocationSettings = [
56   {
57     location: locations.LOCATION_APP_CONFIG,
58     component: <ConfigScreen cma={cma} sdk={sdk as AppExtensionSDK} />,
59   },
60   {
61     location: locations.LOCATION_ENTRY_FIELD,
62     component: <Field cma={cma} sdk={sdk as FieldExtensionSDK} />,
63   },
64   {
65     location: locations.LOCATION_ENTRY_EDITOR,
66     component: <EntryEditor cma={cma} sdk={sdk as EditorExtensionSDK} />,
67   },
68   {
69     location: locations.LOCATION_DIALOG,
70     component: <Dialog cma={cma} sdk={sdk as DialogExtensionSDK} />,
71   },
72   {
73     location: locations.LOCATION_ENTRY_SIDEBAR,
74     component: <Sidebar cma={cma} sdk={sdk as SidebarExtensionSDK} />,
75   },
76   {
77     location: locations.LOCATION_PAGE,
78     component: <Page cma={cma} sdk={sdk as PageExtensionSDK} />,
79   },
80 ];
81
82 // Select a component depending on a location in which the app is rendered.
83 ComponentLocationSettings.forEach((componentLocationSetting) => {
84   if (sdk.location.is(componentLocationSetting.location)) {
85     render(componentLocationSetting.component, root);
86   }
87 });
88 });
89 }
```

Contentful Apps

Custom apps

Installed

	<div>Develop Deployer Custom</div> <div>ID: 5sv00NRKHHmDS68IfQESZN </div>	
	<div>Missing Snippets Custom</div> <div>ID: 6haw0B4Mzm0Sijpv3ZvHqS </div>	

Available

Contentful Apps



Dialog.tsx

```
Dialog.tsx M X  
src > components > Dialog.tsx > ...  
1  import React from "react";  
2  import { createClient, PlainClientAPI } from "contentful-management";  
3  import {  
4    Paragraph,  
5    Accordion,  
6    AccordionItem,  
7    List,  
8    ListItem,  
9    Note,  
10   TextLink,  
11  } from "@contentful/forma-36-react-components";  
12  import getBufferedFetch from "../getBufferedEntries";  
13  import { DialogExtensionSDK } from "@contentful/app-sdk";  
14  interface DialogProps {  
15    sdk: DialogExtensionSDK;  
16    cma: PlainClientAPI;  
17  }  
18  
19  type ErrLog = { locale: string; snippets: { name: string; id: string }[] };  
20
```




Dialog.tsx

```
21 const Dialog = (props: DialogProps) => {
22   const [allMissingSnippets, setAllMissingSnippets] = React.useState<ErrLog[]>(
23     []
24   );
25   const [dataFetched, setDataFetched] = React.useState(false);
26   const cma = createClient({ apiAdapter: props.sdk.cmaAdapter });
27   // link to the contentful-management library by CMA
28   const allLocales = props.sdk.locales.available;
29   // allLocales is an array of all the locales available in the space (lusini)
30
31   React.useEffect(() => {
32     async function fetchData() {
33       const space = await cma.getSpace(props.sdk.ids.space);
34       // space (lusini) is the space object from the contentful-management library
35       const environment = await space.getEnvironment(props.sdk.ids.environment);
36       // environment is branch of the space object
37       const allSnippets = await getBufferedFetch(environment, {
38         content_type: "snippets",
39         limit: 3000,
40         bufferSize: 250,
41       });
42       let allMissing: ErrLog[] = [];
43     }
```



Dialog.tsx

```
44 allLocales.forEach((loc) => {
45   // loc is the locale from the allLocales array from the contentful
46   allSnippets.forEach((snippet) => {
47     // snippet is of allSnippets array from the contentful
48     if (!snippet.value) {
49       console.log("Likely there is an Untitled Snippet!");
50       // https://contentful-missing-snippets.netlify.app/
51     }
52     if (!(loc in snippet.value)) {
53       // Handling missing snippet
54       if (!allMissing.find((snips) => snips.locale === loc)) {
55         // If the locale is not found in allMissing array, add it.
56         allMissing.push({
57           locale: loc,
58           snippets: [
59             {
60               // de-DE acts like a reference to the snippet props from the contentful (like a basket)
61               name: snippet.fieldName["de-DE"],
62               id: snippet.contentfulID,
63             },
64           ],
65         });
```



Dialog.tsx

```
66 |         } else {
67 |             // If the locale is found, update the existing entry.
68 |             // This will data set will be displayed in the contentful web app as DOM Dialogue box
69 |             // mapping list into proper shape
70 |             allMissing = allMissing.map((sni) => {
71 |                 if (sni.locale === loc) {
72 |                     //sni.locale is the locale assainged from the sni object
73 |                     return {
74 |                         locale: loc,
75 |                         snippets: [
76 |                             ...sni.snippets, // not to lose the previous entries
77 |                             {
78 |                                 name: snippet.fieldName["de-DE"],
79 |                                 id: snippet.contentfulID,
80 |                             },
81 |                         ],
82 |                     };
83 |                 } else {
84 |                     return sni;
85 |                 }
86 |             });
87 |         }
88 |     }
89 | });
90 | });
91 | setAllMissingSnippets(allMissing);
92 | setDataFetched(true);
93 | }
94 | if (!dataFetched && allMissingSnippets.length === 0) {
95 |     fetchData();
96 |     return;
97 | }
98 | if (dataFetched && allMissingSnippets.length === 0) props.sdk.close();
99 | }, [allMissingSnippets, allLocales, cma, props.sdk, dataFetched]);
100 |
101 | if (allMissingSnippets.length === 0) return <></>;
102 |
```

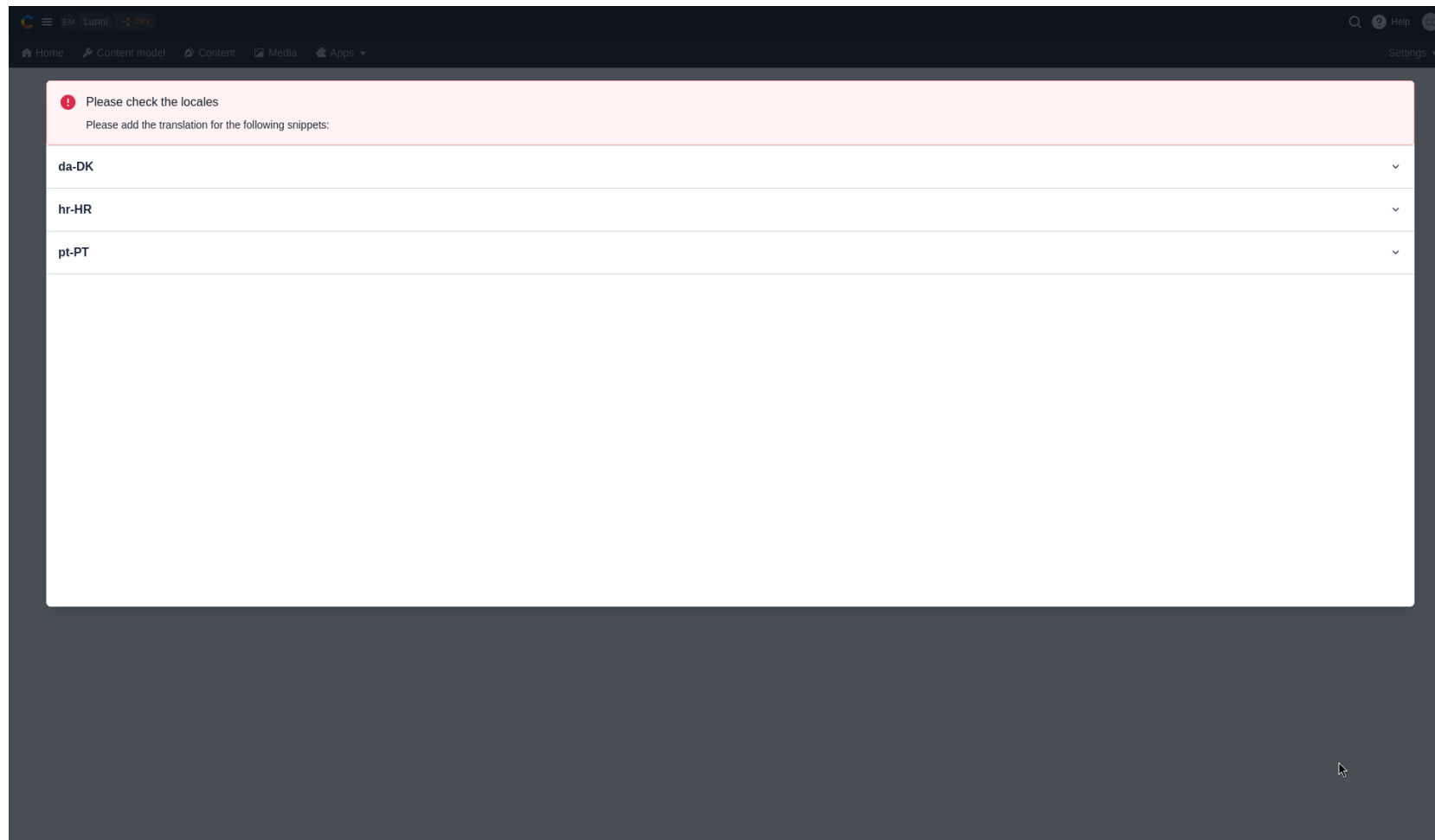
Dialog.tsx

```
// The following code is used to display the missing snippets in the contentful web app as a dialoge box location, when allMissingSnippets is not empty.
return (
  <Paragraph>
    <Note title="Please check the locales" noteType="negative">
      Please add the translation for the following snippets:
    </Note>

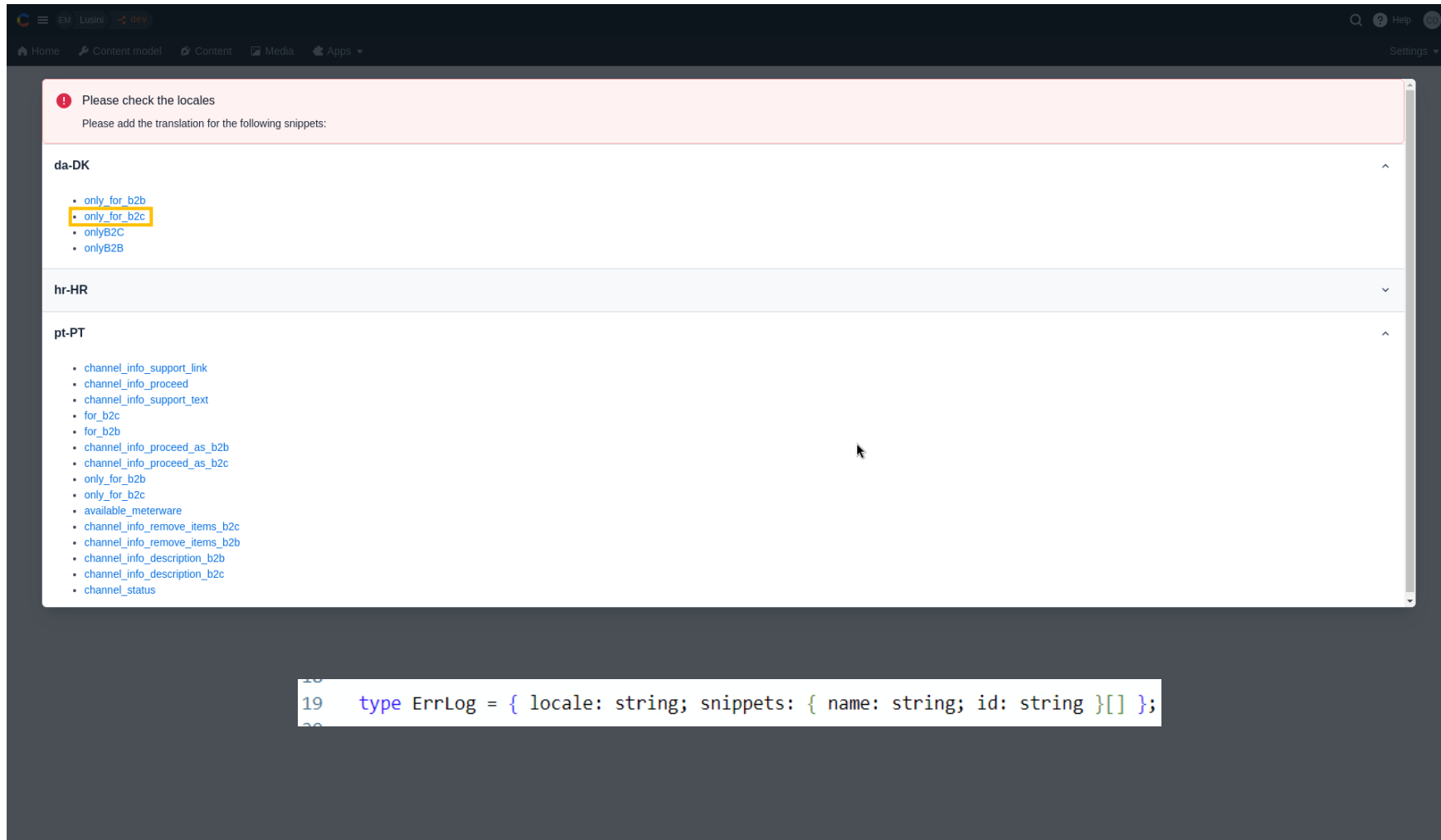
    <Accordion>
      {allMissingSnippets.map((missingSnippet) => {
        return (
          <AccordionItem
            key={missingSnippet.locale}
            title={missingSnippet.locale}
          >
            <List>
              {missingSnippet.snippets.map((snippet, i) => {
                return (
                  <ListItem key={i}>
                    <TextLink
                      linkType="primary"
                      href={
                        "https://app.contentful.com/spaces/aza65graowyr/entries/" +
                        snippet.id
                      }
                      target={"_blank"}
                    >
                      {snippet.name}
                    </TextLink>
                  </ListItem>
                );
              })}
            </List>
          </AccordionItem>
        );
      })}
    </Accordion>
  </Paragraph>
);

export default Dialog;
```

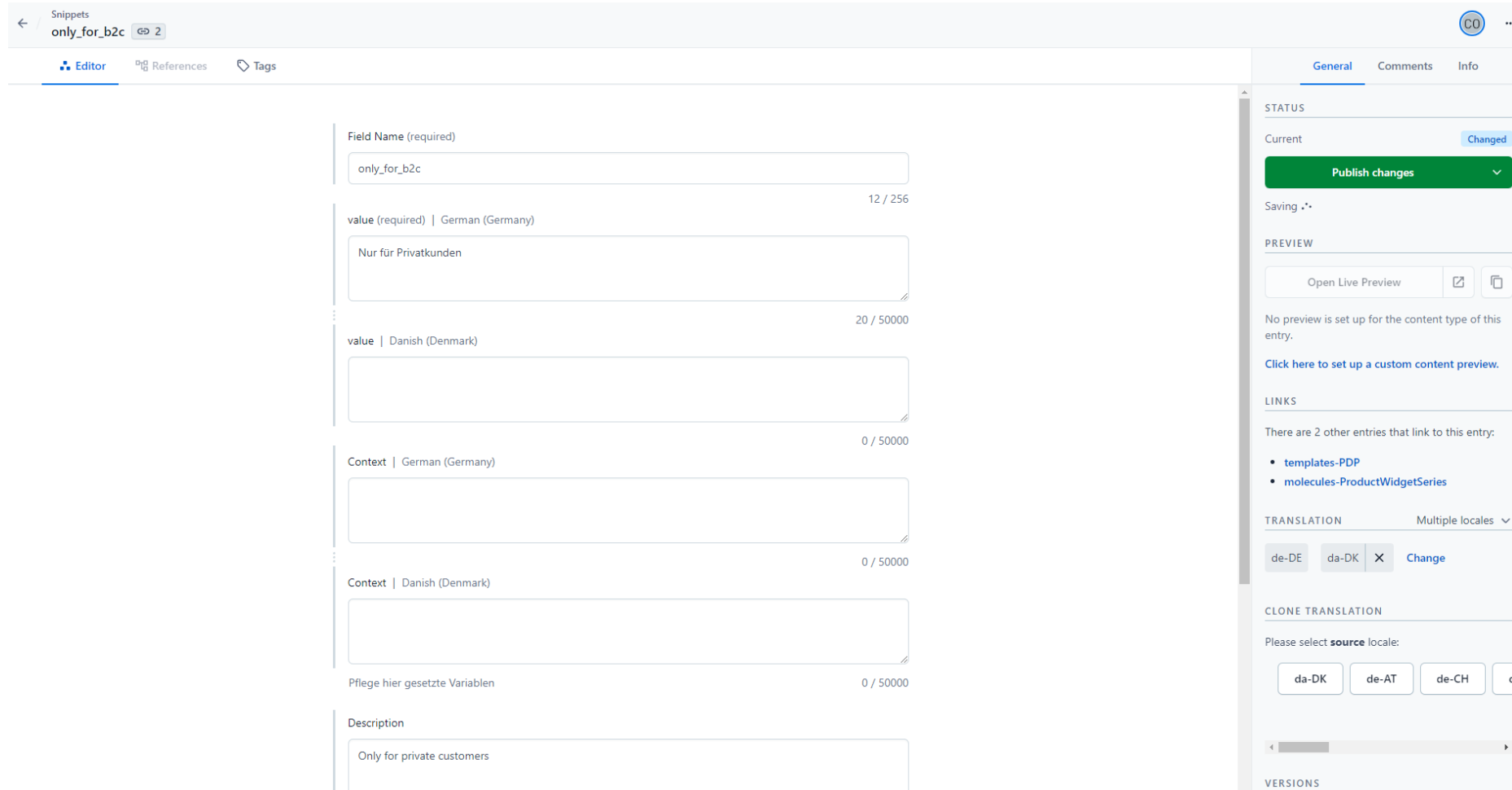
Dialog UI



Dialog UI



Snippets Feld



The screenshot shows the 'Snippets' editor interface for a field named 'only_for_b2c'. The interface is divided into two main sections: the main editor on the left and a right-hand sidebar with various controls.

Main Editor:

- Field Name (required):** A text input field containing 'only_for_b2c' with a character count of 12 / 256.
- value (required) | German (Germany):** A text input field containing 'Nur für Privatkunden' with a character count of 20 / 50000.
- value | Danish (Denmark):** An empty text input field with a character count of 0 / 50000.
- Context | German (Germany):** An empty text input field with a character count of 0 / 50000.
- Context | Danish (Denmark):** An empty text input field with a character count of 0 / 50000.
- Pflege hier gesetzte Variablen:** A text input field with a character count of 0 / 50000.
- Description:** A text input field containing 'Only for private customers'.

Right-hand Sidebar:

- General:** The active tab, showing the 'STATUS' section with a 'Current' status and a 'Publish changes' button.
- Comments:** A tab for adding comments.
- Info:** A tab for additional information.
- PREVIEW:** A section for previewing the content, with a note that no preview is set up for this content type.
- LINKS:** A section showing that there are 2 other entries that link to this entry: 'templates-PDP' and 'molecules-ProductWidgetSeries'.
- TRANSLATION:** A section for managing translations, showing 'Multiple locales' and buttons for 'de-DE', 'da-DK', and 'Change'.
- CLONE TRANSLATION:** A section for cloning translations, with a prompt to select a 'source' locale and buttons for 'da-DK', 'de-AT', and 'de-CH'.
- VERSIONS:** A section for managing versions, with a scrollable list of versions.

Test

Unit-Testing



PASS src/components/ConfigScreen.spec.tsx
PASS src/components/Field.spec.tsx
PASS src/components/EntryEditor.spec.tsx
PASS src/components/Sidebar.spec.tsx
PASS src/components/Page.spec.tsx
PASS src/components/Dialog.spec.tsx

Test Suites: 6 **passed**, 6 total
Tests: 6 **passed**, 6 total
Snapshots: 0 total
Time: 1.406 s



✓ 1/1 4.0s ↺
✓ ✓ contentful-extension-missing-snippets (watch)
✓ ✓ src ▶ 📄
 ✓ components
 > ✓ ConfigScreen.spec.tsx
 ✓ Dialog.spec.tsx
 ✓ Dialog component
 ✓ Component text exists
 > ✓ EntryEditor.spec.tsx
 > ✓ Field.spec.tsx
 > ✓ Page.spec.tsx
 > ✓ Sidebar.spec.tsx

Unit-Testing



LUSINI

```
Run | Debug
7 it("renders missing snippets when there are some", async () => {
8   const missingSnippets = [
9     { locale: "en-US", snippets: [{ name: "Snippet 1", id: "123" }] }
10  ];
11  const { getByText } = render(
12    <Dialog
13      sdk={{ close: jest.fn() }}
14      getSpace={jest.fn(() => ({
15        getEnvironment: jest.fn(() => ({
16          getEntries: jest.fn(() => ({
17            items: [
18              {
19                fields: {
20                  locale: "en-US",
21                  snippets: [{ name: "Snippet 1", id: "123" }]
22                }
23              },
24            ],
25          })),
26        })),
27      })),
28    </>
29  );
30
31  await waitFor(() => {
32    missingSnippets.forEach((missingSnippet) => {
33      expect(getByText(missingSnippet.locale)).toBeInTheDocument();
34      missingSnippet.snippets.forEach((snippet) => {
35        expect(getByText(snippet.name)).toBeInTheDocument();
36      });
37    });
38  });
39 });
40 });
```

```
1 import React from 'react'; 7.3k (gzipped: 3k)
2 import ConfigScreen from './ConfigScreen';
3 import { render } from '@testing-library/react'; 248.2k (gzipped: 60.5k)
4 import { mockCma, mockSdk } from '../test/mocks';
5
Run | Debug
6 describe('Config Screen component', () => {
7   Run | Debug
8   it('Component text exists', async () => {
9     const { getByText } = render(<ConfigScreen cma={mockCma} sdk={mockSdk} />);
10
11     // simulate the user clicking the install button
12     await mockSdk.app.onConfigure.mock.calls[0][0]();
13
14     expect(
15       getByText('Welcome to your contentful app. This is your config page.')
16     ).toBeInTheDocument();
17   });
18 });
```

Fazit & Ausblick

Fazit – 6 Projektmerkmale

Nach Dr. Alexander Blumenau



☒ Ziel?

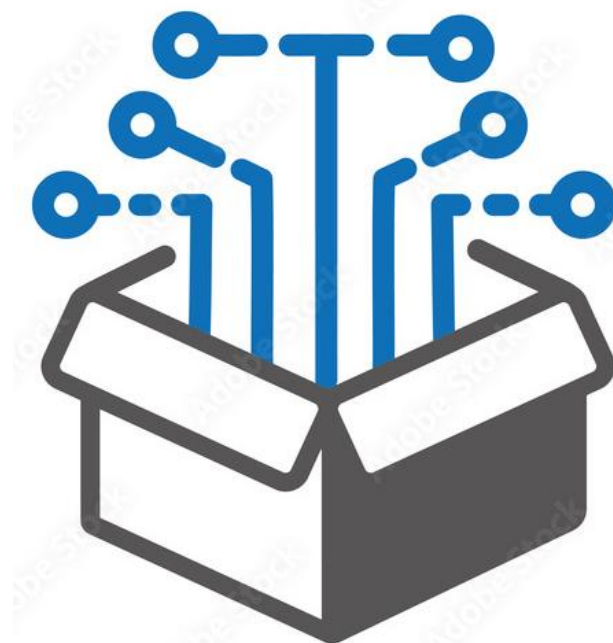
☒ Projektspezifische Organisation?

☒ Zeitliche Begrenzung?

☒ Neuartig & Einmalig?

☒ Begrenzte Ressourcen?

☒ Komplex?



OPEN SOURCE

Bildquellen

- Seite 5

Ikonen:

<https://www.flaticon.com/>

- Seite 6

Lusini Hauptsitz Wertigen:

<https://www.lusini-corporate.com/de/karriere/standorte/>

- Seite 7

Landes Flaggen:

<https://www.countryflags.com/>

- Seite 36

ChatGPT Logo:

https://de.m.wikipedia.org/wiki/Datei:ChatGPT_logo.svg

Open Source Symbol:

<https://www.iconbolt.com/iconsets/remix-icon-fill/open-source>

- Seite 15

Agiles Logo:

<https://www.cleanpng.com/png-agile-software-development-scrum-computer-icons-ag-2145050/>

TypeScript Logo:

<https://az.wikipedia.org/wiki/TypeScript>

Jest Logo:

<https://logowik.com/jest-js-logo-vector-svg-png-free-download-24413.html>

VSC Logo:

<https://blog.noblinkyblink.com/2017/09/25/connecting-visual-studio-code-to-github/vsc-logo/>

Contentful Logo:

<https://seeklogo.com/vector-logo/304532/contentful>

Github Action Logo:

<https://marketplace.visualstudio.com/items?itemName=GitHub.vscode-github-actions>

Vielen Dank für Ihre Aufmerksamkeit.

Für Rückfragen stehe ich Ihnen sehr gerne zur Verfügung.