

Allgemeine Beschreibung

Der *TEI Publisher* wurde mit dem Ziel entwickelt, Forschenden und Webredakteur:innen ein leistungsstarkes Werkzeug zur Verfügung zu stellen, das ohne umfangreiche Programmierkenntnisse die Veröffentlichung von Materialien ermöglicht. Nutzer:innen werden bei der Arbeit mit dem *TEI Publisher* nicht in ein starres Framework gezwängt, sondern können flexibel agieren. Auch erfahrene Entwickler:innen profitieren von diesem Tool, da sich ihr Programmieraufwand durch das Tool reduzieren lässt.

Der *TEI Publisher* bietet ein Framework bzw. eine Bibliothek für die Gestaltung einer digitalen Edition mit vorgefertigten Komponenten und verfügt über eine eigene interne Arbeits- und Verwaltungsumgebung, die die Bearbeitung von ODD-Dateien, die Verwaltung und Annotation der projektspezifischen Daten sowie eine Anbindung an externe Dienste über klar definierte APIs ermöglicht. Die Kernfunktionalität des *TEI Publisher* liegt in der Veröffentlichung von TEI-XML-Dokumenten in verschiedenen Ausgabeformaten wie HTML oder PDF. Die beiden grundlegenden TEI-Publisher-Komponenten dafür sind erstens die *TEI Publisher Lib*, die das [TEI Processing Model](#) implementiert und die Transformation der XML-Dateien zu HTML-Output ermöglicht, und zweitens der Baukasten für die Benutzeroberfläche - HTML5-Webkomponenten für Buttons, Navigation, Suchfelder etc. -, die es ermöglichen, Vorlagen für die Webseiten der digitalen Edition zu erstellen. Ein zusätzliches Feature ist der Annotationseditor, der sich derzeit noch im Ausbau befindet.

Im Fokus steht nicht nur die Einhaltung von Standards, sondern auch die Nutzung von Modulbauweisen, die einerseits die Wiederverwendbarkeit fördert und andererseits zur Sicherstellung von langfristiger Nachhaltigkeit beiträgt.

Schematischer Ablauf der Datenaufbereitung im TEI Publisher: Daten(-stand) → [Datenbank](#) → [Abfrage](#) → [Prozessierung](#) → [Aufbereitung](#) → Darstellung

Anwendungsbereiche

- Transformation von Importformaten wie TEI, DocBook, MS Word (DOCX) oder JATS zu Exportformaten wie HTML, eBook-Formaten, PDF oder LaTeX
- Publikation in Web und Druck
- TEI-konforme Annotation

Funktionsübersicht

- Benutzer- und Datenverwaltungssystem
- Kollaborationsmöglichkeit (über eXist)
- Erstellung von einfachen digitalen Editionen mit:
 - synoptischer Text-Faksimile-Ansicht
 - benutzerdefiniertem Rendering des Textes
 - Anzeige von Informationen zu verschiedenen Szenarien
- Formularbasierte Arbeitsumgebung für die [ODD](#)-Erstellung und -Anpassung
- Annotationsmodul - Möglichkeit zur rudimentären Auszeichnung von XML-TEI-Daten in graphischer Oberfläche (Voraussetzung ist, dass das XML-TEI bereits minimalen Anforderungen entspricht)
- Webkomponenten (Navigation, Paginierung, Suchfeld, Faksimile-Anzeige, Sprachauswahl usw.)
- Anbindung an externe Dienste (über APIs)
- Diverse Ansichtsmöglichkeiten: Web-Ansicht und PDF-Vorschau

Voraussetzungen

Jedes Tool kann einerseits bestimmte Vorkenntnisse der Benutzer:innen voraussetzen und andererseits auch hinsichtlich der Software-Umgebung gewisse Anforderungen stellen.

Erforderliche Kenntnisse

- TEI-XML
- XPath/XQuery
- HTML/CSS Grundkenntnisse

Benötigte Software

- eXist-db
- Docker Desktop (für eine einfachere Installation von eXist-db)

Tool-Kompatibilität

	IIIF	Transkribus	FromThePage	FairCopy	ediarum	OpenRefine	ba[sic?]	ediarum.WEB
TEI Publisher	<input checked="" type="checkbox"/>							

Legende

- Integrationen vorhanden und Entwickler:innen bewerben Übergänge
- Kein Entwicklung einer Transition im DigEdTnT-Projekt, keine Entwicklerintegration beschrieben
- Transition im DigEdTnT-Projekt erstellt

Kostenübersicht

- **TEI Publisher & eXist-db:**

- kostenlos

Möglichkeiten & Grenzen

Da jedes Projekt unterschiedliche Anforderungen mit sich bringt, sollen nachfolgend mögliche Vor- und Nachteile des getesteten Tools dargestellt werden.

Stärken

- Kollaborationsmöglichkeit
- Nachhaltigkeit und Austauschbarkeit (durch die Möglichkeit der Weiterverwendung und Anpassung von Daten aus Beispielprojekten, die mit der Installation mitgeliefert werden)
- Bereitstellung von Bausteinen einer digitalen Edition (Webkomponenten) zur Navigation, Paginierung, Suche, Faksimile-Anzeige
- Formularbasierter Editor zur Bearbeitung des ODD
- Eignung für jede Art von XML, nicht nur TEI
- Erstellung von hoch-qualitativem und kameratauglichem Material für Buchveröffentlichungen
- Sehr hilfsbereite und reaktionsschnelle Slack-Community

Herausforderungen & Probleme

- Implementierung und Konfiguration von Projekten mit komplexeren Publikationsansichten (z. B. verschiedene ODDs für verschiedene XML-Dateien oder bei erwünschter Verknüpfung von XML-Dateien, wie bspw. Manuskript und Register) erfordert DH-Entwickler:in, da mitunter viel technische Anpassungsarbeit notwendig ist
- Abhängigkeit von eXist-db
- Cache-Probleme im Browser, sodass Änderungen teilweise erst nach Leeren des Cache angezeigt werden
- Informationsarme Fehlermeldungen im ODD-Editor (Fehlermeldung bei erfolglosem Speichern gibt nur "Error" an, nicht jedoch die Ursache)

Einrichtung & Erste Schritte

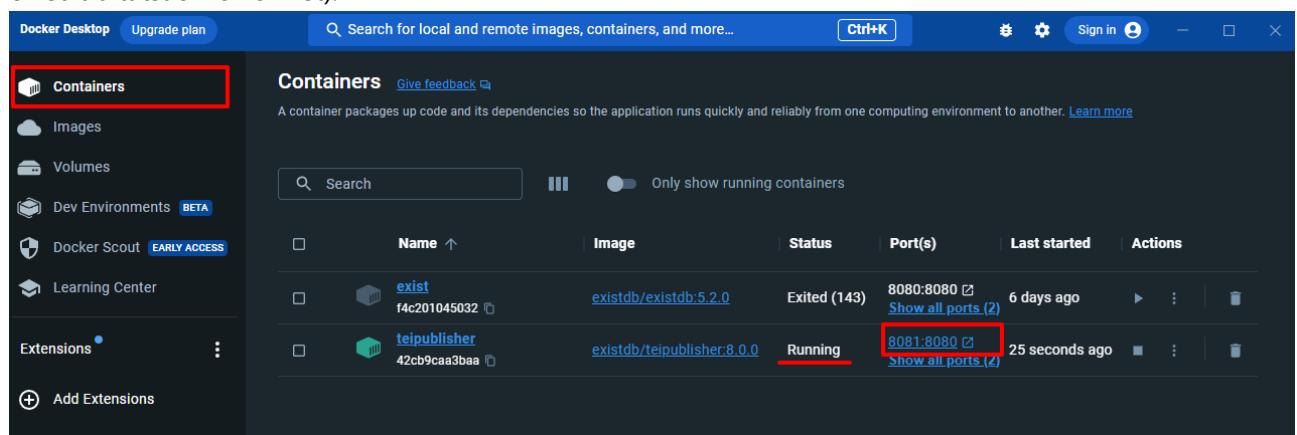
Anhand eines [Beispielprojekts](#), das zum Ziel hat, Kochrezepte aus dem Mittelalter computergestützt zu analysieren und anschließend über eine Forschungsplattform zur Verfügung zu stellen, soll nachfolgend ein möglicher Arbeitsablauf beschrieben werden. Die dafür verwendeten Daten wurden bereits mit dem Tool [FromThePage](#) transkribiert, in [ediarum](#) annotiert und mit Normdaten, die in [OpenRefine](#) bearbeitet wurden, angereichert. Der letzte Abschnitt des Beispielprojekts soll sich nun mit den Möglichkeiten einer Publikation der digitalen Edition mittels *TEI Publisher* beschäftigen.

1. Installation einzelner Komponenten

Da die Grundlage des *TEI Publisher* - ebenso wie bei [ediarum](#) - eine eXist-Datenbank ist, nutzen wir unsere bereits vorhandene Struktur mit Docker Desktop. Hier geht es zur detaillierten [Installationsanleitung für Docker Desktop sowie eines Containers für den TEI Publisher](#).

2. Einrichtung des Projekts

- Wir beginnen mit der Einrichtung unseres Projekts, indem wir uns im *TEI Publisher* anmelden. Dafür müssen wir nur sicherstellen, dass wir die eXist-db über den Docker Container gestartet haben. Mit einem Klick auf den Port öffnet sich im Browser die eXist-db (- mitunter kann es 1-2 Minuten dauern, bis sich die eXist-db tatsächlich öffnet).



Anschließend wählen wir in der eXist-db die TEI-Publisher-Applikation.

The screenshot shows the eXist-db dashboard with various links and icons. The 'TEI Publisher' icon, which is a stylized 'X' with horizontal bars, is highlighted with a red box. Other visible links include 'eXide - XQuery IDE', 'eXist-db Documentation', 'Monex', and 'Shakespeare's Plays (TEI Publisher Edition)'.

- Im *TEI Publisher* gibt es bereits zwei vorangelegte Standard-User (tei-demo und tei), für die es auf der Startseite des *TEI Publisher* auch eine kleine Infobox mit Angaben zum Passwort gibt.

The screenshot shows the TEI Publisher interface. At the top right, there is a 'Anmelden' (Login) button. A red box highlights this button, and a red arrow points to it from the text above. Below the login button, a tooltip says: 'Log in as **tei-demo** with password **demo** to experiment with your own files and ODDs'. In the center, there is a section titled 'The Instant Publishing Toolbox' with a sub-section 'Spielwiese' (highlighted with a red box and labeled '2.') containing an image of a person on a swing. To the right, there is a sidebar titled 'ODD Dateien' with several entries, one of which is also highlighted with a red box.

→ Wir loggen uns als "tei-demo" mit dem Passwort "demo" ein, um anschließend die **Spielwiese** zu öffnen.

- In der Spielwiese können wir nun unsere XML-Dokumente - in unserem Fall die bereits in [ediarum](#) annotierten Manuskripte - hochladen.

The screenshot shows the TEI Publisher interface. On the left, the 'Dokumente' tab is active, displaying a list of imported XML files ('manuscript1.xml' and 'manuscript2.xml') in a table format. A red box highlights this table. On the right, the 'Hochladen' (Upload) panel is open, containing a text input field for file uploads and a message: 'Dateien vom Typ .xml, .tei, .odd, .docx, .md, .mei hier hinziehen zum Hochladen.' A red arrow points from the 'Hochladen' panel towards the 'Dokumente' table.

→ Unsere beiden Manuskripte scheinen im Anschluss im Dokumentenverzeichnis des *TEI Publisher* auf.

The screenshot shows the TEI Publisher interface. On the left, the 'Dokumente' tab is active, displaying a search interface with fields for title, query scope, search content, and language. It shows 2 entries found. In the main area, two manuscript entries are listed: 'MS Augsburg, Universitätsbibl., Cod.III.1.2°43 (A1)' and 'MS Solothurn, Zentralbibliothek, Cod. S 490 (So1)'. Each entry has a download button labeled 'HERUNTERLADEN'. On the right, the 'Hochladen' panel is open, showing the uploaded files 'manuscript1.xml' and 'manuscript2.xml' with checkmarks. A red 'X' icon is visible in the top right corner of the panel.

- Um ein projektspezifisches ODD zu erstellen, scrollen wir im rechten Seitenbereich unter **ODD Dateien** bis zum Seitenende, wo wir schließlich einen Dateinamen für unsere ODD sowie einen Titel für die Anzeige eingeben, und mittels "ERZEUGEN" ein eigenes ODD anlegen.

The screenshot shows the TEI Publisher Base interface. On the left, the 'Dokumente' (Documents) section displays search results for 'MS Augsburg, Universitätsbibl., Cod.III.1.2°43 (A1)' and 'MS Solothurn, Zentralbibliothek, Cod. S 490 (So1)'. Each entry has a trash icon and a download link labeled 'HERUNTERLADEN'. On the right, the 'Hochladen' (Upload) panel includes sections for 'HOCHLADEN' (Upload files of type .xml, .tei, .odd, .docx, .md, .mei), 'ODD Dateien' (ODD files), and 'ERZEUGEN' (Create). A red box highlights the 'ODD Dateien' section, which lists 'Erasmus Adagia', 'Dantiscus' Letters Demo', 'Leewel Letter', and 'Van Gogh Letters'. Another red box highlights the 'ERZEUGEN' section, which contains fields for 'Dateiname des ODD (ohne Suffix)' (ODD filename (without suffix)) set to 'MA Rezepte', and 'Titel für die Anzeige' (Title for display) also set to 'MA Rezepte'. Red arrows point from the 'Leewel Letter' and 'Van Gogh Letters' entries in the 'ODD Dateien' list to their respective edit icons.

→ Das neue ODD erscheint anschließend als eigenständige Instanz in der Liste.

- Wir wählen nun eines unserer Dokumente in der Spielwiese und öffnen dieses. Wie wir an der Kursiv-Schreibung der ersten Zeilen erkennen können, wurde unser Dokument bereits mit einem Standard-ODD verknüpft. Mit einem Klick auf das Hamburger-Symbol in der Navigationsleiste lassen sich die Einstellungen zu unserem Dokument öffnen.

The screenshot shows the document view for 'MS Augsburg, Universitätsbibl., Cod.III.1.2°43 (A1)'. The top navigation bar includes icons for document, file, copy, search, and more. The main content area displays a block of medieval German text in a Gothic script. The text reads:

Hienach volgenn Erlich eritznej vonn mangerlaj dingenn zu sammenn gesammer
 Als du danne bernach wol vindenn wurdest
Ein purgatz Zu der zejtt des herbstes furr allenn gebrechenn
 [N]ym hieffenn die zeÿtig sein vnd mit dem reÿf wol beczwungenn
 vnd nÿm die mit kernn vnd mit all vnd derre die in eyнем ofenn vnd
 lege sie danne herauß vnd laß den windt dar durch genn vnd
 nÿm dann einen moersert vnd stoß die hieffenn vnd misch die mit honig
 vnd das isse das des morgens nüchternn
 Meßigkeit Ist ein aufenthaltung des ganntzen leichnams . Vnd ein starck
 aller gelider . Vnd ein außderung aller boesenn duenst Wann meßigkeit ist
 ein zustoerung aller lasterr
 Wiltu gut lekuchenn machenn Recipe zu einem seÿdlein hoenigs 1/2 lot negelleyn
 1/2 lot muscat 1/2 lot Ingwert 1/2 lot pfeffers vnd stoß das zu sammenn vnd dar
 auß mach einen teig vnd zu einer moß honigs nÿm zwirt soulis als vor be=

A circular navigation arrow is visible in the bottom right corner of the text area.

Bei der Auswahl der ODD-Datei wählen wir nun anstelle des voreingestellten "TEI Publisher Base" unser soeben erstelltes, eigenes ODD-Dokument ("MA Rezepte").

The screenshot shows the TEI Publisher interface. On the left, a manuscript page from 'MS Augsburg, Universitätsbibl., Cod.III.1.2*43 (A1)' is displayed with text in Middle High German. On the right, there's a sidebar titled 'Einstellungen' (Settings) with a dropdown menu set to 'MA Rezepte'. Below it are sections for 'Template Auswählen' (Select Template) and 'Default single text layout'. A warning box titled 'Achtung' (Attention) provides instructions on how to use ODDs and templates.

→ Da das ODD vorerst noch leer ist bzw. auf dem Standard-ODD im *TEI Publisher* basiert, ergeben sich vorerst keine Änderungen.

3. Bearbeitung der Dokumente

Für eine vollständige Publikationsansicht benötigt man die TEI-XML-Ressource, die über den *TEI Publisher* in die eXist-db hochgeladen werden, sowie zumindest ein ODD mit Processing Instructions und ein Page Template in HTML. Das ODD wird dabei auf Grundlage eines Basis-ODD im *TEI Publisher* erstellt und kann nachfolgend für das eigene Projekt angepasst werden. Für das Page Template ist es am sinnvollsten, sich in der Demo-Sammlung des *TEI Publisher* Beispielprojekte genauer anzusehen, und sich von dort ein passendes Template für das eigene Projekt zu kopieren und anzupassen.

a. Bearbeitung des ODD der Manuskripte

- Bei jedem Start unserer Arbeitsroutine müssen wir zuerst Docker Desktop aktivieren und anschließend den teipublisher-Container starten, indem wir auf die Play-Schaltfläche klicken.

The screenshot shows the Docker Desktop interface. On the left, a sidebar lists 'Containers', 'Images', 'Volumes', 'Dev Environments (BETA)', 'Docker Scout (EARLY ACCESS)', and 'Learning Center'. The main area is titled 'Containers' and shows a table of running containers. The 'teipublisher' container, which has been started, is highlighted with a red box around its play button in the 'Actions' column.

Name	Image	Status	Port(s)	Last started	Actions
exist	existdb/existdb:5.2.0	Exited (143)	8080:8080	6 days ago	▶ ⋮ 🗑
teipublisher	existdb/teipublisher:8.0.0	Exited (143)	8081:8080	5 minutes ago	▶ ▶ ⋮ 🗑

→ Mit einem Klick auf den Port (8081:8080) öffnet sich schließlich im Browser das eXist-db-Dashboard, über das wir zum *TEI Publisher* gelangen, in dem wir uns einloggen müssen, um unsere Arbeit fortsetzen zu können.

- Um das eigens angelegte ODD anzupassen und damit unsere annotierten XML-Dateien für die Publikation aufzubereiten, gehen wir in der Navigation auf **Administration** und wählen dort "ODD editieren". (Der

Administration-Button steht im Übrigen nur zur Verfügung, wenn man eingeloggt ist.)

The screenshot shows the TEI Publisher interface. At the top, there is a navigation bar with links: Start, Dokumentation, Neuigkeiten, DTS, Herunterladen, Administration, and a search bar. The 'Administration' link is highlighted with a red box. Below the navigation bar, there is a toolbar with icons for file operations like 'ODD editieren' (highlighted with a red box), 'Anwendung generieren', and 'Train Named Entity Recognition'. The main content area displays a manuscript page from 'MS Augsburg, Universitätsbibl., Cod.III.1.2*43 (A1)' containing Middle High German text. A large red arrow points to the right side of the page.

- In der ersten Ansicht des bisher noch leeren ODD können wir nun die Metadaten ausklappen und für unser Projekt anpassen.

The screenshot shows the configuration page for 'MA Rezepte.odd'. On the left, there is a sidebar with 'Element hinzufügen' and 'Springe zu Element ...'. The main area is titled 'MA Rezepte' and contains the following fields:

- Titel:** MA Rezepte
- Kurztitel:** Mittelalterliche Kochrezepte
- Beschreibung:** ODD für die digitale Edition einer mittelalterlichen Manuskriptsammlung mit Rezepten
- Basis ODD:** [Basis-ODD, welches erweitert wird]
- Auswählen, um einen anderen Namespace als TEI zu benutzen:** (checkbox)
- Namespace:** [Default Namespace URI (wenn nicht TEI)]
- External CSS File:** [External CSS file with additional class definitions]

- Um die einzelnen TEI-Elemente, mit denen wir unsere Manuskripttexte annotiert haben, für die Publikation aufzubereiten, müssen wir als Erstes im ODD das Element angeben, das wir bearbeiten möchten. Sollten wir nicht (mehr) wissen, welche TEI-Elemente wir bei der Annotation verwendet haben, können wir in der Navigationsleiste des **TEI Publisher** auf **Herunterladen** klicken und im Drop-Down-Menü "XML" auswählen.

The screenshot shows the 'Herunterladen' menu open. The menu items are: Print CSS, PDF (LaTeX Version), and XML (highlighted with a red box). Below the menu, the manuscript page is shown again with the 'XML' option highlighted.

Das XML unseres Manuskriptes öffnet sich daraufhin in einem neuen Browserfenster in eXide.

The screenshot shows the eXide interface with the title bar "existdb-dashboard", "TEI Publisher", and "eXide". The main window displays an XML document titled "manuscript1.xml". The XML code is as follows:

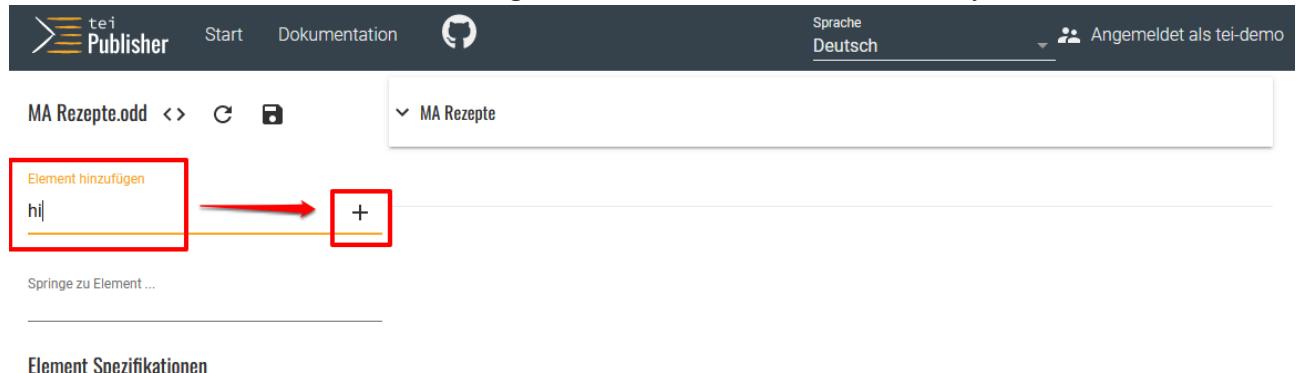
```

<TEI xmlns="http://www.tei-c.org/ns/1.0" xmlns:telota="http://www.telota.de" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xml:id="dtabf" telota:doctype="document">
  <teiHeader>
    <fileDesc>
      <titleStmt>
        <title type="main">MS Augsburg, Universitätsbibl., Cod.III.1.2°43 (A1)</title>
        <author>
          <persName>Unknown</persName>
        </author>
        <respStmt>
          <!-- HINWEIS: Das nachfolgende Element ist obligatorisch und sollte daher ausgefüllt werden! -->
          <orgName>Zentrum für Informationsmodellierung</orgName>
        <resp>
          <note type="remarkResponsibility">Erstellung der Transkription nach DTA-Richtlinien</note>
          <date type="importDTA" when="2023-05">Mai 2023</date>
        </resp>
      </respStmt>
    </titleStmt>
    <publicationStmt>
  </publicationStmt>

```

→ Bei der Arbeit im *TEI Publisher* empfiehlt es sich, stets mehrere Tabs geöffnet zu haben: die Manuskriptansicht, auf der man die Änderungen direkt nachverfolgen kann; die XML-Ansicht in eXide, um einen Überblick über die Annotationen zu haben; sowie den ODD-Editor, wo die einzelnen Elemente bearbeitet werden können.

- Zurück zum ODD-Editor: In der Formularansicht können wir nun das Layout der einzelnen Annotationen bzw. der entsprechenden TEI-Elemente gestalten. Als erstes möchten wir, dass in der Publikationsansicht die mit roter Farbe geschriebenen Textstellen ebenfalls in rot dargestellt werden. Wir geben also in das Feld mit der Überschrift **Element hinzufügen** "hi" ein und klicken auf das Plus-Symbol (+).



- Wenn wir das `<hi>`-Element aufklappen, öffnet sich ein Formular zur Bearbeitung.

The screenshot shows the tei Publisher interface for editing an 'hi' element. The 'behaviour' field is set to 'inline'. In the 'Renditions' section, a new entry 'color: red;' has been added, highlighted with a red box. The 'Use source rendition' checkbox is checked.

Standardmäßig ist als behaviour "inline" eingestellt, was wir in diesem Fall auch so belassen, da sich die roten Textstellen innerhalb des Fließtextes befinden. Etwas weiter unten im Formular klicken wir bei Rendition auf das Plus-Zeichen, um hier das Rendering festzulegen. Da wir möchten, dass die `<hi>`-Elemente, die für die Annotation roter Textstellen angelegt wurden, auch im Publikationsmedium eine rote Farbe erhalten, müssen wir in dem Feld die entsprechende [CSS-Syntax](#) für die Rotfärbung angeben. Die vom Standard-ODD übernommene Kursivschreibung für das `<hi>`-Element, die im oberen Rendition-Feld gespeichert ist, löschen wir. Zuletzt müssen wir zum Speichern unserer Modellspezifikation auf das Diskettensymbol klicken. Wenn wir anschließend in einem anderen Browser-Tab die Manuskript-Ansicht öffnen (oder das Tab aktualisieren), können wir unsere Änderungen bereits sehen.

The screenshot shows the MS Augsburg manuscript page with red text annotations. A red box highlights the 'C' icon in the toolbar, indicating the changes have been applied.

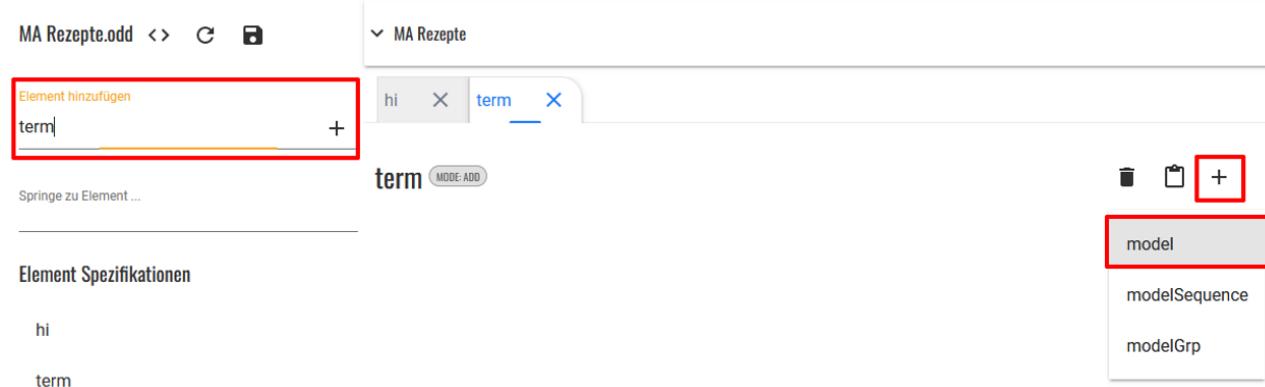
MS Augsburg, Universitätsbibl., Cod.III.1.2°43 (A1)

Hienach volgenn Etlich ertzney vonn mangerlay dingenn zu sammenn gesammet
 Als du danne hernach wol vindenn wurdest
 Ein purgatz Zu der zeÿt des herbstes furr allenn gebrechenn
 [N]ym hieffenn die zeÿtig sein vnd mit dem rejff wol beczwungenn
 vnd njm die mit kernn vnd mi all vnd derre die in eynem ofenn vnd
 lege sie danne herauß vnd laß den windt dar durch genn vnd
 njm dann einen moerserr vnd stoß die hieffenn vnd misch die mit honig
 vnd das isse das des morgens nüchternn
Meßigkeit Ist ein aufenthaltung des ganntzen leichnams . Vnd ein starck
 aller gelider . Vnd ein außderung aller boesenn duenst Wann meßigkeit ist
 ein zustoerung aller lasterr

→ Sollten die Änderungen nicht sichtbar sein, muss möglicherweise überprüft werden, ob das Dokument nach wie vor mit dem eigenen ODD verknüpft ist, da sich dies mitunter automatisch auf die Standardeinstellung zurücksetzt, oder ob man nach wie vor eingeloggt ist, da man teilweise bereits nach

kürzeren Bearbeitungspausen automatisch ausgeloggt wird. Es kann auch nötig sein, mit der F5-Taste zu aktualisieren oder den Cache zu leeren, um die Änderungen zu sehen.

- Wenn wir nun ein Element anlegen, das im Standard-ODD-Template noch keine Modellspezifikationen besitzt, wie in diesem Fall das `<term>`-Element, dann müssen wir nach dem Hinzufügen des Elements außerdem das Plus-Zeichen in der Elementansicht anklicken und aus dem Drop-Down auswählen, ob wir ein Modell, eine Modellsequenz oder eine Modellgruppe erstellen möchten.



Wir möchten unsere Zutaten mit einer punktierten Unterstreichung hervorheben und geben daher in dem Eingabefeld für die Rendition den entsprechenden CSS-Code an:

```
text-decoration-line: underline; text-decoration-style: dotted;
```

Außerdem möchten wir in der Webansicht ein Popover über das `<term>`-Element legen, in dem einerseits der Wikidata-Link zu der in [OpenRefine erstellten normalisierten Entität](#) und andererseits eine Information mit Link zum Register erscheint. Zuerst stellen wir dafür das behaviour auf "alternate" und geben zwei Parameter an. Einmal den "default"-Parameter, wo wir nur einen Punkt setzen, da wir standardmäßig den Inhalt des `<term>`-Elements angezeigt bekommen wollen, und als "alternate"-Parameter geben wir an, was im Popover angezeigt werden soll. Wir möchten hier gerne die Informationen aus dem Register anzeigen und wählen daher den entsprechenden XPath, der uns genau zu diesem Registereintrag führt:

```
id(substring-after(@key, '#'), doc("/db/apps/tei-publisher/data/playground/Sachbegriffe.xml"))
```

Die gesamte Modellspezifikation sieht schließlich folgendermaßen aus:

Wenn wir nun wieder zur Ansicht unseres Dokuments wechseln, sehen wir, dass die Zutaten unterstrichen sind und beim Bewegen des Mauszeigers über die Zutat ein Popover mit allen Informationen aus dem Register, die es für das `<item>`-Element, auf welches das `<term>`-Element referenziert, erscheint. Da wir für die Kindelemente des `<item>`-Elements noch keine Modelle erstellt haben, werden vorerst nur die Inhalte dieser Elemente in direkter Aneinanderreihung ausgegeben.

- Als nächstes passen wir die Modellspezifikationen der Kindelemente der Registereinträge an. Die `<item>`-Elemente in unserem Register (Sachbegriffe.xml), auf das wir verlinken, beinhalten ein `<idno>`-Element mit der Wikidata-Referenz, ein `<label>`-Element mit dem Attribut `@type="reg"` für die reguläre Schreibweise sowie ein `<label>`-Element mit dem Attribut `@type="alt"` für diverse frühneuhochdeutsche Varianten.

```

Sachbegriffe_...
15 <ex>
16 <body>
17 <list>
18 <item xml:id="verjuice">
19   <idno type="uri">https://www.wikidata.org/entity/Q1060458</idno>
20   <label type="reg">Agraz</label>
21   <label type="alt">agraeß</label>
22   <label type="alt">agras</label>
23   <label type="alt">agraz</label>
24   <label type="alt">agres</label>
25   <label type="alt">agresß</label>
26   <label type="alt">agrest</label>
27   <label type="alt">agreste</label>
28   <label type="alt">agroße</label>
29 </item>
30 <item xml:id="apple">
31   <idno type="uri">https://www.wikidata.org/entity/Q89</idno>
32   <label type="reg">Apfel</label>
33   <label type="alt">apfel</label>
34   <label type="alt">äpfelen</label>
35   <label type="alt">apfell</label>
36   <label type="alt">apffel</label>
37   <label type="alt">äpfel</label>
38   <label type="alt">apffell</label>

```

Da wir einerseits den Inhalt des `<idno>`-Elements als Link im Popover haben möchten, zusätzlich aber noch eine Zeile mit einem Verweis auf das Register, müssen wir für diese zwei Aktionen zum gleichen Element zuerst eine Modellsequenz anlegen. Das Arbeiten mit `<modelSequence>` ermöglicht es, mehrere Modelle für ein spezifisches Element anzulegen, ohne dass diese sich gegenseitig überschreiben. Alle in einer Modellsequenz befindlichen Modelle für ein Element werden nacheinander abgearbeitet. Wir beschränken diese Sequenz auf `<idno>`-Elemente mit dem Attribut `@type="uri"` und legen darin über das Plus-Symbol zwei weitere Modelle an.

The screenshot shows the 'verjuice' item configuration in the 'Sachbegriffe...' section. A red box highlights the XML code for the 'verjuice' item. Below it, the 'modelSequence' configuration for the 'apple' item is shown. A red box highlights the predicate '1 @type="uri"'. The interface includes buttons for mode selection (MODE ADD), navigation (down, up, back, forward, plus), and a 'Mode' dropdown set to 'Modus setzen für nachfolgende Modelle'.

- In der ersten neu angelegten Modellspezifikation wählen wir anschließend als behaviour "link" aus. Bei den Parametern wählen wir nun "uri" und verweisen dabei mit einem Punkt auf den Inhalt des

<idno>-Elementes.

- In der zweiten Modellspezifikation möchten wir nun einen kleinen Informationstext mit Link zum Zutatenregister erstellen. Dafür wählen wir als behaviour "block" und erstellen ein Template in HTML-Syntax für diesen Text, der in einem <div>-Element angezeigt werden soll:

```
<div>Siehe weitere Schreibweisen dieser Zutat im <a href="[[register]]" target="_blank">Register</a></div>
```

Da wir das Wort "Register" gleich mit unserem Zutatenregister verlinken wollen, setzen wir um dieses Wort einen Anker, und geben als Referenz den Parameter **[[register]]** an, den wir unter den Parametern dann mit dem entsprechenden XPath-Ausdruck näher spezifizieren. Damit sich das Register in einem neuen Tab im Browser öffnet, ergänzen wir das <a>-Element noch mit dem Attribut **@target="_blank"**. Der Link zum Register wird schließlich im Parameter weiter ausgeführt:

```
concat('Sachbegriffe.xml?view=div&#38;odd=ma-zutaten', '#',
parent:::item/@xml:id')
```

Um in unserem Zutatenregister direkt zu der Zutat zu springen, dessen Popover gerade betrachtet wird, müssen wir einerseits die entsprechende XML-Datei (Sachbegriffe.xml) angeben, die aufgerufen werden soll, und andererseits das ODD-Template (ma-zutaten ohne Dateiendung), das für die Darstellung gewählt werden soll. Wir geben zudem an, dass nach dem Aufrufen des Registers, jenes Element ausgewählt werden soll, das wir gerade angewählt haben (- Näheres zum ma-zutaten.odd siehe unter Abschnitt [b. Bearbeitung des ODD des Zutatenregisters](#)). Dafür müssen wir von dem <idno>-Element ausgehend zum übergeordneten <item>-Element, dessen @xml:id

auf die Zutat im Sachregister verweist.

The screenshot shows the TeiPublisher interface for defining a model block. The 'behaviour' field is set to 'block'. The 'Template' field contains the following XQuery code:

```
<div>Siehe weitere Schreibweisen dieser Zutat im \[register\]" target="_blank">Register</div>
```

- Nun muss abschließend noch die Darstellung der `<label>`-Elemente modifiziert werden. Damit diese nicht alle im Popover erscheinen, legen wir eine Modellspezifikation für alle `<label>`-Elemente, unabhängig von ihrem Attributwert an. Da wir alle Elementinhalte ausblenden wollen, wählen wir als behaviour "omit" aus.

The screenshot shows the TeiPublisher interface for defining a label element. The 'behaviour' field is set to 'omit'. The 'Template' field contains the following XQuery code:

```
<div>Code Vorlage für den Inhalt angeben</div>
```

Wenn wir uns jetzt erneut unser aktualisiertes Dokument ansehen und die Maus über eine Zutat bewegen, sieht der Inhalt des Popovers bereits anders aus.

Hienach volgenn Erlich erzney vonn mangerlay dingenn zu sammenn gesammet
 Als du danne hernach wol vindenn wurdest
 Ein purgatz Zu der zeÿtt des herbsbes furr allenn gebrechenn
 [N]ym hieffenn die zeÿtrig sein vnd mit dem reijff wol beczwungenn
 vnd njm die mit kernn vnd mit all vnd derre die in eynen ofenn vnd
 lege sie danne herauß vnd laß den wind dar durch genn vnd
 njm dann einen moerserr vnd stoß die hieffenn vnd misch die mit honig
 vnd das isse das des morgens nüchternn <https://www.wikidata.org/entity/Q10987>
Meßigkeit Ist ein aufenthaltung des ganzen le Siehe weitere Schreibweisen dieser Zutat im Register
 aller gelider . Vnd ein außderung aller boesenn duenst Wann meßigkeit ist
 ein zusoerung aller lasert
Wilu gur lekuchenn machenn R zu einem seÿdlein hoenigs 1/2 lor negelleyn
 1/2 lor muscat 1/2 lor Ingwert 1/2 lor pfeffers vnd stoß das zu sammenn vnd dar
 auf mach einen teig vnd zu **ein** moß honigs njm zwirt soul als vor be=zeichent stett

Während uns der obere Link direkt zur entsprechenden Wikidata-Entität führt, kommen wir über den Link

zum Register direkt an die entsprechende Stelle im Zutatenregister, wo die ausgewählte Zutat in die Mitte der Registeransicht rückt.

The screenshot shows a software interface with a list of ingredients (Zutaten) and their alternative spellings. The list includes:

- Zutat: **Holunderbeere**
Alternative Schreibweisen im Frühneuhochdeutschen:
pern
- Zutat: **Holzapfel**
Alternative Schreibweisen im Frühneuhochdeutschen:
holcz appfel, holcz ephel, holcz oepfelf, holcz oppfel, holzapfel, holczepfeln, holczepfeln
- Zutat: **Holzapfelsaft**
Alternative Schreibweisen im Frühneuhochdeutschen:
holcz eppfelf saff
- Zutat: **Honig**
Alternative Schreibweisen im Frühneuhochdeutschen:
henig, hoenigs, hong, hongs, hongy, honic, honich, honicht, honick, honig, honigs
- Zutat: **Huhn**
Alternative Schreibweisen im Frühneuhochdeutschen:
hen, hener, henern, henn, henndlein, henne, hennen, hentr, hunetr, pulli
- Zutat: **Hühnerei**
Alternative Schreibweisen im Frühneuhochdeutschen:
aren, eyernn, eyerr, eyes, eyg, eye, eyger, henn ay
- Zutat: **Ingwer**
Alternative Schreibweisen im Frühneuhochdeutschen:
ingwer, Ingwert, ingwert, ingwerts, ingwers, ingwerß, inigber, innber, inngber, inngewer, inngper, inngwer, innver, innwear, innwer, innwers

- In der XML-Datei unseres Projektes haben wir nun außerdem noch Auszeichnungen für Streichungen und Hinzufügungen durch die/den ursprüngliche:n Schreiber:in in Form von `` und `<add>`, Abkürzungen und deren Langform im `<choice>`-Element, Hinzufügungen durch Editor:innen oder Transkribierende als `<supplied>` und Fußnoten als `<note>`. Wie bereits erwähnt, wurden beim Erstellen des projekteigenen ODD bereits einige grundlegende Modellspezifikationen des TEI-Publisher-Base-ODD übernommen, die häufig in digitalen Editionen vorkommen. Dementsprechend gibt es auch für alle genannten Elemente bereits Modelle. Während einige dieser Standard-Darstellungen ganz gut für unser Projekt passen, möchten wir andere jedoch noch weiter anpassen.
- Starten wir also der Reihe nach zuerst mit der Modellspezifikation für Streichungen bzw. das ``-Element. Diese sieht standardmäßig als Rendition einen durchgestrichenen Text vor. Da dieses Element laut **Auszeichnungsrichtlinien für Manuskripte im DTA-Basisformat** (dessen Schema wir im Zuge der **Annotation in ediarum** verwendet haben) jedoch verschiedene Werte im `@rendition`-Attribut haben kann, wollen wir im Modell für das ``-Element etwas spezifischer sein und fügen dort hinzu, dass das Rendering als durchgestrichener Text nur für ``-Elemente mit `@rendition="#s"` gelten soll.

The screenshot shows the TEI-Publisher model editor for the `del` element. The configuration includes:

- model [inline]**: `@rendition="#s"`
- Predicate**: `1 @rendition="#s"`
- behaviour**: `inline` (selected), `oder` (Selbstdefiniertes Bi)
- CSS Class**: `[CSS Klasse (zur Verwendung in externem CSS)]`
- Template**: `<> <...> <X> [...]`
- Code Vorlage für den Inhalt angeben**: `1`
- Parameters**: `+`
- Renditions**: `+`
- Rendition**: `Scope`: `1 text-decoration: line-through;`
- Use source rendition**:

In der Publikationsansicht des *TEI Publisher* sehen ~~~~-Elemente entsprechend so aus:

Wiltu einen guten stockfisch machenn So laß in siedenn als ein kalb
fleisch vnd laß in ein siedenn auff halben wegk vnd seig dann die
brue dar vonn vnd schuet den stockvisch herauß vnd erclaub in
wol vnd nÿm dann ein pfannen vnd thue smalz dar ein vnd laß
es warmm werdenn vnd leg dann den stockvisch in die putternn
vnd laß sie dar ýnnen erhitzenn vnd nÿm dann Ingwert vnd saffran
mit des visch brue vnd ferb das do mit ein vnd geuß es **veb**
den **stock in die visch** in die pfannen vnd laß es ein wal oder
drei thun so ist er gerecht

Item Hienach volgt wie man ein ganß pratenn soll So laß sie
vor zwenn oder drei tag wol hungernn das die bösen predenn

- Als nächstes widmen wir uns im ODD-Editor der Spezifikationen des **<add>**-Elements. Auch dort nehmen wir mittels Änderung im Predicate-Feld eine Limitation vor, indem wir das bestehende Modell auf das Attribut **@place="superlinear"**-Element beschränken. Außerdem adaptieren wir die Darstellung mit dem entsprechenden CSS Code, sodass diese nachträglich eingefügten Textstellen hochgestellt und in einem Braunton erscheinen.

The screenshot shows the 'ODD Editor' interface for the 'model [inline]' section. In the 'Predicate' field, the '@place="superlinear"' attribute is highlighted with a red box. In the 'Renditions' section, a CSS rule 'color: #bd804a; position: relative; top: -0.5em; font-size: 80%;' is entered and also highlighted with a red box. This configuration ensures that the added text will be displayed in a specific color and font size in the final publication.

Nach dem Aktualisieren in der Publikationsansicht, wird das **<add>**-Element entsprechend unserer Änderungen dargestellt.

Die kichernn Hitzen vnd plewend vnd machenn pruntzen vnd machenn
den frawenn ire Recht zu vier wochenn einß komenn als es sein soll
vnd spricht **Aurois** das sie prechenn den stein der do lejt in den
lendenn oder der do leit in der blosenn vnd auch die brüe mit
der sie gesotenn werdenn vnd darczu sein die swartzen kichernn pesserr
dann die weissenn Vnd **Galien** spricht das bru von kichernn so
man sie seudt die ist der lebernn gut wann es reinigt sie vnd die
nÿren do der stein wechszt den schleim vnd treibt vil vnfplatz
vonn dem menschenn
Arbeißenn Brüe hat die selben krafft aber nit als krefftiglichenn
do vonn so man kichernn nit hat so seudt man arbeiß kichernn
oder arbeiß die grun sein so soll man **nt** essenn wann sie machenn
em menschenn boese feuchtigkeit

- Für **<choice>**-Elemente gibt es ebenfalls bereits Voreinstellungen, die vorsehen, dass die Langform ausgegeben und die Abkürzung in einem Popover angezeigt wird. Wir möchten dies für unser Projekt jedoch umkehren und wie im Originaltext nur die Abkürzung anzeigen und diese farblich markieren, um Nutzer:innen unserer Edition anzuzeigen, dass diese Textstelle besonders ist und somit eine Erkundung der

Stelle mit der Maus anzuregen. Im Popover wird schließlich die Langform angezeigt.

The screenshot shows the TeiPublisher configuration interface for a 'choice' element. It includes sections for 'model [alternate]' (sic and corr), 'Output' (Description: Beschreibung des Modells), 'Predicate' (1 abbr and expan), 'behaviour' (alternate oder Selbstdefiniertes Bi), 'CSS Class' (CSS Klasse (zur Verwendung in externem CSS)), 'Template' (<div>
 <x> [</>]), and 'Parameters' (default: abbr[1], alternate: expan[1]). A red box highlights the 'Parameters' section. Below it, under 'Renditions' (Scope: 1 color: #bd804a), another red box highlights the color setting.

Zurück in der Publikationsansicht können wir nun unsere Änderungen für das `<choice>`-Element (b) sehen. Außerdem ist in diesem Screenshot auch die unveränderte Darstellung für das `<supplied>`-Element (a) ersichtlich, bei der die editorische Hinzufügung mittels eckigen Klammern um den Zusatz gekennzeichnet wird. Da wir diese Darstellung passend finden, nehmen wir hier keine Änderungen vor.

Hienach volgenn Etlich ertzney vonn mangerlay dingenn zu sammenn gesammet
Als du danne hernach wol vindenn wurdest
Ein purgatz Zu der zejtt des herbstes furr allenn gebrechenn
a [N]ym hieffenn die zejttig sein vnd mit dem reijff wol beczwungenn
vnd nym die mit kernn vnd mit all vnd derre die in eyнем ofenn vnd
lege sie danne herauß vnd laß den windt dar durch genn vnd
nym dann einen moerserr vnd stoß die hieffenn vnd misch die mit honig
vnd das isse das des morgens nüchternn
Meßigkeit Ist ein aufenthaltung des ganzen leichnams . Vnd ein starck
aller gelider . Vnd ein außerung aller boesenn duenst Wann meßigkeit ist
ein zustoerung aller lasterr b
Wiltu gut leukuchen nachenn R zu einem sjydelein hoenigs 1/2 lot negelcyn
1/2 lot muscat 1/2 lot Ingwe pfeffers vnd stoß das zu sammenn vnd dar
auß mach einen teig vnd zu em now honige nym zwirt souil als vor be=
zeichent stett

- Zuletzt haben wir noch die `<note>`-Elemente für Fußnoten, für die wir ebenfalls die bereits festgelegten Modellspezifikationen ohne Anpassungen übernehmen.

Wiltu machen ein gestroczes gepachens So mach ein teÿck von
eytell eyernn vnd wurtz in wol vnd mach in gel vnd warmm [du] gutt
holmm in den teick das sie naß werdenn vnd nym sie dann
her auß vnd pack es in einem schmaltz vnd vsaltz nit
Wiltu machen gut kuchenn vonn ejernn So nym ejer wie vil
du wilt vnd zu slach die wol vnd schneid semel funf lot dar
vnt vnd thue dar ein weinpert vnd schmaltz in ein pfannen des
genug sej vnd geuß die ejer dar ein vnd laß es packenn jnnenn
vnd aussenn do mit slach es auff ein panck vnd hack dar vnt
gut wurtz vnd schneid es zu scheubenn vnd rict es an
Item Ein essenn von milch kuchenn So soltu sie clein schnëy

- * Lesart nach Feyl: durzend
- * Rezept endet hier

- Für die `<div>`-Elemente, die jeweils den ausgezeichneten Text der einzelnen Manuskriptseiten beinhalten, nutzen wir nun erneut eine Modellsequenz. Von diesen Elementen ausgehend möchten wir nämlich jeweils auf die `<pb>`-Elemente zugreifen, um einerseits die Webkomponente zu erstellen, die die Faksimiles

verlinkt, sowie um die Seitenzahl anzuzeigen, und um andererseits auch den Inhalt des `<div>`-Elements anzuzeigen. Wir erstellen dementsprechend eine Sequenz für das `<div>`-Element und beschränken dieses mit dem Predicate `preceding-sibling::pb` auf jene `<div>`-Elemente, denen ein `<pb>`-Element auf gleicher Ebene vorangeht. In der Modellsequenz erstellen wir anschließend 2 Modelle:

- Zuerst benötigen wir ein Modell mit dem behaviour "block", das in der Ausgabe des `<div>`-Elements auf das vorhergehende `<pb>`-Element zugreift und anschließend nur die letzten 4 Zeichen ausgibt, da diese die Seitenzahl sowie die zusätzliche Information enthält, ob es sich bei der Manuskriptseite um eine Vorder- oder Rückseite handelt. Um die Seitenzahl legen wir außerdem eckige Klammern, um anzuzeigen, dass dies ein editorischer Zusatz ist und kein Teil der Transkription. Für den content-Parameter fügen wir dabei folgenden XPath-Ausdruck ein:

```
concat("[", substring(preceding-sibling::pb[1]/@n, string-length(preceding-sibling::pb[1]/@n)-3), ")"])
```

Außerdem geben wir hinsichtlich der Rendition an, dass die Seitenangabe grau und zentriert ausgegeben werden soll.

The screenshot shows the TEI Publisher interface with two model definitions. The first model, under the heading 'preceding-sibling::pb', has a 'Predicate' field containing 'preceding-sibling::pb'. The second model, under the heading 'block', has a 'behaviour' field set to 'block'. Both models have a 'content' parameter with the value `concat("[", substring(preceding-sibling::pb[1]/@n, string-length(preceding-sibling::pb[1]/@n)-3), ")"])`. The 'Renditions' section for the 'block' model contains the CSS rule `color: gray; text-align: center;`.

- Das zweite Modell soll das Einbinden der Faksimiles regeln. Diese Modellspezifikation verschieben wir nach dem Anlegen nun an die letzte Stelle in der Sequenz, und nutzen einen XPath-Ausdruck im Predicate, um anzugeben, dass diese Spezifikation nur für jene `<div>`-Elemente gelten soll, denen ein `<pb>`-Element mit einem `@fac`-Attribut vorausgeht. Als behaviour wählen wir in diesem Fall "webcomponent" aus und setzen für diese Komponente die entsprechenden Parameter. Das Verbindungselement zwischen dem Faksimile und dem Text trägt den Namen "pb-facs-link" und enthält die Attribute `@emit` mit dem Wert "transcription", das für den Kanal steht, über den kommuniziert wird. Der Pfad zum Faksimile, auf das verwiesen wird, befindet sich letztlich im `@fac`-Attribut des `<pb>`-Elements, wobei wir über den Wert im `@fac` zur `@xml:id` in dem dazugehörigen `<graphic>`-Element gelangen und dort die `@url` wählen, die die URI zu unseren Faksimiles enthält.

The screenshot shows the configuration interface for a model named 'model [webcomponent]'. The 'preceding-sibling::*[1]@facsc' predicate is defined. The behaviour is set to 'webcomponent' or 'oder'. A CSS class '[CSS Klasse (zur Verwendung in externem CSS)]' is specified. The template contains code for generating a link: '<x> <_> <x> [<_>]' with the code '1 [Code Vorlage für den Inhalt angeben]'. Parameters include 'name' (value: 'pb-facs-link'), 'emit' (value: 'transcription'), and 'facsc' (value: 'preceding-sibling::*[1]/id(substring-after(@facsc, '#'), root().)/@url'). A checkbox 'Use source rendition' is checked.

→ Damit die Faksimiles tatsächlich angezeigt werden, bedarf es noch weiterer Schritte, die im Abschnitt [c.\) Bearbeitung des Page Templates](#) näher ausgeführt sind.

Mit all diesen Adaptionen haben wir aber jedenfalls in der Publikationsansicht eine zusätzliche Zeile vor Textbeginn erzeugt.

The screenshot shows a page from 'MS Augsburg, Universitätsbibl., Cod.III.1.2*43 (A1)'. A red box highlights the header '[059r]'. The main text reads:

Hienach volgt vonn dem kochenn vnd hat gemacht meyster Eberhart Ein koch
herczog heinrichs zu lands hut Zum Ein salsenn von weichselnn zu machen
Item Wiltu machen ein gute salsenn von weichselnn So ? thue die weich=
sell in einen hafen vnd setz die auff ein glut vnd laß sie siedenn vnd
laß dann wider erkaltenn vnd streich sie durch ein tuch vnd thue sie
dann wider in den hafenn vnd setz sie auff ein glut vnd laß sie wol sieden
vnd rurr sie piß sie dick wirt vnd thue dann honig dar an vnd geribens

- In dem zweiten Manuskript unseres Beispielprojekts gibt es jetzt bis auf das `<fw>`-Element keine zusätzlichen Elemente mehr, die einer Bearbeitung bedürfen. Wie legen also eine neue Modellspezifikation für `<fw>`-Elemente mit dem Attribut `@type="header"` an, wählen als behaviour "block" und definieren anschließend die Rendition mit CSS-Code, der dafür sorgen soll, dass diese Textstelle etwas vergrößert, fettgedruckt und zentriert ausgegeben wird.

fw MODE:CHANGE

model [block]
@type="header"

Output Mode
[Modus setzen für nachfolgende Modelle]

Description
[Beschreibung des Modells]

Predicate
1 @type="header"

behaviour
block oder [Selbstdefiniertes Bi]

CSS Class
[CSS Klasse (zur Verwendung in externem CSS)]

Template
<|> <..> <X> [[..]]
1 [Code Vorlage für den Inhalt angeben]

Parameters +

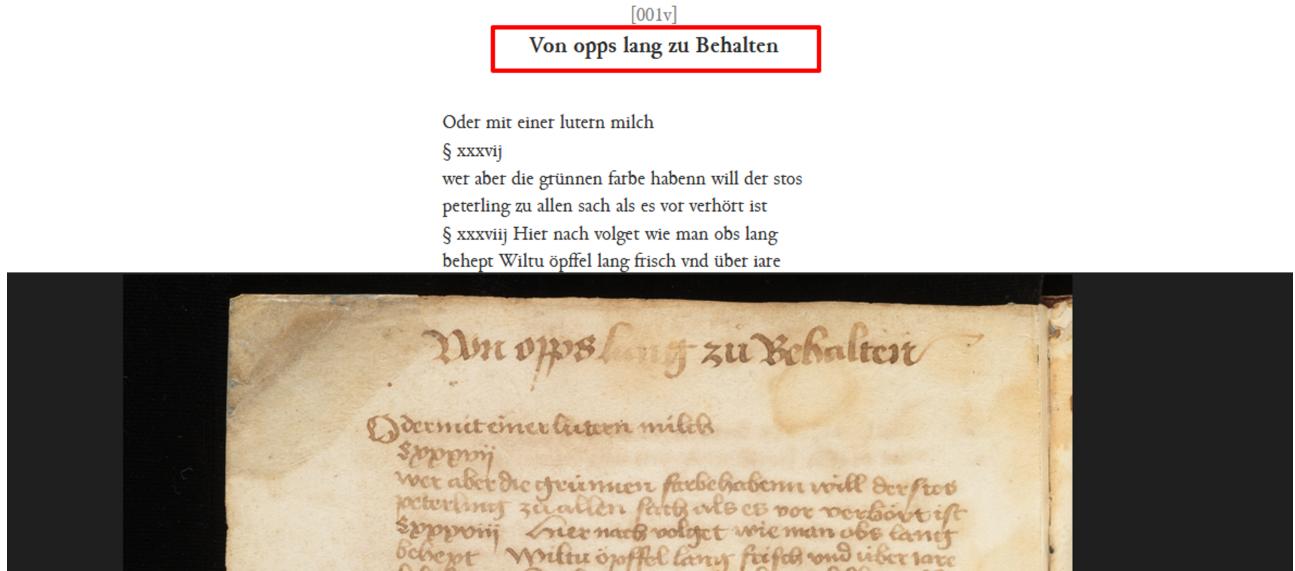
Name	Parameter	1	<input type="checkbox"/> setze	
------	-----------	---	--------------------------------	--

Renditions +

Scope	1 font-size: 1.5rem; font-weight: bold; text-align: center;	<input type="checkbox"/> Use source rendition	
-------	---	---	--

In der Publikationsansicht sehen wir nach dem Aktualisieren schließlich das Ergebnis unserer Änderung.

MS Solothurn, Zentralbibliothek, Cod. S 490 (So1)



b. Bearbeitung des ODD des Zutatenregisters

In unserem Projekt gibt es aber nicht nur die XML-Dateien für das Manuskript, sondern noch eine weitere XML-Datei, die das Zutatenregister beinhaltet. Da wir für dieses Register eine andere Publikationsansicht haben möchten, müssen wir hierfür ein eigenes ODD erstellen.

- Zuerst laden wir das Register ([Sachbegriffe.xml](#)) in der Spielwiese hoch, wo wir auch unsere Manuskript-Dateien hochgeladen haben. Nach erfolgreichem Upload können wir sie in der Dokumentenliste der

Spielwiese einsehen. Danach legen wir uns unter ODD Dateien (am Seitenende) eine ODD-Datei mit dem Dokumentnamen ma-zutaten und der Bezeichnung "MA Zutaten" an.

The screenshot shows the 'The Instant Publishing Toolbox' interface. On the left, there's a search bar and a sidebar with language selection ('deu'). The main area displays two entries under 'ÜBERGEORDNETES VERZEICHNIS': 'MS Augsburg, Universitätsbibl., Cod.III.1.2°43 (A1)' and 'MS Solothurn, Zentralbibliothek, Cod. S 490 (So1)'. Below these are sections for 'Sachbegriffe' and 'Hochladen'. A red box highlights the 'Hochladen' section, which contains a 'HOCHLADEN' button and instructions: 'Dateien vom Typ .xml, .tei, .odd, .docx, .md, .mei hier hinziehen zum Hochladen.' To the right, a red box highlights the 'ODD Dateien' section, which contains a button 'ALLE ODDS NEU GENERIEREN (LANGSAM)' and a link 'MA Rezepte'. A red arrow points from the 'Hochladen' section towards the 'ODD Dateien' section.

→ Möglicherweise muss die Seite aktualisiert werden, damit alle neu erstellten oder hochgeladenen Dokumente sichtbar sind.

- Wenn wir nun unser Register öffnen, sehen wir in der Publikationsansicht mit dem Standard-Template, dass unser Register als eine Liste mit allen Informationen zu den einzelnen Zutaten ausgegeben wird. Denn das Zutatenregister besteht aus einem `<list>`-Element mit vielen `<item>`-Elementen für die einzelnen Zutaten, und in dem Standard-Template sind nur für `<list>`- sowie `<item>`-Elemente Modellspezifikationen vordefiniert. Alle weiteren Elemente, für die es keine Regeln gibt, werden in Textform ausgegeben.

The screenshot shows the publication view with the 'MA-Zutaten' template selected. The right sidebar includes options for 'Seitenansicht', 'Template Auswählen', and 'Achtung'. The main content area displays a list of ingredients extracted from Wikidata entities, such as 'Arbeitsmehl', 'Walderdbeere', and various types of cheese ('Ziegenmilch', 'Gewürz gestupp').

Wir verbinden unser neu erstelltes ODD mit unserem Register und belassen bei der Auswahl der HTML-Templates die Verknüpfung mit dem Standard-Template (view.html).

- Danach wechseln wir in den ODD-Editor zur Anpassung des neuen ODD und beginnen damit, dem Register eine Überschrift zu geben. Dafür öffnen wir über das linke Seitenmenü das `<list>`-Element, wo es bereits ein vordefiniertes Modell mit dem behaviour "list" gibt. Da wir über diese Liste nun die Überschrift setzen wollen und zusätzlich die Listenelemente anzeigen wollen, benötigen wir wieder eine

Modellsequenz. Diese Modellsequenz soll nun einerseits ein Modell für die Überschrift und eines für die Ausgabe der Listenelemente beinhalten, weshalb wir 2 Modelle erstellen.

The screenshot shows the 'MA-Zutaten' model sequence editor. At the top, there's a toolbar with icons for file operations. Below it is a sidebar with 'Element Spezifikationen' containing fields for 'idno', 'item', 'label', and 'list'. The main area shows a tree structure under 'list MODE CHANGE'. It contains a 'modelSequence' node, which has two children: 'model [heading]' and 'model [list]'. The 'model [list]' node is highlighted with a red box. To the right of the tree are standard drag-and-drop icons for reordering.

- Das erste der beiden Modelle in der Sequenz bekommt das behaviour "heading" und der Inhalt der Überschrift soll "Zutatenregister" in Großbuchstaben sein, was wir dementsprechend im content-Parameter festhalten. Außerdem geben wir für die Rendition an, dass die Überschrift fettgedruckt und mittig ausgegeben werden soll.

This screenshot shows the configuration dialog for the 'model [heading]' node. It includes sections for 'Output', 'Description', 'Predicate', 'behaviour' (set to 'heading'), 'Parameters' (with a parameter 'content' set to 'ZUTATENREGISTER'), and 'Renditions' (with a scope setting 'font-weight: bold; text-align: center;'). A red box highlights the 'content' parameter entry. Another red box highlights the 'font-weight: bold; text-align: center;' entry in the 'Renditions' section.

- Damit nun auch der Inhalt der Liste und nicht nur die Überschrift angezeigt wird, setzen wir beim zweiten Modell der Sequenz das behaviour auf "list". Im content-Parameter nutzen wir diesmal einen XQuery-Ausdruck, der die `<item>`-Elemente anhand des ersten `<label>`-Elements mit dem Attribut `@type="reg"` alphabetisch sortieren soll. (Für diese Sortierung haben wir im Übrigen den

Slack-Channel genutzt, wo uns sehr schnell mit dem Code weitergeholfen wurde!

The screenshot shows the configuration for a 'list' model. The 'behaviour' field is set to 'list' (highlighted with a red box). In the 'Parameters' section, there is a parameter named 'content' with the value 'for \$i in item let \$label := \$i/label[@type="reg"] order by \$label return \$i'. The 'Renditions' section contains a single entry with a scope of '1' and a value of 'padding-bottom: 1rem;'. A red box highlights the 'Parameters' and 'Renditions' sections.

- Das Modell für die `<item>`-Elemente ändern wir auch nur sehr geringfügig. Diesem Element ist standardmäßig das behaviour "listItem" zugeordnet, das wir genau so belassen wollen. Wir möchten nur den Abstand zwischen den Listenelementen etwas vergrößern, weshalb wir die Rendition entsprechend anpassen.

The screenshot shows the configuration for an 'item' model. The 'behaviour' field is set to 'listItem' (highlighted with a red box). In the 'Parameters' section, there is a parameter named 'content' with the value '1'. The 'Renditions' section contains a single entry with a scope of '1' and a value of 'padding-bottom: 1rem;'. A red box highlights the 'Parameters' and 'Renditions' sections.

- Als nächstes widmen wir uns den Kindelementen des `<item>`-Elements und beginnen mit dem `<kidno>`-Element. Dieses wollen wir gar nicht ausgeben bzw. die Information später nur anderweitig im Hintergrund für einen Link verwenden. Insofern müssen wir für das `<kidno>`-Element nur eine Modellspezifikation anlegen, die mit dem behaviour "omit" attribuiert wird. Zusätzlich wird dieses Modell noch auf jene

Elemente eingeschränkt, die das Attribut `@type="uri"` besitzen.

The screenshot shows the TeiPublisher interface with the title 'idno' and 'MODE: ADD'. A red box highlights the first item in the sequence: 'model [omit] @type="uri"'. Another red box highlights the 'Predicate' section below it, which contains '1 @type="uri"'. A third red box highlights the 'behaviour' section, which shows 'omit' selected. The interface also includes a 'Description' field and a 'Mode' dropdown.

- Die nachfolgenden Kindelement im `<item>`-Element sind die `<label>`-Elemente mit einer regulären Schreibweise im modernen Standarddeutsch, die den Attributwert "reg" führen, und allen weiteren frühneuhochdeutschen Varianten, die mit dem Attributwert "alt" ausgezeichnet sind. Insgesamt haben wir 4 Vorhaben, die jeweils unterschiedliche behaviour verlangen. Daher benötigen wir wieder eine Modellsequenz mit 4 Modellen.

The screenshot shows the TeiPublisher interface with the title 'label' and 'MODE: ADD'. A red box highlights the first item in the sequence: 'modelSequence'. Below it, four individual models are listed, each with a red box highlighting the 'model' label and its type: 'model [inline] @type="reg"', 'model [link] @type="reg"', 'model [block] @type="reg"', and 'model [inline] @type="alt"'. The interface includes standard navigation buttons like down, up, delete, and add.

Gehen wir diese Modelle und erwünschten Verhaltensweise einzeln durch:

- Unser erstes Modell ist für `<label>`-Elemente mit dem Attributwert "reg" bestimmt und soll vor den tatsächlichen Inhalt des Elements das fettgedruckte Wort "Zutat" stellen. Dementsprechend sieht

diese Modellspezifikation aus:

The screenshot shows the 'model [inline]' configuration for a 'reg' element. Key fields highlighted with red boxes are:

- Predicate:** 1 @type="reg"
- behaviour:** inline oder Selbstdefiniertes Bi
- Parameters +**: A table with a row for 'content' where Name is 'content' and Parameter is '1 "Zutat: "'.
- Renditions +**: A table with a row for 'Scope' where Rendition is '1 font-weight: bold;'.

- Das zweite Modell bezieht sich ebenfalls auf `<label type="reg">`-Elemente. Diesmal möchten wir jedoch den Inhalt sehr wohl ausgeben, diesen aber mit einem Link hinterlegen. Daher wählen wir nun als behaviour "link" und verweisen dann im uri-Parameter auf das vorhergehende `<idno>`-Element, welches den Link zur Wikidata-Entität enthält. Zudem verändern wir wieder typographische Merkmale mit entsprechendem CSS-Code.

The screenshot shows the 'model [link]' configuration for a 'reg' element. Key fields highlighted with red boxes are:

- Predicate:** 1 @type="reg"
- behaviour:** link oder Selbstdefiniertes Bi
- Parameters +**: A table with a row for 'uri' where Name is 'uri' and Parameter is '1 preceding-sibling::idno'.
- Renditions +**: A table with a row for 'Scope' where Rendition is '1 font-weight: bold;'.

- Als nächstes möchten wir eine informative Zeile einbauen, die angibt, dass es sich nachfolgend (bei den `<label type="alt">`-Elementen) um alternative Schreibweisen im Frühneuhochdeutschen handelt. Erneut erstellen wir ein Modell für das vorhergehende `<label type="reg">`-Element, da

wir andernfalls diesen Informationstext vor jeder Schreibvariante stehen hätten. Die Modellspezifikation enthält als behaviour den Wert "block" und im content-Parameter die gewünschte Zusatzinformation. Mittels entsprechenden CSS-Ausführungen im Abschnitt Rendition legen wir fest, dass dieser Text unterstrichen werden soll.

model [block]
@type='reg'

Description
[Beschreibung des Modells]

Predicate
1 @type="reg"

behaviour
block oder [Selbstdefiniertes Bi]

CSS Class
[CSS Klasse (zur Verwendung in externem CSS)]

Template
<|> <...> <X> [...]
1 [Code Vorlage für den Inhalt angeben]

Parameters +

Name	content	X	Parameter 1 "Alternative Schreibweisen im Frühneuhochdeutschen: "	<input type="checkbox"/> setze	█
------	---------	---	--	--------------------------------	------------------------------------

Renditions +

Scope	1	Rendition text-decoration: underline;	<input checked="" type="checkbox"/> Use source rendition	█
-------	---	--	--	------------------------------------

- Zu guter Letzt passen wir nun noch die Modellspezifikation jener `<label>`-Elemente mit dem Attributwert "alt" an. In der standardmäßigen Ausgabe werden die Inhalte dieser Elemente nacheinander als inline-Elemente ausgegeben. Hier möchten wir noch Beistriche zwischen die einzelnen Varianten setzen, wobei dem letzten `<label>`-Element in dieser Sequenz kein Beistrich mehr nachgestellt werden soll. Dies machen wir mit folgendem XPath im content-Parameter:

```
concat(.,  
      if (following-sibling::label[1][@type='alt'])  
        then ','  
        else '')
```

Die 4. Modellspezifikation in dieser Sequenz sieht dementsprechend folgendermaßen aus:

The screenshot shows the TEI Publisher interface for editing a model. The 'model [inline]' section is open, revealing the following configuration:

- Predicate:** 1 @type="alt"
- behaviour:** inline
- Template:** <|> <..> <X> [[...]]
1 [Code Vorlage für den Inhalt angeben]
- Parameters:** content

A red box highlights the 'Parameters' section, specifically the parameter definition:

```
1 concat(.,  
2     if (following-sibling::label[1][@type='alt'])  
3     then ','  
4     else '')
```

The 'Renditions' section is also visible at the bottom.

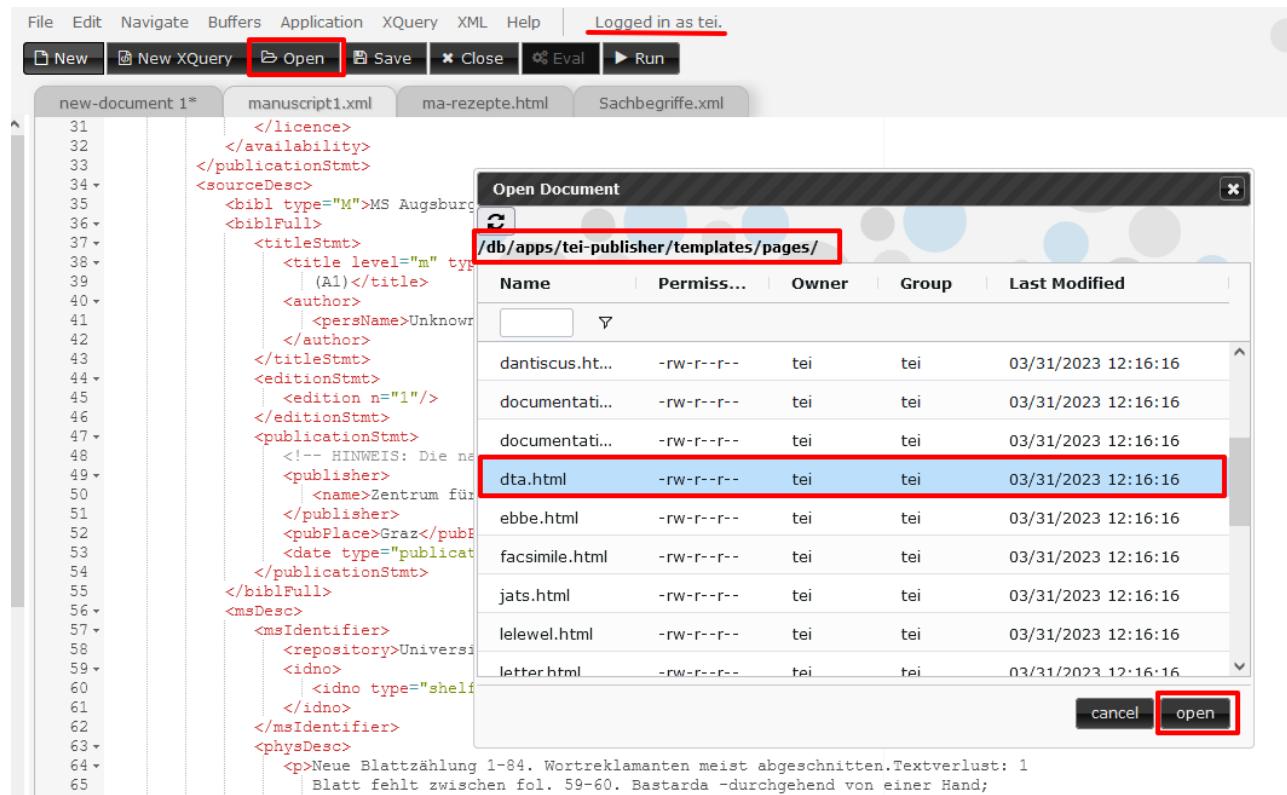
Nachdem wir all diese Anpassungen vorgenommen haben, wechseln wir in die Publikationsansicht, aktualisieren diese und haben nun eine unseren Vorstellungen angepasste Webansicht des Zutatenregisters.

The screenshot shows the published 'ZUTATENREGISTER' page. It lists various ingredients with their alternative spellings:

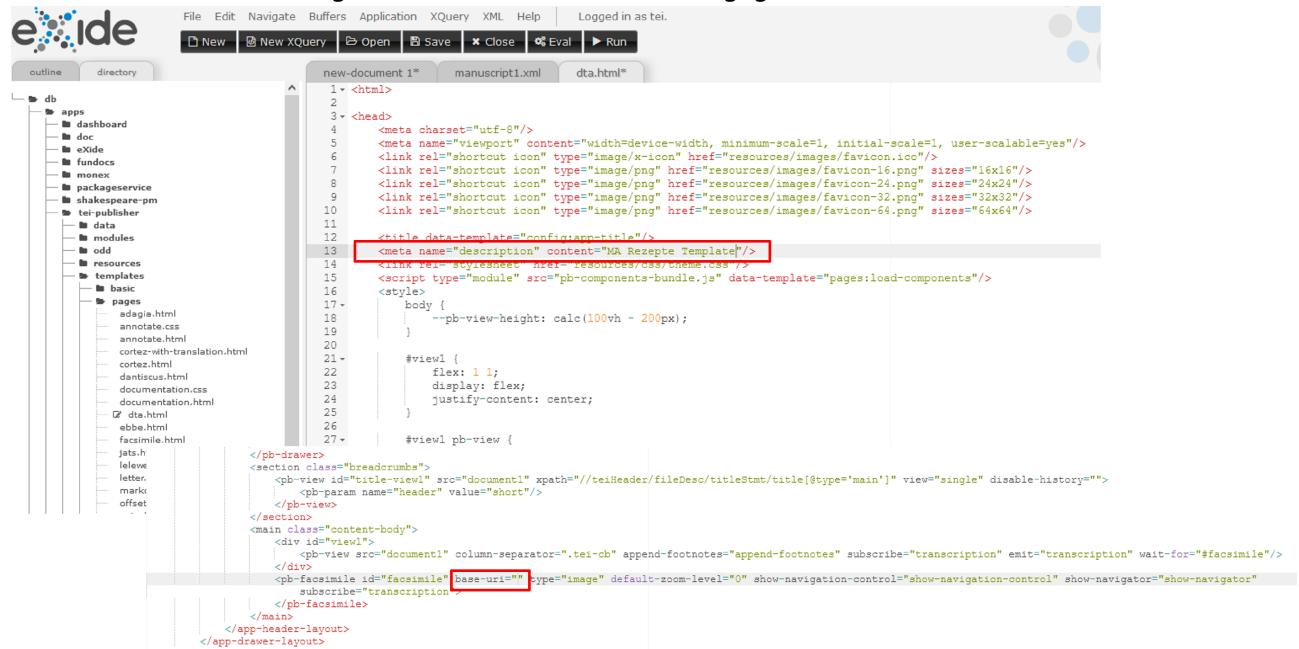
- Zutat: [Agraz](#)
Alternative Schreibweisen im Frühneuhochdeutschen:
agraef, agras, agraz, agres, agresf, agrest, agreste, agroße
- Zutat: [Apfel](#)
Alternative Schreibweisen im Frühneuhochdeutschen:
apfel, apfelen, apfell, appfel, apffel, apfl, aphel, aphell, aphell, aplh, appel, appfel, appfelen, appfell, appfelen, appfeln, appfelf, appffel, appfll, apphel, apphellen, apphels, mala
- Zutat: [Arznei](#)
Alternative Schreibweisen im Frühneuhochdeutschen:
arcenie, artzenie, artzny, ercznei, erczney, ertzney
- Zutat: [Bertram](#)
Alternative Schreibweisen im Frühneuhochdeutschen:
pertram, petram, pyterrum
- Zutat: [Bier](#)
Alternative Schreibweisen im Frühneuhochdeutschen:
pyer

c. Bearbeitung des Page Templates (+ weitere ODD-Anpassungen)

- Da wir in unserem Projekt über die Faksimiles unserer Manuskripte verfügen, möchten wir für die Publikation eine Bild-Text-Synopse erstellen. Dies lässt sich im *TEI Publisher* über die Webkomponente `<pb-facsimile>` realisieren. Der einfachste Weg, um dies umzusetzen, besteht darin, dass wir uns zuerst ein Template kopieren, das diese synoptische Ansicht bereits integriert hat. Da in der Demo-Sammlung des *TEI Publisher* die XML-Struktur des Projekts Immanuel Kant unseren importierten XML-Dokumenten sehr ähnlich ist und uns auch die Darstellung ganz gut gefällt, wollen wir dieses Template für uns nutzen. Dafür navigieren wir in eXide unter "Open" in der Ordnerstruktur zu dem Template des Deutschen Textarchivs (dta.html), indem wir folgenden Pfad nutzen: '/db/apps/tei-publisher/templates/pages/'



Wir ändern in diesem Template nur im `<meta>`-Element mit dem Attribut `@name="description"` den Wert des Attributs `@content` auf "MA Rezepte Template" und entfernen weiter unten im Dokument im `<pb-facsimile>`-Element den Wert aus dem Attribut `@base-uri`, da wir in den `<graphic>`-Elementen in unserem XML, bereits den gesamten Pfad zum Faksimile angegeben haben.



Wir speichern das HTML-Dokument anschließend unter "ma-rezepte.html" in der gleichen Ordnerstruktur ab.

→ Für Änderungen in eXide ist die Eingabe eines Passworts erforderlich, wobei hier ebenfalls die User "tei" oder "tei-demo" mit dem jeweiligen Passwort genutzt werden kann. In manchen Ordnern kann es auch erforderlich sein, sich als "admin" einzuloggen, da dieser User die umfangreichsten Schreibrechte hat.

The screenshot shows the eXide interface with the 'File' menu highlighted. A 'Save Document As ...' dialog is open over the main workspace. The workspace shows a file tree under 'db/apps' and a code editor with XML/HTML content. The dialog contains a table for saving the document, with the 'Name' field set to 'ma-rezepte.html' and the 'Save' button highlighted.

Zurück in der Manuscript-Ansicht wählen wir im Menü in der zweiten Navigationsleiste, wo wir bereits unser projekteigenes ODD verknüpft haben, nun zusätzlich auch unser eigenes Template aus.

The screenshot shows the tei-Publisher playground interface. A red arrow points from the 'MA Rezepte Template' dropdown in the sidebar to the 'C' icon in the top navigation bar. The main content area displays a text block with red annotations. The sidebar includes sections for Einstellungen, Template Auswählen (with 'MA Rezepte Template' selected), and Achtung.

Nachdem wir den Browser aktualisiert haben, gelangen wir schließlich zur Text-Bild-Synopse.

The screenshot shows the tei-Publisher Text-Bild-Synopse interface. The main content area displays the annotated text from the previous screen. A large black rectangular redaction box is present on the right side of the page.

Die Seite ist in zwei Bereiche aufgeteilt, wobei die rechte Hälfte bereits für unsere Faksimiles vorbereitet ist, die wir jedoch noch einbinden müssen.

- Zum Einbinden der Faksimiles unserer Manuskripte müssen wir im ODD eine Kommunikation zwischen den Webkomponenten herstellen. Wir gehen also zurück in den ODD-Editor, wo wir Ausgaberegeln für das `<div>`-Element spezifizieren wollen. Dafür werfen wir zuerst noch einen Blick auf das XML des Manuskripts. Dort befinden sich die Links zu den Faksimiles in den `<graphic>`-Elementen, und jedes `<pb>`-Element enthält eine Referenz.

```

98+   <facsimile>
99+     <graphic xml:id="f0001" url="http://fromthepage.com/image-service/33038493/full/full/0/default.jpg"/>
100+    <graphic xml:id="f0002" url="http://fromthepage.com/image-service/33038494/full/full/0/default.jpg"/>
101+    <graphic xml:id="f0003" url="http://fromthepage.com/image-service/33038495/full/full/0/default.jpg"/>
102+    <graphic xml:id="f0004" url="http://fromthepage.com/image-service/33038496/full/full/0/default.jpg"/>
103+    <graphic xml:id="f0005" url="http://fromthepage.com/image-service/33038497/full/full/0/default.jpg"/>
104+  </facsimile>
105+ <text>
106+   <div>
107+     <pb xml:id="F33038493" n="UB_Augsburg_Cod_III_1_2_43_002r" facs="#f0001"/>
108+     <div xml:id="O1F33038493b">
109+       <p xml:id="O1F33038493b0">
110+         <hi rendition="#red">Hiennach volgenn Etlich ertzney vonn
111+           mangerley dingenn za sammenn gesammet<lb/> Als du danne hernach wol vindenn
112+             wurdest<lb/> Ein purgatz Zu der zeytt des herbstes fur allenn gebrechenn<hi>
113+           <lb/>
114+           <supplied><supplied><term type="ingredient" key="#rose_hip">hieffenn</term> die
115+             zeystig sein vnd mit dem reyff wol beczwungenn<lb/> vnd nyn die mit <term type="ingredient" key="#pip">kernn</term> vnd mit all vnd derre die in eynem ofenn
116+             vnd<lb/> lege sie danne herauß vnd laß den wind dar durch gen vnd<lb/> nyn dann
117+               einer moerserr vnd stöß die <term type="ingredient" key="#rose_hip">hieffenn</term>
118+             vnd misch die mit <term type="ingredient" key="#honey">honig</term>
```

Die Modellspezifikation für die Webkomponente haben wir bereits im vorherigen Abschnitt während der Bearbeitung der Seitenangabe aus den `<pb>`-Elementen, die über das `<div>`-Element angesteuert werden, beschrieben. Zum besseren Verständnis möchten wir aber noch mal einen kurzen Überblick über die einzelnen Schritte geben: Im ODD-Editor haben wir für jene `<div>`-Elemente, denen ein `<pb>`-Element mit einem `@facss`-Attribut vorausgeht, eine Modellspezifikation angelegt. Als behaviour haben wir in diesem Fall "webcomponent" ausgewählt und für diese Komponente die entsprechenden Parameter gesetzt. Das Verbindungselement zwischen dem Faksimile und dem Text trägt den Namen "pb-facs-link" und enthält die Attribute `@emit` mit dem Wert "transcription", das für den Kanal steht, über den kommuniziert wird. Der Pfad zum Faksimile, auf das verwiesen wird, befindet sich letztlich im `@facss`-Attribut des `<pb>`-Elements vor dem `<div>` - wobei wir über den Wert im `@facss` zur `@xml:id` in der `<graphic>` gelangen und dort die `@url` wählen, die die URI zu unseren Faksimiles enthält.

model [webcomponent]
preceding-sibling::*[1]/@facss

Output Mode
Modus setzen für nachfolgende Modelle

Description [Beschreibung des Modells]

Predicate
1 preceding-sibling::*[1]/@facss

behaviour
webcomponent oder [Selbstdefiniertes B]

CSS Class
[CSS Klasse (zur Verwendung in externem CSS)]

Template
<|> <..> <X> [<..>]
1 [Code Vorlage für den Inhalt angeben]

Parameters +

Name name	Parameter	1 'pb-facs-link'	<input type="checkbox"/> setze
Name emit	Parameter	1 'transcription'	<input type="checkbox"/> setze
Name facss	Parameter	1 preceding-sibling::*[1]/id(substring-after(@facss, '#'), root(..))/@url	<input type="checkbox"/> setze

Nach erneutem Aktualisieren in der Dokumentansicht werden die entsprechenden Faksimiles nun zu jeder

Manuskriptseite in den OpenSeadragon-Viewer geladen.

MS Augsburg, Universitätsbibl., Cod.III.1.2*43 (A1)

Hienach volgenn Erlich erzneij vonn mangerlaj dingenn zu sammenn gesammet
Als du danne hernach wol vindenn wurdest
Ein purgaz Zu der zejtt des herbstes furr allenn gebrechenn
[N]lym hieffenn die zejttig sein vnd mit dem reijff wol beczwungenn
vnd nym die mit kern vnd mit all vnd derre die in eynem ofenn vnd
lege sie danne heraus vnd laß den windt dar durch genn vnd
nym dann einen moerserr vnd stoß die hieffenn vnd misch die mit honig
vnd das isse das des morgens nüchternn
Meßigkeit Ist ein aufenthaltung des ganztzen leichnams . Vnd ein starck
aller gelider . Vnd ein außderung aller boesenn duenst Wann meßigkeit ist
ein zustoerung aller lasterr
Wiltu gut lekuchenn machenn Recipe zu einem seydelein hoenigs 1/2 lot negelllyn
1/2 lot muscar 1/2 lot Ingwerr 1/2 lot pfeffers vnd stoß das zu sammenn vnd dar
auß mach einen teig vnd zu einer moß honigs nym zwirt souil als vor be=
zeichent stett
Wiltu ein gute salsen machen in der fastenn Nym merrech vnd mandel
kern vnd stoß die in einem moerserr vnd thue wein dar zu das ist gut
Die vier kunigreich der werle
Orientalis [que est] Calida et humida
Meridionalis [que est] Calida et sicca
Occidentalis que est frigida et humida
Septembtrialis [que est] Frigida et sicca

- Eine weitere Webkomponente, die uns die Seitennavigation im Manuskript erleichtert, ist **<pb-navigation>**. Auch hier nutzen wir wieder die Beispielprojekte der Demo-Sammlung, in der wir gleich im ersten Beispiel (Adagia in Latine and English) die entsprechenden Navigationsbuttons finden. Wenn wir in der Seitenleiste nun überprüfen, welches Template eingebunden wurde, können wir sehen, dass dieses Beispielprojekt das **Default single text layout** nutzt.

Start Dokumentation Neuigkeiten DTS Herunterladen Administration Suchen nach

Adagia in Latine and English containing five hundred proverbs : very profitable for the vse of those who aspire to further perfection in the Latine tonge.

LECTORI. S.

CVm antē annos non ita mult * s, ludimodatorem Brechinensem agerem; placuit haec meis discipulis (nunquam satis, in hac thesi laudati viri Chiliadibus) tera & maxime polita Adagia seligere: Semper fuit ic Londini, ex multis meis Vigilijs & lucubrationibus, doctis & pijs iampridem innotunt; * eo, Principi r Christianae inseruire. Mihi nihil minus cogitanti, sumpatriota, (nomen lubentor subticco) farraginem ha exhibuit. Foetum agno * i, et vt paulo ornatius loqueretur Anglicè volui. Si quid pecc * ti arte [Page @tcp:27691:2] [Page @tcp:27691:3]

—Carpore vel noli nostra vel ede tua,
Sivoles — his vtore mecum.

Achtung
Die Optionen hier erlauben es, mit verschiedenen Seitenvorlagen und ODDs zu experimentieren. Bitte beachten Sie, dass nicht jede Kombination für jedes Dokument Sinn macht. Zum Beispiel ist eine Karte nur sinnvoll, wenn Koordinaten vorhanden sind. Ebenso erfordert eine seitenweise Ansicht, dass Seitennummern ausgezeichnet wurden. Die Fehlermeldung 'the server did not return any content' weißt in der Regel auf solche Unstimmigkeiten zwischen Markup, ODD und Seitenvorlage hin.

Wir öffnen daher in eXide das entsprechende Template (view.html) und können erneut per "Copy-Paste" die gewünschte Webkomponenten - in diesem Fall die jeweiligen **<pb-navigation>**-Komponenten des **<main>**-Abschnitts vor und nach der **<pb-view>**-Webkomponente - auswählen und in unser projektspezifisches Template an entsprechender Stelle vor und nach den anderen Webkomponenten des **<main>**-Abschnitts einbauen. Dadurch ersparen wir uns die Suche in der [Dokumentation](#) nach entsprechenden Komponenten bzw. müssen die Webkomponenten nicht selbst anlegen.

```

new-document 1* view.html ma-rezepte.html
71 </pb-load>
72 </ph-drawer>
73 <section class="breadcrumbs">
74   <pb-view id="title-view1" src="document1" xpath="//teiHeader/fileDesc/titleStmt/title[1]" view="single">
75     <pb-param name="header" value="short"/>
76   </pb-view>
77 </section>
78 <main class="content-body">
79   <pb-navigation direction="backward" unit="page" emit="transcription" subscribe="transcription">
80     <paper-fab icon="icons:chevron-left"/>
81   </pb-navigation>
82   <pb-view id="view1" src="document1" column-separator=".tei-cb" append-footnotes="append-footnotes" subscribe="transcription" emit="transcription" wait-for="#facsimile"/>
83   <pb-navigation direction="forward" unit="page" emit="transcription" subscribe="transcription">
84     <paper-fab icon="icons:chevron-right"/>
85   </pb-navigation>
86 </main>
87 </app-header-layout>
88 </app-drawer-layout>
89 </>
90 </>
91 </>
92 </>

```



```

new-document 1* view.html ma-rezepte.html
72 <section class="breadcrumbs">
73   <pb-view id="title-view1" src="document1" xpath="//teiHeader/fileDesc/titleStmt/title[@type='main']" view="single" disable-history="">
74     <pb-param name="header" value="short"/>
75   </pb-view>
76 </section>
77 <main class="content-body">
78   <pb-navigation direction="backward" unit="page" emit="transcription" subscribe="transcription">
79     <paper-fab icon="icons:chevron-left"/>
80   </pb-navigation>
81   <div id="view1">
82     <pb-view src="document1" column-separator=".tei-cb" append-footnotes="append-footnotes" subscribe="transcription" emit="transcription" wait-for="#facsimile"/>
83   </div>
84   <pb-facsimile id="facsimile" base-uri="" type="image" default-zoom-level="0" show-navigation-control="show-navigation-control" show-navigator="show-navigator" subscribe="transcription">
85     <pb-navigation direction="forward" unit="page" emit="transcription" subscribe="transcription">
86       <paper-fab icon="icons:chevron-right"/>
87     </pb-navigation>
88   </pb-facsimile>
89 </main>
90 </app-header-layout>
91 </app-drawer-layout>
92 </>

```

In der Publikationsansicht sehen wir nun, dass die Navigationsbuttons noch nicht an gewünschter Stelle sind.

MS Augsburg, Universitätsbibl., Cod.III.1.2*43 (A1)

Hienach volgt vonn dem kochenn vnd har gemacht meyster Eberhart Ein koch herzog heinrichs zu lands hur Zum Ein salsenn von weichselnn zu machen

Item Wiltu machen ein gurte salsenn von weichselnn So 2 thue die weichsell in einen hafen vnd setz die auff ein glut vnd laß sie siedenn vnd laß dann wider erkaltenn vnd streich sie durch ein tuch vnd thue sie dann wider in den hafenn vnd setz sie auff ein glut vnd laß sie wol sieden vnd rutt sie piß sie dick wirt vnd thue dann honig dar an vnd getibens prot vnd negellein vnd gut gestüp vnd thue sie in ein feßlein sie pleibt dir gut drew oder vier iarr

Ein gurte salsen zu machen in der fastenn

Item Nym merrich vnd zustoß den in einem moerserr vnd nym mandelkernn oder nuß vnd zustoß die auch vnd geuß ein wein dar an Merrich bricht den stein garr serr wenn man in isset in der kost

Item Ein andre salsenn Salvia petrocilius menta vnd pfefferr das soll man zustossenn mit essig das ist ein salsen die macht lustig zu essenn

Item Ein essenn von gebratenn arbeissenn Nym gesorenn arbeyß vnd slach sie durch ein tuch oder durch ein sib vnd slach vil eyter darzu als vil der arbeyß sein vnd seude es in puttern vnd steck es an einen spiß vnd brot sie wol vnd beslach sie mit eytern vnd mit kraut vnd gib es hin versalz es nit

Ein holder muß zu machen Nym holderplur vnd zureib die in kuec milch vnd nym mel vnd mach ein mus dar auf das ist gur zu

Um diese Elemente entsprechend anzupassen, müssen wir die CSS-Informationen ändern. Hier können wir ebenfalls aus dem Template des Demo-Projekts die für das Styling der Navigationsbutton verantwortlichen Regeln übernehmen und in unser projektspezifisches Template übernehmen.



```
new-document 1* view.html ma-rezepte.html
12 <meta name="description" content="Default single text layout"/>
13 <meta name="pb-template" content="view.html"/>
14
15 <link rel="stylesheet" href="resources/css/theme.css"/>
16 <script type="module" src="pb-components-bundle.js" data-template="pages:load-components"/>
17 <style>
18   @media (min-width: 769px) {
19     .toolbar pb-navigation {
20       display: none;
21     }
22   }
23
24   .content-body {
25     position: relative;
26     display: flex;
27     justify-content: center;
28   }
29
30   .content-body pb-navigation {
31     position: fixed;
32     bottom: 45%;
33     --paper-fab-background: #35424b;
34     color: white;
35   }
36
37   .content-body pb-navigation[direction=backward] {
38     left: 20px;
39   }
40
41   .content-body pb-navigation[direction=forward] {
42     right: 20px;
43   }
44
45   @media (max-width: 769px) {
46     .content-body pb-navigation {
47       display: none;
48     }
49   }
50 </style>
```

Wir kopieren die Informationen unter die Styling-Informationen für die Faksimile-Webkomponente und ändern dabei noch die Farbe der Navigationsbutton.



```
new-document 1* view.html ma-rezepte.html
12 <meta name="description" content="Default single text layout"/>
13 <meta name="pb-template" content="view.html"/>
14
15 <link rel="stylesheet" href="resources/css/theme.css"/>
16 <script type="module" src="pb-components-bundle.js" data-template="pages:load-components"/>
17 <style>
18   @media (min-width: 769px) {
19     .toolbar pb-navigation {
20       display: none;
21     }
22   }
23
24   .content-body {
25     position: relative;
26     display: flex;
27     justify-content: center;
28   }
29
30   .content-body pb-navigation {
31     position: fixed;
32     bottom: 45%;
33     --paper-fab-background: #35424b;
34     color: white;
35   }
36
37   .content-body pb-navigation[direction=backward] {
38     left: 20px;
39   }
40
41   .content-body pb-navigation[direction=forward] {
42     right: 20px;
43   }
44
45   @media (max-width: 769px) {
46     .content-body pb-navigation {
47       display: none;
48     }
49   }
50 </style>
```

Nach dem Aktualisieren der Publikationsansicht unseres Manuskripts, haben die Buttons nun eine andere

Farbe und sind vertikal mittig ausgerichtet.

Hienach volgt vonn dem kochenn vnd hat gemacht meyster Eberhart Ein koch herczog heinrichs zu lands hut Zum Ein salsenn von weichselnn zu machen Item Wilku machen ein gutte salsenn von weichselnn So z thue die weichsell in einen hafen und setz die auff ein glur vnd laß sie siedenn vnd laß dann wider erkalten vnd streich sie durch ein tuch vnd thue sie dann wider in den haffenn vnd setz sie auff ein glut vnd laß sie wol sieden vnd rurr sie piß sie dick wirt vnd thue dann honig dar an vnd geribens bro vnd negellein vnd gut gestüp vnd thue sie in ein feßlein sie pleibt dir gut drew oder vier iarr
Ein gute salsen zu machen in der fastenn
Item Nym merrich vnd zustoß den in einem moerser vnd nym mandelkernn oder nuß vnd zustoß die auch vnd geuß ein wein dar an Merrich bricht den stein gart serr wenn man in isset in der kost Item Andre salsenn Saluia petrocellis menta vnd pfeffetr das soll man zustosseñ mit essig das ist ein salsen die macht lustig zu essen Item Ein essenn von gebratenn arabisseñ Nym gesorenñ arbeit vnd slach sie durch ein tuch oder durch ein sib vnd slach vil ejerr darzu als vil der arbeit sein vnd seud es in puttern vnd steck es an einen spiß vnd brot sie wol vnd beslach sie mit ejern vnd mit kraut vnd gib es hin versalz es nit
Ein holder muß zu machen Nym holderplur vnd zureib die in kuce

- In der Publikationsansicht möchten wir nun außerdem für die einzelnen Manuskripte auch die Metadaten darstellen. In der Demo-Sammlung haben wir dabei in dem Projekt "Mauritius Ferber" eine ein- und ausklappbare Metadaten-Darstellung gefunden, die wir für unser Projekt nachnutzen wollen. Wir navigieren dafür in eXide zu dem entsprechenden Page-Template (danticus.html). Dort finden wir innerhalb des `<app-header>`-Abschnitts sogar einen Hinweis darauf, dass in der entsprechenden Darstellung die Toolbar, die eigentlich nur das Inhaltsverzeichnis anzeigt, um ein Toggle-Element für die Metadaten ergänzt wurde.

```

ma-rezepte.html*          danticus.html
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73

```

```

</style>
</custom-style>
</head>
<body unresolved="unresolved">
  <pb-page data-template="pages:pb-page" unresolved="unresolved">
    <pb-document id="document1" data-template="pages:pb-document"/>
    <app-drawer-layout force-narrow="force-narrow">
      <app-drawer data-template="lib:include" data-template-path="templates/drawer.html"/>
      <app-header>
        <app-header slot="header" reveals="reveals" fixed="fixed" effects="waterfall">
          <!-- Overwrite toolbar to add metadata toggle -->
          <app-toolbar data-template="lib:include" data-template-path="templates/toolbar.html">
            <paper-button id="metaToggle" raised="raised" data-target="toolbar-before">
              <iron-icon icon="icons:info-outline"/>Metadaten
            </paper-button>
          </app-toolbar>
        </app-header>
      </app-drawer-layout>
    </pb-page>
  </body>

```

Um diesen Button in unserem Page-Template zu übernehmen, kopieren wir also die zweite `<app-toolbar>` im `<app-header>` und fügen diese an gleicher Stelle in unser projektspezifisches HTML-Template ein. Wie bei den Copy-Paste-Aktionen für andere Webkomponenten dürfen wir hier ebenfalls nicht vergessen, zusätzlich auch den CSS-Code für die ausklappbare Metadaten-Anzeige zu übernehmen.

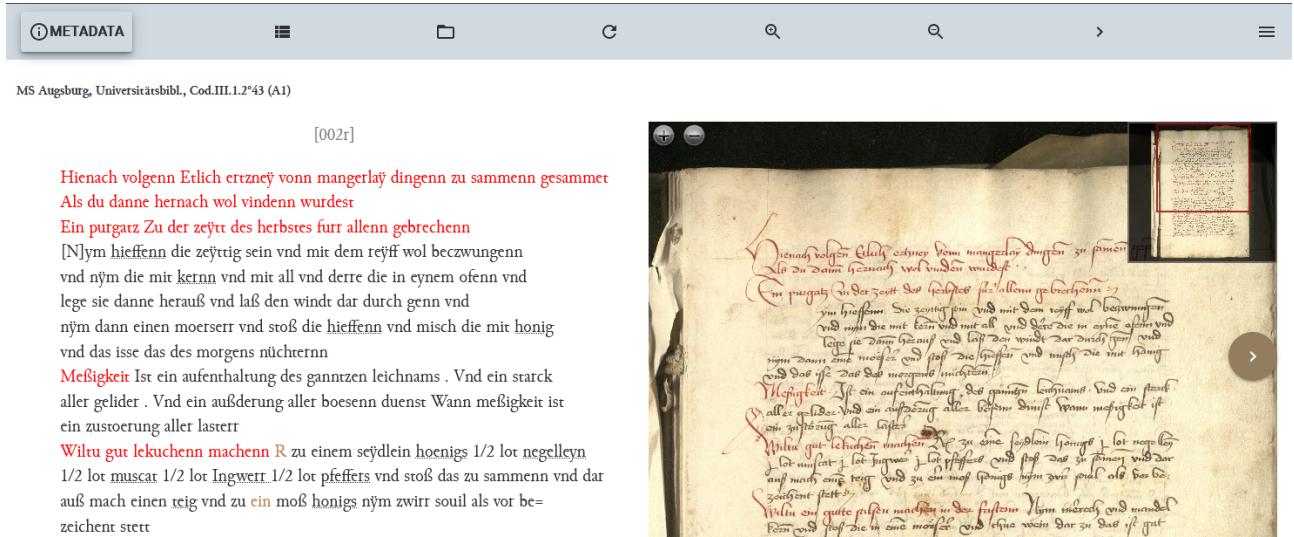


```

63
64    @media (max-width: 769px) {
65        .content-body pb-navigation {
66            display: none;
67        }
68    }
69
70    .metadata {
71        width: 100%;
72        padding: 0 20px 20px;
73    }
74    .metadata pb-view {
75        font: var(--pb-base-font);
76    }
77    .metadata h3 pb-view {
78        font-family: var(--pb-heading-font-family);
79        font-weight: var(--pb-heading-font-weight);
80    }
81
82    </style>
83 </head>
84
85 <body>
86     <pb-page data-template="pages:pb-page" unresolved="unresolved">
87         <pb-document id="document1" data-template="pages:pb-document"/>
88         <app-drawer-layout force-narrow="force-narrow">
89             <app-drawer data-template="lib:include" data-template-path="templates/drawer.html"/>
90             <app-header-layout>
91                 <app-header slot="header" fixed="fixed">
92                     <app-toolbar data-template="lib:include" data-template-path="templates/menu.html"/>
93                     <app-toolbar data-template="lib:include" data-template-path="templates/toolbar.html"/>
94                 </app-header>
95                 <pb-drawer toggle="metaToggle" class="tocDrawer" emit="toc" subscribe="transcription">
96                     <div class="drawer-content">

```

Wenn wir in der Publikationsansicht unser Browser-Fenster aktualisieren, erscheint nun oben links in der Navigationsleiste ein Metadaten-Button.



MS Augsburg, Universitätsbibli., Cod.III.1.2*43 (A1)

[002r]

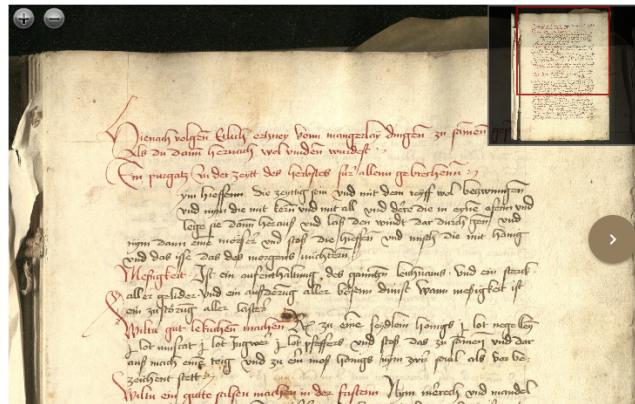
Hienach volgenn Etlich ertzney vonn mangerlāy dingenn zu sammenn gesammet
Als du danne hernach wol vindenn wurdest

Ein purgatz Zu der zeitt des herbstes furr allenn gebrechenn

[N]lym hieffenn die zeittig sein vnd mit dem reiff wol beczwungenn
vnd nym die mit kerñ vnd mir all vnd derre die in eyнем ofenn vnd
leg sie danne heraus vnd laß den windt dar durch genn vnd
nym dann einen moerserr vnd stoß die hieffenn vnd misch die mit honig
vnd das isse das des morgens nüchternn

Meßigkeit Ist ein aufenthaltung des ganzen leichnams . Vnd ein stark
aller gelider . Vnd ein außderung aller boesen duenst Wann meßigkeit ist
ein zustoerung aller lastert

Wiltu gut lekuchenn machenn R zu einem seydlein hoenigs 1/2 lot negelley
1/2 lot muscar 1/2 lot Ingwert 1/2 lot pfeffers vnd stoß das zu sammenn vnd dar
auß mach einen teig vnd zu **ein** moß honigs nym zwirt souil als vor be=
zeichnet stett



Um diesen ausklappbaren Seitenbereich mit Metadaten zu füllen, müssen wir jedoch noch ein paar Anpassungen in unserem ODD vornehmen.

- Wir wechseln also erneut in den ODD-Editor und suchen dort nach den Modellspezifikationen für den `<teiHeader>`. Wir legen hier ein neues inline-Modell an und weisen im Predicate mit der XQuery-Angabe `$parameters?mode='commentary'` darauf hin, dass dieses Modell nur für die zuvor im HTML-Template angelegte Metadaten-Toggle-Komponente gilt. Im Metadatenbereich möchten wir einerseits Informationen zum Repository, zur Signatur sowie zur physischen Beschreibung des Manuskripts anzeigen. Dementsprechend basteln wir ein Template in HTML mit Überschriften und geben in doppelten eckigen Klammern Platzhalter-Elemente an, denen wir darunter die entsprechenden Parameter zuordnen: Unter `[[repository]]` im Template sollen schließlich jene Daten angegeben werden, die wir im `teiHeader` im Element `<repository>` finden. Daher geben wir bei diesem Parameter vom `<teiHeader>`-Element ausgehend den entsprechenden Pfad `'//repository'` an. Genauso gehen wir auch für die weiteren Parameter vor. Beim `<idno>`-Element bedarf es genauerer Spezifikation, da es einerseits die `<idno>`-Elemente in unserem Register, und andererseits aber auch das `<idno>`-Element im `<teiHeader>` gibt. Daher fügen wir hier als XPath-Bedingung noch das Attribut `@type="shelfmark"` hinzu. Zuletzt stellen wir noch sicher, dass es für all die in den Parametern angegebenen Elemente auch tatsächlich Modellspezifikationen gibt. Dabei reicht es, dass es für jedes Element, das wir in den Metadaten ausgeben wollen, ein inline-Modell

existiert.

The screenshot shows the TEI Editor interface with the 'ma-rezepte.odd' file open. A red box highlights the 'teiHeader' element in the tree view. The 'Predicate' field in the 'model [inline]' section contains the expression '\$parameters?mode='commentary''. Below it, the 'Template' field displays a div structure with three children: 'repository', 'signature', and 'physDesc'. Red boxes also highlight the 'idno', 'physDesc', and 'repository' fields in the left sidebar under the 'Element Spezifikationen' section. The 'behaviour' dropdown is set to 'inline'.

Wenn wir schließlich alle Modelle erstellt haben und die Publikationsansicht aktualisieren, erscheint nach einem Klick auf den Metadaten-Button nun auch ein entsprechender Inhalt.

The screenshot shows the 'METADATA' view with the 'Metadata' tab selected. On the left, there are sections for 'Repository' (Universitätsbibliothek Augsburg) and 'Signatur' (gebrecenn). The 'Physische Beschreibung' section contains the text: 'dangenn zu sammenn gesammet'. To the right, there is a large thumbnail image of a medieval manuscript page with handwritten text in two columns.

4. Erstellung einer eigenen App

Damit wir nun zu einer eigenständigen Edition bzw. Applikation gelangen, die wir auf unserer eigenen Website präsentieren können, müssen wir die erstellten ODDs, die HTML Templates und Webkomponenten zusammenführen.

- Wir klicken dafür zunächst in der Navigationsleiste auf **Administration** und wählen im Drop-Down "Anwendung generieren".

Hienach volgenn Etlich ertzney vonn mangerlay dingenn zu sammenn gesammet
Als du danne hernach wol vindenn wurdest
Ein purgatz Zu der zeytt des herbstes furr allenn gebrechenn
[N]ym hieffenn die zeytig sein vnd mit dem reyff wol beczwungenn
vnd nym die mir kenn vnd mit all vnd derre die in eyнем ofenn vnd
lege sie danne heraus vnd laß den windt dar durch genn vnd
nym dann einen moerserr vnd stoß die hieffenn vnd misch die mit honig
vnd das isse das des morgens nüchterm
Meßigkeit Ist ein aufenthaltung des ganzen leichnams . Vnd ein stark
aller gelider . Vnd ein außerder aller boesenn duenst Wann meßigkeit ist
ein zustorung aller laster
Wiltu gut lekuchenn machenn R zu einem seydelein hoenigs 1/2 lot negelleyen
1/2 lot muscat 1/2 lot Ingwert 1/2 lot pfeffers vnd stoß das zu sammenn vnd dar
auß mach einen teig vnd zu ein moß honigs nym zwirr soul als vor be=
zeichent stett
Wiltu ein gute salsen machen in der fastenn Nym merrech vnd mandel
kernn vnd stoß die in einem moerserr vnd thue wein dar zu das ist gut

- In dem Tab, das sich daraufhin öffnet, wählen wir als Erstes jene ODDs aus, die für unsere App verwendet werden sollen. Dies ist in unserem Fall das ODD "MA Rezepte" sowie "MA Zutaten". Außerdem geben wir hier die URL zur App sowie das gewünschte Kürzel an.

Anwendung generieren

ODD Auswählen

- MA Rezepte
- Erasmus Adagia
- Dantisca's Letters Demo
- Dantisca's Letters
- TEI Publisher Customization for ODD Authoring
- EBBE
- DOCK Output Preview
- Deutsches Textarchiv
- Annotations
- Serafin Letters
- Docbook v5
- Bogactwa mowy polskiej
- Graves' Letters
- Shakespeare Plays
- ODD_TIMEUS
- DOCX Import
- JATS
- Potocki Moralia
- TEI Publisher Base
- Lelewel Letters
- Van Gogh Letters
- MA-Zutaten

URL zur eindeutigen Identifizierung der App
<http://exist-db.org/apps/ma-rezepte-2023>

Kurzname zur Verwendung in URLs und Dateinamen
ma-rezepte-23

Bei den weiteren Feldern lassen wir die Eingabezeile zum Namen der Subcollection frei, da wir hier die vordefinierte Struktur nutzen wollen. Wir geben den Titel unserer App an, und wählen unsere zuvor erstellte HTML-Vorlage. Die Strukturierung in der Standardansicht soll sich nach den <div>-Elementen richten und auch beim Standard-Volltextindex wählen wir wieder eine Organisation nach <div>-Elementen. Hinsichtlich des Benutzerkontos hat sich bei uns herausgestellt, dass wir nur mit dem Benutzer **tei-demo** und dem entsprechenden Passwort **demo** später auch ohne weitere Konfigurierungen Dokumente hochladen können, weshalb wir diesen Standardnutzer nun auch für unsere App anlegen.

URL zur eindeutigen Identifizierung der App
http://exist-db.org/apps/ma-rezepte-2023

Kurzname zur Verwendung in URLs und Dateinamen
ma-rezepte-23

Name der Subcollection mit den TEI Dokumenten
data

Titel der App zur Anzeige in Dashboard usw.
Mittelalterliche Rezepte

Wähle die HTML-Vorlage aus, die als Standardansicht verwendet werden soll
HTML-Vorlage
MA Rezepte Template

Wähle die Texteinheit, die jeweils auf einer Seite angezeigt werden soll. Dies kann ein ganzer Abschnitt (div) sein oder eine einzelne Seite des Originals. Letztere Option funktioniert nur, wenn im TEI Seitenumbrüche korrekt ausgezeichnet sind.
Standardansicht
Nach div (Kapitel / Abschnitt ...)

Definieren Sie die kleinsten Texteinheiten, für die ein Volltextindex erstellt wird. Bei längeren Dokumenten mit mehreren Abschnitten (divs), empfiehlt es sich in der Regel, "Für Abschnitt (div) erstellen" zu wählen. Wenn keine Unterteilungen vorhanden sind, wähle "Für den gesamten Text erstellen".
Standard-Volltextindex
Für Abschnitt (div) erstellen

Benutzerkonto für administrative Aufgaben. Der Benutzer wird angelegt, wenn er noch nicht existiert.
Das Benutzerkonto, dem die App gehören wird.
tei-demo

Passwort für den Benutzer, der diese App besitzt.

 SPEICHERN/ERZEUGEN

Nachdem das Formular vollständig bearbeitet wurde und wir auf "SPEICHERN/ERZEUGEN" geklickt haben, erscheint ein kleines Dialogfenster, das uns bestätigt, dass unsere App erfolgreich erstellt wurde.



Definieren Sie die kleinsten Texteinheiten, für die ein Volltextindex erstellt wird. Bei längeren Dokumenten mit mehreren Abschnitten (divs), empfiehlt es sich in der Regel, "Für Abschnitt (div) erstellen" zu wählen. Wenn keine Unterteilungen vorhanden sind, wähle "Für den gesamten Text erstellen".

- Wenn wir unsere App nun über den Link (oder andernfalls über die eXist-db, wo unsere App nun so wie der *TEI Publisher* in einer eigenen Kachel erscheint) geöffnet haben, sehen wir, dass unsere App im Grunde wie der *TEI Publisher* aufgebaut ist, wobei sie bereits unseren Projekttitel trägt. Zunächst müssen wir sicher gehen, dass wir auch dort mit den zuvor eingetragenen Benutzerdaten angemeldet sind. Daraufhin sind die Projektdateien erneut hochzuladen, da diese nicht automatisch übernommen wurden.

Mittelalterliche Rezepte

Dokumente

Sortiere nach
Titel

Query scope
Search content

Suche nach
Q

SUCHE

Datei hochladen

Name	Änderungsdatum	Typ	Größe
manuscript1.xml	17.10.2023 10:02	XML Document	30 KB
manuscript2.xml	17.10.2023 09:45	XML Document	24 KB
Sachbegriffe.xml	18.09.2023 07:14	XML Document	42 KB

Hochladen

HOCHLADEN

Dateien vom Typ .xml, .tei, .odd, .docx, .md, .mei hier hinziehen zum Hochladen.

Mit einem Klick auf unser erstes Manuskript sehen wir auch gleich, dass das ODD direkt mit unserem XML verknüpft wurde. Unter dem Button **Administration** in der Navigationsleiste können wir uns außerdem einen Überblick darüber verschaffen, welche Optionen es für die Bearbeitung jetzt noch gibt.

The screenshot shows the TEI Publisher interface. On the left, there's a sidebar with navigation links like 'Start' and 'Herunterladen'. The main area displays a list of ODD files: 'ODD editieren: ma-rezepte.odd', 'ODD editieren: ma-zutaten.odd', 'ODD editieren: docx.odd', 'Download App as .xar', 'API Documentation', and 'Webcomponents API'. Below this list, there's a block of German text from a medieval manuscript. To the right of the text is a large image of a double-page spread from a medieval manuscript, showing dense handwritten text in two columns.

→ Es ist also auch jetzt noch möglich, Änderungen an den beiden ODDs vorzunehmen, sowie das HTML-Template oder die XMLs in eXide anzupassen.

- Einzig das XML mit dem Zutatenregister wird noch nicht richtig dargestellt, da es ebenfalls standardmäßig mit dem MA-Rezepte-ODD für die Manuskripte verknüpft wird und nicht mit dem eigens für die Zutaten erstellten MA-Zutaten-ODD.

The screenshot shows the TEI Publisher interface with a search bar containing 'Sachbegriffe'. Below the search bar, it says 'Die Anfrage an den Server schlug fehl: Document Sachbegriffe.xml not found [at line 536 column 9 in module /db/apps/ma-rezepte-23/modules/lib/api/document.xql]'. The main content area shows a list of links to Wikidata entities:

- <https://www.wikidata.org/entity/Q1060458>
Siehe weitere Schreibweisen dieser Zutat im [Register](#)
- <https://www.wikidata.org/entity/Q89>
Siehe weitere Schreibweisen dieser Zutat im [Register](#)
- <https://www.wikidata.org/entity/Q25237>
Siehe weitere Schreibweisen dieser Zutat im [Register](#)
- <https://www.wikidata.org/entity/Q107263051>
Siehe weitere Schreibweisen dieser Zutat im [Register](#)
- <https://www.wikidata.org/entity/Q107246700>
Siehe weitere Schreibweisen dieser Zutat im [Register](#)

Wie in der [Dokumentation des TEI Publisher](#) beschrieben, kann man die ODD- und HTML-Verknüpfung für einzelne Dokumente aber über die Datei "config.xqm" im Ordner **modules** unserer App konfigurieren.

Dementsprechend verknüpfen wir für unser Projekt das Register-XML mit dem ODD für die Zutaten und dem "view.html" anstelle des eigens angepassten HTML-Templates.

The screenshot shows the eXide XML editor with the file 'config.xqm' open. The code contains a 'switch' statement that maps different document URIs to specific ODD files and view configurations. A red box highlights this section of the code:

```

switch ($docUri)
  case "Sachbegriffe.xml" return
    map {
      "oddi": "ma-zutaten.odd",
      "view": "body",
      "depth": $config:pagination-depth,
      "fill": $config:pagination-fill,
      "template": "view.html"
    }
  default return
()

```

Nach dem Aktualisieren des Zutatenregisters in der Publikationsansicht, wird auch dieses XML wieder

richtig dargestellt.

The screenshot shows the TEI Publisher application's search results page. At the top, there is a navigation bar with links for 'About', 'Start', 'Herunterladen', 'Administration', 'Suche nach' (with a magnifying glass icon), 'Sprache' (set to 'Deutsch'), and a user account section. Below the navigation bar, there are several search icons: a list icon, a folder icon, a refresh/circular arrow icon, a magnifying glass icon, another magnifying glass icon, and a three-dot menu icon. The main content area is titled 'ZUTATENREGISTER'. It lists various ingredients with their alternative spellings in Frühneuhochdeutsch:

- Zutat: [Agraz](#)
Alternative Schreibweisen im Frühneuhochdeutschen:
agraeß, agras, agraz, agres, agresß, agrest, agreste, agroße
- Zutat: [Apfel](#)
Alternative Schreibweisen im Frühneuhochdeutschen:
apfel, äpfelen, apfell, apffel, apffel, apfll, aphel, aphell, aphel, aphell, aplh, appel, appfel, äppfel, appfelen, appfeln, äppfeln, appffel, appffel, apphel, apphell, apphellen, apphels, mala
- Zutat: [Arzni](#)
Alternative Schreibweisen im Frühneuhochdeutschen:
arcenie, artzenie, artzny, ercznei, erczney, ertzney
- Zutat: [Bertram](#)
Alternative Schreibweisen im Frühneuhochdeutschen:
pertram, petram, pyrterum
- Zutat: [Bier](#)
Alternative Schreibweisen im Frühneuhochdeutschen:
pyer

Sachbegriffe

ZUTATENREGISTER

- Zutat: [Agraz](#)
Alternative Schreibweisen im Frühneuhochdeutschen:
agraeß, agras, agraz, agres, agresß, agrest, agreste, agroße
 - Zutat: [Apfel](#)
Alternative Schreibweisen im Frühneuhochdeutschen:
apfel, äpfelen, apfell, apffel, apffel, apfll, aphel, aphell, aphel, aphell, aplh, appel, appfel, äppfel, appfelen, appfeln, äppfeln, appffel, appffel, apphel, apphell, apphellen, apphels, mala
 - Zutat: [Arzni](#)
Alternative Schreibweisen im Frühneuhochdeutschen:
arcenie, artzenie, artzny, ercznei, erczney, ertzney
 - Zutat: [Bertram](#)
Alternative Schreibweisen im Frühneuhochdeutschen:
pertram, petram, pyrterum
 - Zutat: [Bier](#)
Alternative Schreibweisen im Frühneuhochdeutschen:
pyer
- Da die App aber weiterhin im TEI-Publisher-Design erscheint, widmen wir uns zunächst der Anpassung des Layouts. Wir möchten jedenfalls das Logo sowie die Farben unserer App für unser Projekt adaptieren. Dafür wechseln wir wieder in einen Tab, in dem eXide geöffnet ist. Wir navigieren als erstes innerhalb unserer App zum Ordner **templates/pages**, wo wir die Datei "ma-rezepte.html" finden, in der alle weiteren Bausteine unserer Website vorzufinden sind. Im Element `<app-header>` sehen wir schließlich, dass die Toolbar über ein anderes Template geladen wird, und zwar "menu.html", das sich im übergeordneten Ordner **templates** befindet.

```

outline directory
new-document 1 ma-rezepte.html
5v
81      </style>
82  </head>
83
84  <body>
85    <pb-page data-template="pages:pb-page" unresolved="unresolved">
86      <pb-document id="document1" data-template="pages:pb-document"/>
87      <app-drawer-layout force-narrow="force-narrow">
88        <app-drawer data-template="lib:include" data-template-path="templates/drawer.html"/>
89        <app-header>
90          <app-header slot="header" fixed="fixed">
91            <app-toolbar data-template="lib:include" data-template-path="templates/menu.html"/>
92            <app-toolbar data-template="lib:include" data-template-path="templates/toolbar.html"/>
93          </app-header>
94          <pb-drawer toggle="metalogic" class="tcbDrawer" emit="toc" subscribe="transcription">
95            <div class="drawer-content">
96              <h3>Metadata</h3>
97              <pb-view id="commentary" src="document1" xpath="//teiHeader" view="single" subscribe="toc">
98                <pb-param name="mode" value="commentary"/>
99              </pb-view>
100            </div>
101          </pb-drawer>
102          <pb-drawer toggle="tocToggle" class="toc" emit="toc" subscribe="transcription">
103            <div class="drawer-content">
104              <h3></h3>
105              <pb-118n key="document.contents">Contents</pb-118n>
106            </div>
107          </pb-drawer>
108        </div>
109      </app-drawer>
110      <pb-load id="toc" url="api/document/{doc}/contents?target=transcription&icons=true" expand="expand" src="document1" subscribe="toc" load-once="load-once">
111        <pb-118n key="dialogs.loading">Loading</pb-118n>
112      </pb-load>
113    </pb-page>
114  </body>
115</html>

```

In diesem HTML finden wir leider keine eingebundene Ressource, sondern nur das Attribut `@class="logo"`, in dem sich möglicherweise der Link zum Logo befindet. Um herauszufinden, wo das CSS abgelegt wurde, scrollen wir in unserer Datei also noch mal an den Anfang.

```

new-document 1 ma-rezepte.html
1 <html>
2
3   <head>
4     <meta charset="utf-8"/>
5     <meta name="viewport" content="width=device-width, minimum-scale=1, initial-scale=1, user-scalable=yes"/>
6     <link rel="shortcut icon" type="image/x-icon" href="resources/images/favicon.ico"/>
7     <link rel="shortcut icon" type="image/png" href="resources/images/favicon-16.png" sizes="16x16"/>
8     <link rel="shortcut icon" type="image/png" href="resources/images/favicon-24.png" sizes="24x24"/>
9     <link rel="shortcut icon" type="image/png" href="resources/images/favicon-32.png" sizes="32x32"/>
10    <link rel="shortcut icon" type="image/png" href="resources/images/favicon-64.png" sizes="64x64"/>
11
12   <title data-template="config:app-title"/>
13   <meta name="description" content="VA Rezepte Template"/>
14   <link rel="stylesheet" href="resources/css/theme.css"/>
15   <script type="module" src="pb-components-bundle.js" data-template="pages:load-components"/>
16   <style>
17     body {
18       --pb-view-height: calc(100vh - 200px);
19     }
20

```

Dort finden wir den Path zum standardmäßig eingebundene CSS ("theme.css"), wo wir anschließend nach der entsprechenden Klasse.

```

new-document 1* ma-rezepte.html menu.html* theme.css
1 @import url("../fonts/font.css");
2
3 /* Due to a safari issue, relative paths to background images won't work as variables, so we list them here */
4
5 .logo {
6   background-image: url("../images/tei-publisher-logo-contrast-color.svg");
7 }
8
9 .splash {
10   background-image: url("../images/tei-publisher-logo-color.svg");
11 }
12
13 .menubar {
14   background-image: none;
15 }
16

```

Hier wird ersichtlich, dass das Logo tatsächlich über das CSS eingebunden wurde und dass die Logos über den Pfad 'db/apps/ma-rezepte-23/resources/images' zu finden sind. Dort laden wir nun unser eigenes Logo Ordner hoch.

Name	Upload Files	Permiss...	Owner	Group	Last Modified
body-bg.png	-rw-r--r--	tei-de...	tei	Today 06:48:2	
favicon-16.png	-rw-r--r--	tei-de...	tei	Today 06:48:2	
favicon-24.png	-rw-r--r--	tei-de...	tei	Today 06:48:2	
favicon-32.png	-rw-r--r--	tei-de...	tei	Today 06:48:2	
favicon-48.png	-rw-r--r--	tei-de...	tei	Today 06:48:2	
favicon-64.png	-rw-r--r--	tei-de...	tei	Today 06:48:2	
favicon.ico	-rw-r--r--	tei-de...	tei	Today 06:48:2	

Als nächstes müssen wir im CSS auf unsere Bilddatei verweisen. Bevor wir dies tun, suchen wir in dem "theme.css" aber auch jene Stelle, an der die Farbeigenschaften für die Menüleiste hinterlegt sind und sehen dort, dass die Farbschemata größtenteils in Variablen abgelegt wurden.

```

new-document 1 ma-rezepte.html menu.html theme.css* new-document 2*
173   color: var(--pb-menubar-color);
174 }
175
176 .toolbar a {
177   color: inherit;
178   text-decoration: none;
179 }
180
181 [drawer-toggle] {
182   padding-right: 0;
183 }
184
185 .menubar {
186   justify-content: space-between;
187   background-color: var(--pb-menubar-background-color);
188   color: var(--pb-menubar-color);
189   --pb-search-suggestions-color: var(--pb-menubar-color);
190   --pb-search-suggestions-background: var(--pb-menubar-background-color);
191 }
192

```

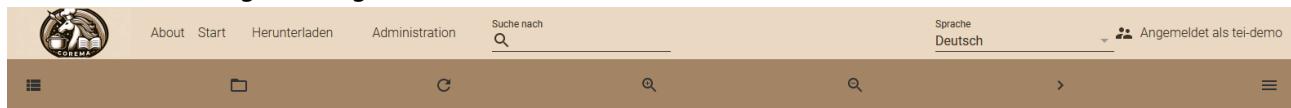
Um unsere projektspezifischen Anpassungen separat zu bearbeiten, erstellen wir eine weitere CSS-Datei und überschreiben dort jene Klassen, die wir ändern möchten. Wir ändern einerseits die Bilddateien in der `.logo`-Klasse und zusätzlich auch in einer weiteren Klasse (`.splash`), die für die Übergangsbilder beim Laden verantwortlich ist. Außerdem passen wir die Text- und Hintergrundfarben in der Menüleiste und der darunterliegenden Navigationsleiste an, indem wir entsprechende Variablen einführen und die alten Variablen mit unseren Farbcodes überschreiben.

```

new-document 1 ma-rezepte.html menu.html theme.css ma-rezepte.css
1 .logo {
2   background-image: url("../images/logo-marezepte.png");
3 }
4
5 .splash {
6   background-image: url("../images/logo-marezepte.png");
7 }
8
9
10 body {
11   --pb-mar-color: #3d3224;
12   --pb-mar-color-dark: #a38566;
13   --pb-menubar-color: var(--pb-mar-color);
14   --pb-menubar-background-color: #ead8c3;
15   --pb-color-focus: var(--pb-mar-color);
16   --pb-lang-input-color: var(--pb-mar-color);
17   --pb-lang-label-color: var(--pb-mar-color);
18   --pb-toolbar-background-color: var(--pb-mar-color-dark);
19 }

```

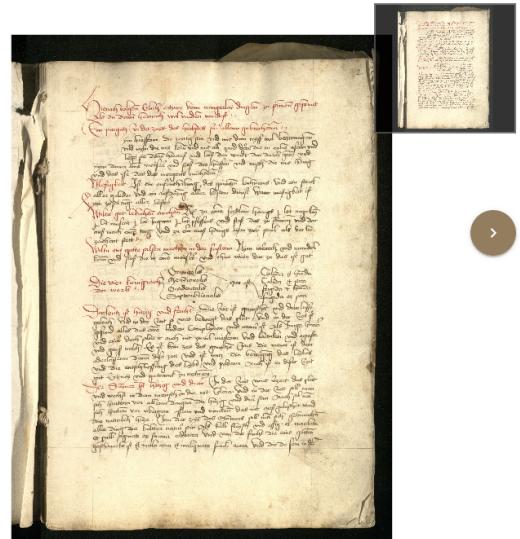
Wenn wir nun erneut in die Publikationsansicht gehen und die Seite aktualisieren, erscheint unsere App nun mit unserem eigenen Logo und in anderen Farben.



MS Augsburg, Universitätsbibli., Cod.III.1.2*43 (A1)

[002r]

Hienach volgenn Etlich erzneÿ vonn mangerlaÿ dingenn zu sammenn gesammet
Als du danne hernach wol vindenn wurdest
Ein purgatz Zu der zeÿt des herbstes furr allenn gebrechenn
[N]ym hieffenn die zeÿtig sein vnd mit dem reÿf wol beczwungenn
vnd nÿm die mit kern vnd mit all vnd derre die in eynen ofenn vnd
lege sie danne herauß vnd laß den windt dar durch genn vnd
nÿm dann einen moersert vnd stoß die hieffenn vnd misch die mit honig
vnd das iss das des morgens nüchtern
Meßigkeit Ist ein aufenhaltung des ganztzen leichnams . Vnd ein starck
aller gelider . Vnd ein außderung aller boesenn duenst Wann meßigkeit ist
ein zustrocung aller lastet
Wilu gut lekuchenn machen R zu einem seydelein hoenigs 1/2 lor negelleyn
1/2 lor muscar 1/2 lor Ingwert 1/2 lor pfeffers vnd stoß das zu sammenn vnd da
auß mach einen reig vnd zu ein moß honigs nÿm zwirt soul als vor be=
zeichent stett
Wilu ein gute salsen machen in der fastenn Nÿm merrech vnd mandel
kernn vnd stoß die in einem moerserr vnd thue wein dar zu das ist gut
Die vier kunigreich der werlt
Orientalis [que est] Calida et humida
Meridionalis [que est] Calida et sicca
Occidentalis que est frigida et humida
Septembrisinalis [que est] Frigida et sicca
Der lento ist hirzig vnd feucht Die Zeit ist gemischet vnd dem luftt



- Zuletzt möchten wir in unsere App noch eine weitere Seite einbinden, auf der das Projekt vorgestellt werden soll. Wir haben dafür bereits eine [About-Seite als XML](#) vorbereitet. Diese externe Seite laden wir nun über die App im Dokumentebrowser hoch.

The screenshot shows the teipublisher application interface. On the left, there's a search interface with fields for 'Dokumente' (Title, Query scope, Search content, Suche nach), 'Sprache' (Language, Zeige 50 häufigste, deu), and 'Sachbegriffe'. It lists two entries: 'MS Augsburg, Universitätsbibl., Cod.III.1.2°43 (A1)' and 'MS Solothurn, Zentralbibliothek, Cod. S 490 (So1)'. Each entry has a 'HERUNTERLADEN' button. A red box highlights the 'Über das Projekt' section at the bottom. On the right, a 'Hochladen' (Upload) dialog is open, showing a red box around the 'HOCHLADEN' section which contains the text: 'Dateien vom Typ .xml, .tei, .odd, .docx, .md, .mei hier hinziehen zum Hochladen.' and a file named 'about.xml'.

→ Hier können wir auch sehen, dass noch nicht alle Bereiche farblich angepasst sind, und es dafür noch weiterer Nachbearbeitungsschritte im CSS bedarf.

Wenn wir das About-XML anwählen, verfügen wir aber jedenfalls über diese Ansicht:

The screenshot shows the 'Über das Projekt' (About the Project) page. The top navigation bar includes the logo 'tei Publisher', 'About', 'Start', 'Herunterladen', 'Administration', 'Suche nach', 'Sprache Deutsch', and 'Angemeldet als tei-demo'. Below the navigation is a header with icons for 'Über das Projekt', 'Herunterladen', 'Administration', 'Suche nach', and 'Sprache'. The main content area is titled 'Über das Projekt' and contains a detailed description of the project's goals and methods.

Über das Projekt

Dieses Beispielprojekt beschäftigt sich mit der digitalen Edition von frühneuhochdeutschen Manuskripten aus dem Mittelalter, die Rezepte beinhalten. Die Daten für das Beispielprojekt wurden vom Projekt [CoReMa](#) übernommen, wobei sich die Zielsetzung im Rahmen des DigEdTnT-Projekts darauf beschränkt, nach der Erstellung von Transkripten auf Grundlage der Manuskript-Digitalisate, die textuelle Struktur sowie die in den Rezepten angeführten Zutaten TEI-konform zu annotieren, und zusätzlich die Zutaten mit Wikidata-Normdaten anzureichern. Abschließend sollen die annotierten und normalisierten Daten publiziert werden.

Im Zuge der Umsetzung dieses Beispielprojekts in der [Pipeline_1](#) wurden die Faksimiles der Manuskriptsammlungen A1 ([Augsburg, Universitätsbibliothek, Cod.III.1.2°43](#)) sowie So1 ([Solothurn, Zentralbibliothek, Cod. S 490](#)) für die Transkriptionsarbeit in FromThePage herangezogen und die in FromThePage verfügbaren Annotationsmöglichkeiten getestet. Das Ziel war hierbei, Transkripte der einzelnen Manuskriptseiten zu erstellen, die textuelle Merkmale wie Überschriften, Zeilenumbrüche, Streichungen, Ergänzungen usw. genau wiedergeben. Zusätzlich wurde in FromThePage auch die Möglichkeit des "Subject Linking" (Auszeichnung von Entitäten wie Personen, Orte oder eigens angelegten Kategorien) getestet. Die aus dem Subject Linking resultierenden Annotationen wurden vor der Weiterbearbeitung in ediarum wieder entfernt, um in ediarum ebenfalls die Annotation mit Normdaten zu testen. Dafür wurde eine bereits zur Verfügung stehende Zutatenliste in OpenRefine mit Einträgen aus Wikidata angereichert bzw. normalisiert, und diese in Form eines Registers exportiert. Außerdem wurden in ediarum weitere Annotationen vorgenommen, die über FromThePage nicht möglich waren bzw. Annotationen von FromThePage in ediarum entsprechend dem DTA-Basisformat erweitert. Nach Abschluss der Annotations- und Normalisierungsprozesse wurde das Beispielprojekt mittels teipublisher für die Publikation aufbereitet.

→ Auch diese Ansicht bedarf noch weiterer Anpassungen an das projekteigene Schema, die aber hier nicht weiter beschrieben werden. Da wir auf diese Projektseite von der Navigationsleiste aus zugreifen möchten, und nicht vom Dokumentenverzeichnis der App, erstellen wir im HTML einen entsprechenden Link. Und zwar müssen wir dafür wieder in das "menu.HTML", in dem sich die Navigationsleiste befindet, und fügen dort nach dem Logo einen Link zur About-Seite hinzu.

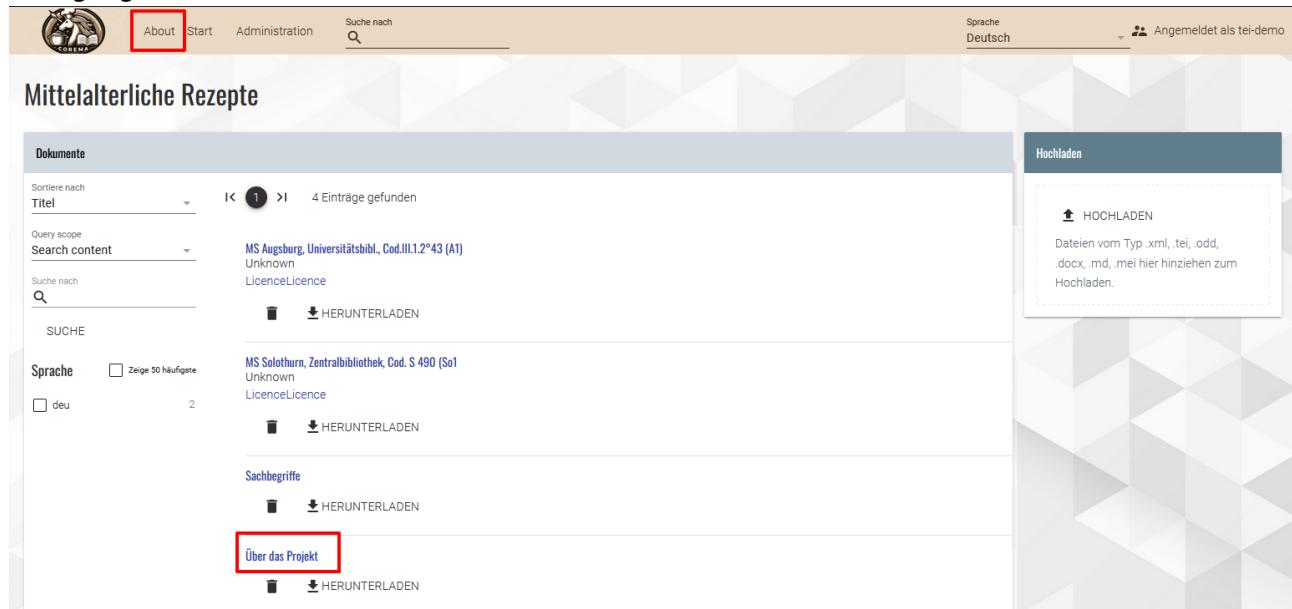


```

new-document 1 ma-rezepte.html menu.html theme.css ma-rezepte.css index.html
1 - <app-toolbar class="menubar">
2   <a href="${app}/index.html" class="logo" data-template="pages:parse-params"/>
3   <pb-media-query query="min-width: 1024px">
4     <a href="about.xml" about="About">About</a>
5   <a href="${app}/index.html" data-template="pages:parse-params">Start</a>
6   <paper-menu-button id="download-button">
7     <a slot="dropdown-trigger" data-i18n="[title]menu.download.title" title="i18n(download)">
8       <pb-i18n key="menu.download.title">Download</pb-i18n>
9     </a>
10    <paper-listbox slot="dropdown-content" data-template="pages:parse-params">
11      <paper-item data-template="pages:if-supported" data-template-media="print">
12        <a href="${app}/print/${doc}" target="_new">Print CSS</a>
13      </paper-item>

```

Zurück in unserer App sehen wir nun, dass gleich rechts neben dem Logo ein Link zur About-Seite zur Verfügung steht.



Um die XML-Datei zur About-Seite aus der Manuskriptliste zu entfernen, müssen wir nun noch entsprechende Änderungen der Konfigurationsdatei "config.xqm" vornehmen, die wir im Ordner **modules** finden.



```

outline directory new-document 1* manuscript1.xml ma-rezepte.html menu.html theme.css ma-rezepte.css index.html about.xml config.xqm
ma-rezepte-23 .gitignore
  data
  modules
    annotation-config.xqm
    app.xql
    config.xqm
    custom-aplxql
    facets.xql
    generate-pm-config.xql
    map.xql
    navigation-dbk.xql
    navigation-jats.xql
    navigation-xql
    nlp-config.xqm
    pm-config.xql
    query-db.xql
    query-jats.xql
    query-tei.xql
    query.xql
311  (~
312  : The root of the collection hierarchy whose files should be displayed
313  : on the entry page. Can be different from $config:data-root.
314  :
315  :
316  declare variable $config:data-default := $config:data-root;
317  :
318  (~
319  : A sequence of root elements which should be excluded from the list of
320  : documents displayed in the browsin view.
321  :
322  declare variable $config:data-exclude := (
323    doc($config:data-root || "/taxonomy.xml")//tei:text,
324    doc($config:data-root || "/about.xml")//tei:text
325  );
326  :
327  (~
328  : The main ODD to be used by default
329  :
330  declare variable $config:default-odd := "ma-rezepte.odd";
331

```

Dort ergänzen wir bei der Variablen-Deklaration `$config:data-exclude` eine weitere Zeile, die bewirkt, dass unsere XML-Datei mit der About-Seite nicht in der Liste nicht angezeigt wird. Zurück in der App

überprüfen wir, ob unsere Änderungen übernommen wurden:

The screenshot shows the TEI Publisher dashboard with the following details:

- Header:** Includes a logo of a horse reading a book, navigation links for "About", "Start", "Administration", and a search bar.
- Sidebar:** Shows a list of documents:
 - MS Augsburg, Universitätsbibl., Cod.III.1.2°43 (A1)**: Unknown, Licence
 - MS Solothurn, Zentralbibliothek, Cod. S 490 (S01)**: Unknown, Licence
- Search Bar:** Contains fields for "Sortiere nach Titel", "Query scope Search content", and "Suche nach".
- Language Filter:** Set to "deu".
- Download Buttons:** "HERUNTERLADEN" buttons for each document entry.
- Hochladen:** A section for uploading files with a note: "Dateien vom Typ .xml, .tei, .odd, .docx, .md, .mei hier hinziehen zum Hochladen."
- Footer:** TEI Publisher 0.1 / web components 2.12.8 / API 1.0.0 / background by starline

- Vor dem Export der App, ändern wir noch das Icon, das für unsere App in eXist aufscheinen soll. Dafür navigieren wir in eXide zum Ordner **packageservice** und überschreiben dort das bestehende Icon, indem wir unser Icon mit demselben Dateinamen (icon.png) hochladen. Im Dashboard erscheint nun anstelle des TEI-Publisher-Icons unser eigenes Projekt-Icon.

The screenshot shows the eXide Launcher interface with the following elements:

- Header:** "Launcher" with a rocket icon.
- Left Sidebar:** "Dashboard" and a list of tools:
 - Launcher
 - Package Manager
 - User Manager
 - Backup
 - Settings
 - Logout admin
- Center:** A grid of icons representing various applications:
 - eXide (highlighted with a red box)
 - eXist-db Documentation
 - Mittelalterliche Rezepte (highlighted with a red box)
 - Monex
 - Shakespeare's Plays (TEI Publisher Edition)
 - TEI Publisher
 - Van Gogh Letters
 - XQuery Function Documentation

5. Export

Um die App nun endgültig zu exportieren und dann auf einem Webserver, auf dem eXist-db installiert ist, für die Öffentlichkeit zugänglich zu machen, laden wir die App als XAR-Datei herunter.

- Wir öffnen dafür das Dropdown-Menü unter **Administration** in der Navigationsleiste und wählen "Download App as .xar".

The screenshot shows a digital manuscript page from the University Library of Augsburg. A context menu is open over some text, with the option 'Download App as .xar' highlighted. To the right, a thumbnail of another page from the same manuscript is visible.

- In einer anderen eXist-db-Installation können wir die XAR-Datei anschließend über den PackageManager hochladen.

The screenshot shows the eXist-db PackageManager interface. A file named 'ma-rezepte-23.xar' is being uploaded to a 'DROPZONE - put your package(s) here.' section. Below this, under 'Installed (14)', is a card for 'eXide - XQuery IDE' version 3.5.0. Under 'Available (78)' is a card for 'Dashboard' version 2.0.9.

- Wenn wir daraufhin zum eXist-Dashboard wechseln, steht die App zu unserer Edition dort zur Verfügung.

The screenshot shows the eXist-db Dashboard interface. On the left is a sidebar with navigation links. In the main area, there are several application cards. One card for 'Mittelalterliche Rezepte' is highlighted with a red box. Other cards include 'eXide - XQuery IDE', 'eXist-db Documentation', 'Van Gogh Letters', 'XQuery Function Documentation', 'Monex', 'Shakespeare's Plays (TEI Publisher Edition)', and 'TEI Publisher'.

→ Hier der Link zur [XAR-Datei unseres Beispielprojekts auf Github](#).

Kontakt

Unternehmensgröße: (nicht eruierbar)

Weblink: <https://teipublisher.com>

[Github Issues](#)

[Slack-Community](#) (inkl. Channel für Workshops)

Ressourcen

Dokumentation

- [Offizielle TEI-Publisher-Dokumentation](#)
- [Webkomponenten-API](#)
- [XPath](#)
- [Getting started with P5 ODDs](#)

Tutorials

- [Customising TEI, ODD, Roma](#)
- [Youtube - Learn *TEI Publisher* - Session 1 of 3](#)
- [Youtube - Learn *TEI Publisher* - Session 2 of 3](#)
- [Youtube - Learn *TEI Publisher* - Session 3 of 3](#)
- [Learn *TEI Publisher* From Scratch \(Workshop Material\)](#)

Projekte, die dieses Tool genutzt haben

- [Escher Briefedition](#)
- [Sammlung Schweizerischer Rechtsquellen online](#)

Literatur

- Bastianello, E. (2022). Digital Editions at the Biblioteca Hertziana. *Journal of Art Historiography*, 27s. <https://arthistoriography.wordpress.com/27s-dec22/>
- Chagué, A., Scheithauer, H., Terriel, L., Chiffolleau, F., & Tadjo-Takianpi, Y. (2022, July 25). *Take a sip of TEI and relax: a proposition for an end-to-end workflow to enrich and publish data created with automatic text recognition*. Digital Humanities 2022: Responding to Asian Diversity. <https://inria.hal.science/hal-03739767>
- Chiffolleau, F., Baillot, A., & Ovide, M. (2021, October). A TEI-based publication pipeline for historical egodocuments - the DAHN project. *Next Gen TEI, 2021 - TEI Conference and Members' Meeting*. <https://hal.science/hal-03451421>
- Chiffolleau, F., & Scheithauer, H. (2022, September). From a collection of documents to a published edition: how to use an end-to-end publication pipeline. *TEI 2022 - Text Encoding Initiative 2022 Conference*. <https://hal.science/hal-03780316>
- Kränzle, A., Ritter, G., & Sieber, C. (2023). Sources Online: Eine nachhaltige Infrastruktur für digitale wissenschaftliche Texteditionen auf der Grundlage von *TEI Publisher* und IIIF: Sources Online: A Sustainable Infrastructure for Digital Scholarly Text Editions Based on *TEI Publisher* and IIIF. *ABIT Technik*, 43(3), 158–167. <https://doi.org/10.1515/abitech-2023-0030>
- Kumar, A., Bia, A., Holmes, M., Schreibman, S., Siemens, R., & Walsh, J. (2004). : Bridging the Gap Between a Simple Set of Structured Documents and a Functional Digital Library. In R. Heery & L. Lyon (Eds.), *Research and Advanced Technology for Digital Libraries* (pp. 432–441). Springer. https://doi.org/10.1007/978-3-540-30230-8_39
- Kumar, A., Schreibman, S., Stewart, ArNeil, Holmes, M., Bia, A., & Walsh, J. (2005). : A Repository Management System for TEI Documents. *Literary and Linguistic Computing*, 20(1), 117–132. <https://doi.org/10.1093/lrc/fqh047>

- Morlock, E. (2017, October 6). *TeiPublisher for EpiDoc*. Visible Words: Digital Epigraphy in a Global Perspective, Providence, United States. <https://shs.hal.science/halshs-01773203>
- Scheithauer, H., Chagué, A., & Romary, L. (2021). *From eScriptorium to TEI Publisher*. Brace your digital scholarly edition! <https://inria.hal.science/hal-03538115/document>

Factsheet

System	
Scope des Tools	Publikation
Softwareumgebung/Softwaretyp (Remotesystem im Browser / Lokaler Client)	lokale Browser-Anwendung
Unterstützte Plattformen	Unix, Windows & Mac
Geräte	Desktop
Einbindung anderer Systeme (Interoperabilität)	<input checked="" type="checkbox"/>
Accountsystem	<input checked="" type="checkbox"/> Hinzufügen von mehreren Usern über eXist möglich
Kostenmodell (Kostenübersicht/Open Source)	TEI Publisher & eXist-db: kostenlos
Anforderungen & Methoden	
Erforderte Code Literacy	fortgeschritten
Interface-Sprachen (ISO 639-1)	de, en + 18 weitere Sprachen
Unterstützte Zeichenkodierung	[keine Angaben]
Inkludierte Datenkonvertierung (Im Preprocessing mögliche Anpassung der Daten an für die Software erforderliches Format)	<input checked="" type="checkbox"/>
Abhängigkeit von anderer Software (Falls ja, wird diese Software automatisch mitinstalliert?)	<input checked="" type="checkbox"/> eXist-db ist eigenständig zu installieren
Erforderliche Plug-Ins (bei web-basierten Anwendungen)	[nicht anwendbar]
Dokumentation & Support	
Wartung und ständige Erweiterung	<input checked="" type="checkbox"/>
Einbindung der Community	<input checked="" type="checkbox"/> via GitHub
Dokumentation	<input checked="" type="checkbox"/>

Dokumentationssprache	Englisch
Dokumentationsformat	HTML
Dokumentationsabschnitte	Quickstart, Supported XML Vocabularies, Processing Model Transformations, CSS Styling, Page Templates and pb-components, API, Best Practice Recommendations, Creating & Updating Apps, Facet Search Configuration, Creating Custom Web Components, Adding custom vocabulary, Embedding TEI Publisher in other systems, Annotating Documents, Configuring the Annotation Editor, TEI Publisher in Production
Verfügbarkeit von Tutorials	<input checked="" type="checkbox"/> Youtube-Videos für Nutzer:innen inkl. GitHub Workshop-Material
Aktiver Support/Community \ (Forum, Slack, Issue Tracker etc.)	<input checked="" type="checkbox"/> Slack Channel, Mailingliste, Github-Issue-Mechanismus
Nutzbarkeit & Nachhaltigkeit	
Installationsablauf	fortgeschrittene Kenntnisse nötig
Test (Gibt es ein Test Suite, um zu überprüfen, ob die Installation erfolgreich war?)	<input checked="" type="checkbox"/>
Lizenz, unter der das Tool veröffentlicht wurde	GNU GPL 3.0
Registrierung in einem Repository	<input checked="" type="checkbox"/> Github
Möglichkeit zur Software-Entwicklung beizutragen	<input checked="" type="checkbox"/>
Benutzerinteraktion & Benutzeroberfläche	
Benutzerprofil (erwartete Nutzer:innen)	GeWi-Institutionen, Forschende und Editor:innen als Tool-Nutzende
Benutzerinteraktion (erwartete Nutzung)	Hochladen von XML-Dateien, Dateimanagement (über eXide), Publikationsaufbereitung von XML-Dokumenten, Erstellung einer App
Benutzeroberfläche	browserbasiertes GUI
Visualisierungen (Analyse-, Input-, Outputkonfigurationen)	<input checked="" type="checkbox"/>
Benutzerverwaltung	
Personenverwaltung	<input checked="" type="checkbox"/> inklusive Rollenzuweisung (über eXist)
Interne Kommunikationsmöglichkeiten	<input checked="" type="checkbox"/>

(z. B. Annotationsrichtlinien,
Kommentarfunktionen, ...)

Daten- und Toolverwaltung

Zentrale/dezentrale Verwaltungsmöglichkeit	<input checked="" type="checkbox"/>
Versionskontrolle	<input type="checkbox"/>
Projektspezifische Einstellungen	<input checked="" type="checkbox"/>
API	<input checked="" type="checkbox"/>
Möglichkeit auf simultanes Arbeiten	<input checked="" type="checkbox"/>

Datenupload

Unterstützte Dateiformate	Diverse XML-Formate: TEI, DocBook, MSWord (DOCX), JATS
Informationen zur Datensicherheit	[nicht anwendbar, da Datenbank und Daten auf einem von der Projektleitung selbst gewählten Server gespeichert sind]
Zugänglichkeit von verschiedenen Standorten/Geräten	<input checked="" type="checkbox"/>
Einschränkungen hinsichtlich der Datenmenge	[keine Angaben]
Verlustfreier Upload von bereits bearbeiteten Dokumenten	<input checked="" type="checkbox"/>
Unterstützung von IIIF-Import	<input type="checkbox"/> ABER: eigene Webkomponente für die Anzeige von Faksimiles über den IIIF-Server

Datenbearbeitung (Publikationstool)

Komplexitätsgrad bei der Publikationsvorbereitung (z. B. Verfügbarkeit von Buttons, Drag&Drop-Funktion, ...)	ODD-Editor und App-Erzeugung auf Formularbasis
Standardeinstellungen entsprechend bestimmten Standards für Digitale Editionen	<input checked="" type="checkbox"/> TEI, XML, HTML
Anpassungsmöglichkeit und Validierung	
entsprechend projektspezifischen Konventionen/Schemata	<input checked="" type="checkbox"/>
Metadaten-Anreicherung	<input checked="" type="checkbox"/> über eXide möglich

Ansichtsmöglichkeiten

(z. B. Bearbeitungsansicht,
Synopsen-Ansicht, Vorschauansicht
...)

Publikationsvorschau

Unterstützte Dateiformate

Für Publikation: HTML, EPUB, PDF, LaTex

Allgemein: XML, ODD, CSS, XAR

Datenexport

Datenverlust

(nicht vollständiger Erhalt von
Annotationen, die bereits vor
Verwendung des Tools gemacht
wurden)

**Validierungsmöglichkeit für TEI-
XML vor Export**

über eXide möglich

**Datenaufbewahrung nach
Export**

[nicht anwendbar, da Datenbank und Daten auf einem von der
Projektleitung selbst gewählten Server gespeichert sind]