



netidee

PROJEKTE

faircommons Projekt
fairregister Service

Endbericht | Call 16 | Projekt ID 5867

Lizenz CC-BY-3.0 AT

fairkom Gesellschaft

Inhalt

1	Einleitung	3
2	Projektbeschreibung	3
3	Verlauf der Arbeitspakete	4
3.1	Arbeitspaket 1 - Projektstart	4
3.2	Arbeitspaket 2 - User Stories	4
3.3	Arbeitspaket 3 - Blockchain Technologie bzw. IPFS	5
3.4	Arbeitspaket 4 - Backend	6
3.5	Arbeitspaket 5 - Frontend	10
3.6	Arbeitspaket 6 - nextcloud App	13
3.7	Arbeitspaket 7 - Migration	14
3.8	Arbeitspaket 8 - Projektende	14
4	Umsetzung Förderauflagen	15
5	Liste Projektendergebnisse	15
6	Verwertung der Projektergebnisse in der Praxis	15
7	Öffentlichkeitsarbeit/ Vernetzung	16
8	Eigene Projektwebsite	17
9	Geplante Aktivitäten nach netidee-Projektende	17
10	Anregungen für Weiterentwicklungen durch Dritte	17

1 Einleitung

Digitale Werke wie Texte, Bilder, Musik oder Videos werden gerne online geteilt und gegebenenfalls einer Community für die weitere Nutzung zur Verfügung gestellt. Für den / die Urheber:in kann es dabei wichtig sein, dass der Name bei der Wiederverwendung des Werkes genannt wird und das Werk lediglich unter bestimmten Bedingungen verwendet werden darf. Umgekehrt kann es für den / die Nutzer:in eines Werkes von Bedeutung sein, dass die Rechte für das Werk entsprechend geklärt sind und Rechtssicherheit für die weitere Verwendung besteht.

Wie können digitale Werke registriert und die Urheberschaft zum jeweiligen Zeitpunkt nachgewiesen werden? Wie können Nutzer:innen sicherstellen, dass ein Werk rechtssicher verwendet werden darf? Das faircommons Projekt und der daraus resultierende Service fairregister bieten einen Lösungsansatz.

2 Projektbeschreibung

fairregister geht als Service aus dem von der netidee geförderten Projekt faircommons hervor und stellt die Neuentwicklung von RegisteredCommons dar.

RegisteredCommons diente als bisheriger Webservice dazu, Eigentumsverhältnisse an digitalen Werken zu einem bestimmten Zeitpunkt nachweisen zu können. Das Werk konnte mit einer entsprechenden Lizenz, vorwiegend Creative Commons (CC) Lizenzen, versehen und so die Nutzungsbedingungen festgelegt werden. RegisteredCommons wurde überwiegend von der weltweiten Creative Commons Community genutzt.

Im Rahmen des Aktivitätsfeldes Commons betreut fairkom seit 2004 Creative Commons in Österreich. Im Jahr 2006 wurde RegisteredCommons als ergänzender Service in Berlin zur WizardOfOs Konferenz vorgestellt.

Mit dem faircommons Projekt wurde RegisteredCommons als fairregister neu entwickelt. Dazu ist der Registrierungsprozess überarbeitet und vereinfacht worden. Zusätzlich ist geplant, den Webservice als Integration für die nextcloud anzubieten, so dass Werke direkt aus der nextcloud registriert werden können. Für die Lizenzierung der Werke stehen Creative Commons und gängige Open Source Lizenzen zur Verfügung.

Der Hashwert der Datei und die Metadaten werden standardmäßig in der fairregister Datenbank abgespeichert sowie in ein InterPlanetary File System (IPFS) als unveränderliche Datenbank abgelegt. Als digitaler Service für die Registrierung von urheberrechtlich geschützten Werken, dient so fairregister für die Urheber:innen als Nachweis über die Urheberschaft im Falle eines Plagiats oder Missbrauchs.

Ziel von fairregister ist, Rechtssicherheit für Werke mit einer bei der Registrierung ausgewählten Lizenz zu gewährleisten. Mehr Werke sollen als Commons definiert und der

Allgemeinheit zugänglich gemacht werden. fairregister soll sich langfristig als Verzeichnis für offen lizenzierte Werke etablieren.

Im Hinblick auf die Verwendung von fairregister sind zwei grundlegende Perspektiven zu beachten.

Einerseits gibt es die Urheber:innen. Der / die Urheber:in möchte das Werk unter einer entsprechenden Lizenz registrieren, um die Urheberschaft zu einem bestimmten Zeitpunkt nachweisen zu können.

Andererseits gibt es die Nutzer:innen. Der / die Nutzer:in möchte nach lizenzierten Werken suchen, um diese rechtssicher für beispielsweise die eigene Arbeit verwenden zu können.

Daraus ergeben sich zwei Anforderungen an den Service, die Lizenzierung von Werken und die Suche nach lizenzierten Werken.

3 Verlauf der Arbeitspakete

3.1 Arbeitspaket 1 - Projektstart

Das Projekt faircommons ist im Dezember 2021 erfolgreich gestartet. Die erforderlichen Unterlagen wurden erstellt, unterschrieben und abgenommen. Das Projekt wurde geplant und die angestrebten Projektergebnisse wurden festgehalten.

3.2 Arbeitspaket 2 - User Stories

Die User Stories sind teilweise aus dem bisherigen Service abgeleitet worden. Ebenfalls wurde versucht auf Wünsche der Creative Commons Community einzugehen. Ergänzt wurden die bestehenden User Stories durch neue Anforderungen.

Bezüglich der User Journey gibt es zwei Möglichkeiten für eine:n Nutzer:in ein Werk zu lizenzieren. Der Zugang zu dem Service kann über den Webdienst oder die nextcloud erfolgen. Voraussetzung ist, dass die fairregister Integration in der nextcloud vorhanden ist.

Für die Lizenzierung des Werkes ist eine entsprechende Lizenz auszuwählen. Hierzu werden CC Lizenzen und gängige Open Source Lizenzen vorgeschlagen. Für die Zuordnung der Lizenzen soll SPDX als internationaler, offener Standard verwendet werden. SPDX ermöglicht die Verwendung eines Identifikators für mehrere Standardlizenzen.

Nach Auswahl einer Lizenz wird aus der Datei sowie deren Metadaten ein Hashwert erzeugt und dieser im IPFS gespeichert. Das IPFS (InterPlanetary File System) dient dabei als unveränderliche Datenbank.

Lizenzierte Werke können einem breiten Publikum zugänglich gemacht werden, ohne Sorge um den Nachweis der Urheberschaft haben zu müssen.

Im Hinblick auf die User Stories gibt es zwei grundlegende Perspektiven.

Der / die Urheber:in möchte das Werk unter einer entsprechenden Lizenz registrieren, um den Zeitpunkt der Urheberschaft mittels ausgestellttem Zertifikat rechtssicher nachweisen zu können.

Der / die Nutzer:in möchte nach lizenzierten Werken suchen, um diese rechtssicher für beispielsweise die eigene Arbeit verwenden zu können.

Daraus ergeben sich zwei Anforderungen an den Service, die Lizenzierung von Werken und die Suche nach lizenzierten Werken.

Wünsche von der Community, wie beispielsweise das Design eines handgefertigten Produktes zu schützen, sollen durch die Option ein Foto des Werkes hochladen zu können, ermöglicht werden.

Ebenfalls wurde der Use Case einer Defensivpublikation thematisiert. Eine Defensivpublikation, auch Sperrveröffentlichung genannt, beinhaltet die gezielte Veröffentlichung potenziell patentfähiger Inhalte. Durch die Publikation erlischt die Patentierbarkeit, da die Erfindung bereits bekannt und somit nicht mehr neu ist. Eine Defensivpublikation kann dadurch die dauerhafte freie Nutzung einer Erfindung sichern. Dies ist insbesondere für Open Hardware ein Thema, hierzu gab es Gespräche mit der Open Source Ecology Group Germany, welche für Open Hardware die DIN Norm 3105 entwickelten.

fairregister soll als Service gezielt Anwendungsfälle aufgreifen, welche freie Inhalte schützen und fördern.

3.3 Arbeitspaket 3 - Blockchain Technologie bzw. IPFS

Ursprünglich war geplant, den Hashwert des Werks und der Metadatei auf der Blockchain zu speichern. Zu diesem Zweck wurden verschiedene Blockchains untersucht. Nach den Recherchen haben wir uns jedoch entschieden, die Metadatei und das Werk (optional) ausschließlich im IPFS (InterPlanetary File System) zu speichern. Diese Entscheidung wurde aus folgenden Gründen getroffen.

1. Skalierbarkeit: Skalierbarkeit ist ein zentrales Problem der Blockchain-Technologie. Da jeder Knoten in der Blockchain alle Transaktionen speichern muss, kann die Skalierbarkeit stark von der Anzahl der Transaktionen und Teilnehmer:innen beeinflusst werden. Im Gegensatz dazu bietet das IPFS-Protokoll eine verbesserte Skalierbarkeit, da Dateien über das Netzwerk verteilt werden und die Last nicht auf jedem einzelnen Knoten liegt. Dadurch können mehr Teilnehmer:innen und größere Datenmengen effizienter verarbeitet werden.
2. Effizienz: Die Blockchain erfordert oft einen hohen Rechenaufwand, um einen Konsens über den Zustand der Daten zu erzielen. Dies geschieht typischerweise durch Konsensalgorithmen wie Proof-of-Work (PoW) oder Proof-of-Stake (PoS). Im Vergleich dazu ermöglicht das IPFS-Protokoll eine effizientere Speicherung und Abfrage von Dateien, ohne dass komplexe Konsensalgorithmen verwendet werden müssen.

3. Transaktionskosten: In der Blockchain, insbesondere in öffentlichen Netzwerken wie Bitcoin oder Ethereum, fallen für jede Transaktion Transaktionsgebühren an. Diese Gebühren können je nach Netzwerklast und gewünschter Transaktionspriorität variieren. Im Laufe der Zeit können die Transaktionskosten erheblich ansteigen, was die Nutzung der Blockchain für kleinere Transaktionen oder häufige Interaktionen unwirtschaftlich macht. Durch die Verwendung des IPFS entfallen diese Transaktionskosten, da das Protokoll keine Transaktionen im herkömmlichen Sinne erfordert.

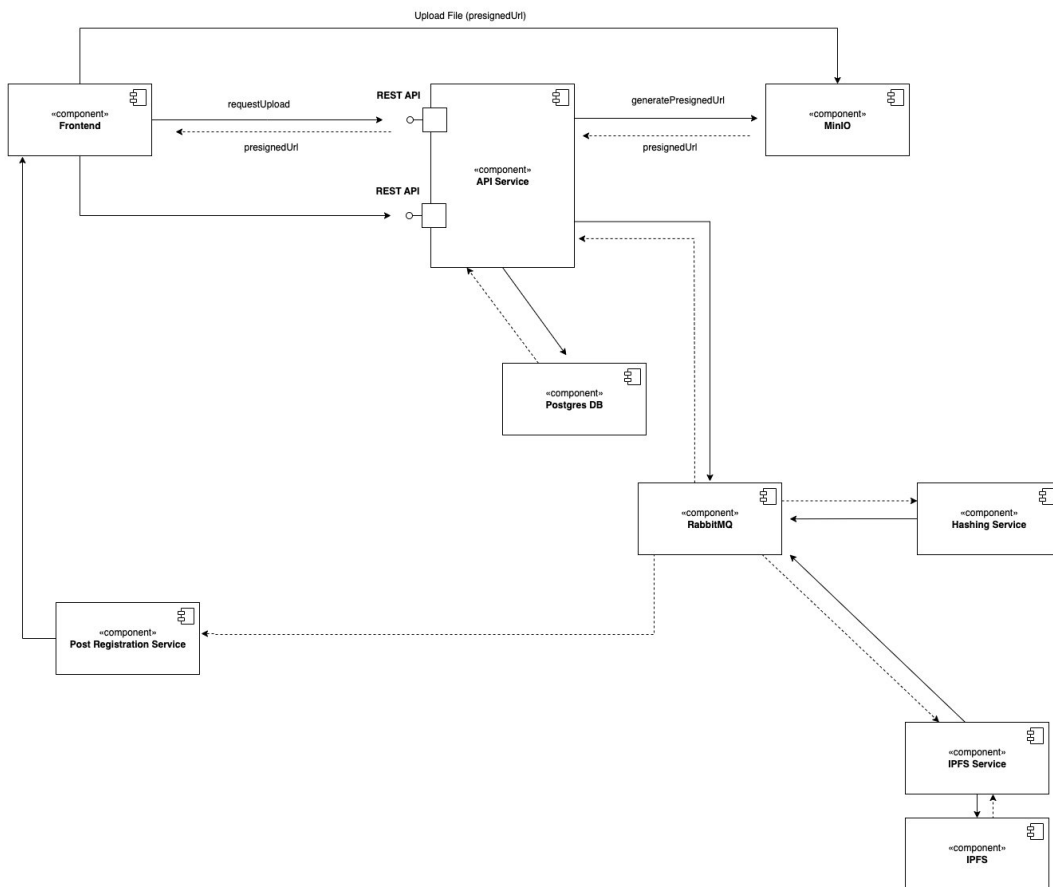
Aufgrund der besseren Skalierbarkeit, der höheren Effizienz und der geringeren Transaktionskosten haben wir uns daher entschieden, das Metafile und das Werk ausschließlich im IPFS als unveränderliche Datenbank zu speichern. Der Hashwert der Datei und die Metadaten werden standardmäßig in der fairregister Datenbank abgespeichert sowie im IPFS abgelegt.

Wir hatten als optionales Feature überlegt, ein Minting von Non-Fungible Tokens (NFTs) mit anzubieten. Nachdem der Hype darum aber wieder rasch abgeflaut ist, beobachten wir einstweilen noch deren Entwicklung und ergänzen diese Möglichkeit ggf. zu einem späteren Zeitpunkt. Nach der Registrierung eines Werkes soll den Nutzer:innen dann optional angeboten werden, aus dem Werk ein NFT zu minten. Hierfür sollen den Nutzer:innen mehrere Blockchains als Auswahl zur Verfügung stehen. Der / die Nutzer:in kann wählen, welche Blockchain für die Sicherung des Hashs verwendet werden soll. Je nach Blockchain, beispielsweise Ethereum oder Solana, variieren die Gebühren für das Minting. Dies bietet den Vorteil, dass der / die Nutzer:in die Minting-Gebühr selbst beeinflussen und gegebenenfalls eine bereits vorhandene Kryptowährung verwenden kann.

Als Krypto-Wallet wurde Metamask evaluiert. Metamask ist ein Open-Source-Ethereum-Wallet. Alle Arten von Ethereum-basierten Token werden unterstützt. Das Wallet lässt sich flexibel von Plattform zu Plattform übertragen. Metamask ist als Erweiterung im Browser zu finden und kann für die jeweilige Plattform aufgerufen werden. Non-Fungible Tokens (NFTs) können darin ebenfalls zu dem Wallet hinzugefügt und verwaltet werden.

3.4 Arbeitspaket 4 - Backend

Die folgende Darstellung bildet die Architektur des Service ab:



Api-Service

Der Api-Service stellt eine REST-API-Schnittstelle bereit, die anderen Anwendungen, wie fairregister und nextcloud, den Zugriff auf seine Funktionen ermöglicht. Der Api-Service bietet verschiedene REST-API-Schnittstellen, über die andere Anwendungen wie fairregister und nextcloud auf seine Funktionen zugreifen können. Er ist auch dafür verantwortlich, Events an die richtigen Queues zu senden und den aktuellen Registrierungsstatus zu schreiben.

RabbitMQ-Service

Der RabbitMQ Service ist ein Open Source Message Broker, der als Vermittler zwischen den verschiedenen Microservices fungiert und Nachrichten über eine Message Queue weiterleitet. Der Einsatz von RabbitMQ ermöglicht eine lose Kopplung zwischen den Microservices und erhöht die Skalierbarkeit der Anwendung.

Hashing-Service

Der Hashing-Service ist dafür verantwortlich, einen kryptographischen Hashwert des Werks und der Metadaten zu erstellen. Dies erhöht die Sicherheit und Integrität der Daten, da Änderungen an den Daten sofort erkannt werden können.

MinIO-Service

Ist eine objektbasierte Open-Source-Storage-Software, die für die Speicherung und den Zugriff auf unstrukturierte Daten wie Bilder, Videos oder Dokumente verwendet wird.

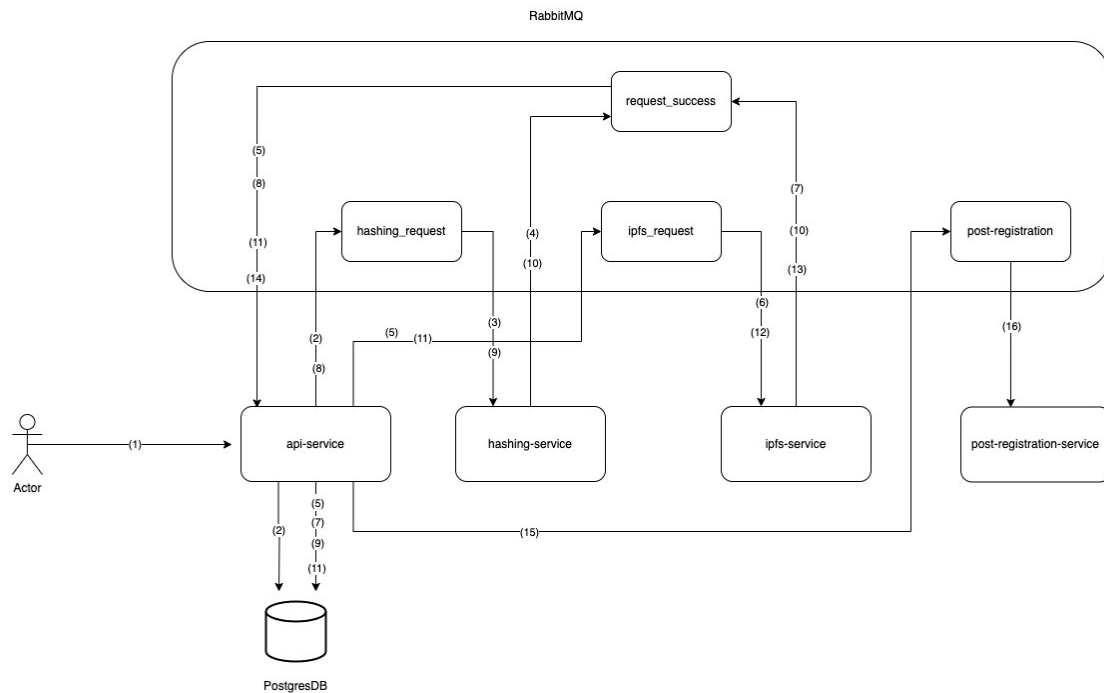
IPFS-Service

Implementiert das IPFS-Protokoll, um das Hochladen und Herunterladen von Dateien über das IPFS-Netzwerk zu ermöglichen.

Post-Registration-Service

Der Dienst ist für die Erstellung des Registrierungszertifikats verantwortlich und sendet dem / der Benutzer:in eine E-Mail, die bestätigt, dass das Werk erfolgreich registriert wurde.

Ablauf einer Registrierung:



1. Der / die Benutzer:in möchte ein Werk registrieren und ruft den Api-Service auf.
2. Der Api-Service erstellt ein neues Werk in der PostgresDB mit dem Status "PENDING" und veröffentlicht ein neues Ereignis in der hashing_request Queue, um das Werk zu hashen.
3. Der Hashing Service konsumiert das Ereignis aus der hashing_request Queue und hasht das Werk.

4. Anschließend veröffentlicht der Hashing-Service ein neues Ereignis in der request_success Queue.
5. Der Api-Service konsumiert das Ereignis aus der request_success Queue und aktualisiert das Werk mit dem neuen Status und dem Werk-Hash. Ein neues Ereignis wird dann in der ipfs_request Queue veröffentlicht.
6. Der ipfs-Service konsumiert das Ereignis aus der ipfs_reqest Queue und speichert das Werk im IPFS.
7. Nachdem das Werk gespeichert wurde, veröffentlicht der ipfs-Service ein neues Ereignis in der request_success Queue.
8. Der Api-Service konsumiert das Ereignis aus der request_success Queue und aktualisiert das Werk mit dem neuen Status und dem CID des Werks. Anschließend wird ein neues Ereignis an hashing_request gesendet.
9. Der Hashing-Service nimmt das Ereignis aus der hashing_request Queue und hasht die Metadaten.
10. Der Hashing-Service veröffentlicht dann ein neues Ereignis in der request_success Queue.
11. Der API-Service konsumiert das Ereignis aus der request_success Queue und aktualisiert das Werk mit dem neuen Status und dem Hash der Metadaten. Ein neues Ereignis wird dann in der ipfs_request Queue veröffentlicht.
12. Der ipfs-Service konsumiert das Ereignis aus der ipfs_request Queue und speichert die Metadaten-Datei im IPFS.
13. Der ipfs-Service veröffentlicht dann ein neues Ereignis in der request_success Queue.
14. Der API-Service konsumiert das Ereignis aus der request_success Queue und aktualisiert das Werk mit dem neuen Status und dem CID der Metadaten.
15. Der API-Service veröffentlicht ein neues Ereignis in der post_registration-Queue.
16. Der post_registration Service konsumiert das Ereignis aus der post_registration Queue, erzeugt ein Registrierungszertifikat und sendet eine E-Mail an den / die Benutzer:in.

Metadaten:

Als Metadaten wurden folgende Attribute definiert. Diese werden im IPFS gespeichert und können nicht geändert werden.

title (String): Der Titel des Werks.

description (String): Eine Beschreibung des Werks.

creationDate (ZonedDateTime): Das Datum und die Uhrzeit, wann das Werk erstellt wurde.

registrationDate (ZonedDateTime): Das Datum und die Uhrzeit, wann das Werk registriert wurde.

registrar (String): Der Registrar, der das Werk registriert hat.

workFile (WorkFileView): Enthält den Dateinamen und den Werk Präfix.

hashFile (HashCode): Ein Hashcode-Objekt, das den Hash-Wert des Werks und den verwendeten Algorithmus enthält.

hashMeta (HashCode): Ein Hashcode-Objekt, das den Hash-Wert der Metadaten und den verwendeten Algorithmus enthält.

ipfsFile (IpfsEntry): Ein IPFS-Eintrag, der den CID (Content Identifier) und den Pfad zur Datei enthält.

ipfsMeta (IpfsEntry): Ein IPFS-Eintrag, der den CID (Content Identifier) und den Pfad zu den Metadaten enthält.

licenseIdentifier (String): Der Lizenzbezeichner für das Werk.

3.5 Arbeitspaket 5 - Frontend

Im Hinblick auf das Frontend stellte die Definition der Metadaten einen intensiven Prozess dar. Zunächst wurden die Metadaten des bisherigen Services evaluiert und überprüft, welche Daten für den neuen Service übernommen werden. Zudem sind weitere Felder hinzugekommen, um beispielsweise neue Use Cases abzudecken.

Eine ausführliche Recherche zu Safe Creative (www.safecreative.org) half dabei, den Registrierungsprozess zu veranschaulichen und in einzelne Schritte zu gliedern. Innerhalb eines Interviews mit Mario Pena, COO von Safe Creative, konnten grundlegende Fragen geklärt und Safe Creative als Best Practice kennengelernt werden.

Das Frontend wurde mithilfe der JavaScript-Bibliothek React und der CSS-Bibliothek Tailwind entwickelt. Die Plattform ist zweisprachig verfügbar, in Deutsch und Englisch. Mehrere Seiten wurden erstellt, darunter die Landingpage, die Werksuche, die Registrierungsseite für Werke und die eigene Werke Seite, zusätzlich gibt es noch eine Einstellungsseite.

Die Landingpage wurde gezielt entworfen, um Besucher:innen anzusprechen. In der Datei "LandingPage.jsx" befindet sich die Komponente namens "LandingPage", welche den gesamten Inhalt der Landingpage enthält. Die Landingpage ist in verschiedene Abschnitte unterteilt, wie zum Beispiel den Hauptkopfbereich und den Features-Bereich. Im Hauptkopfbereich werden der Titel und die Beschreibung der Landingpage angezeigt. Zusätzlich gibt es zwei Schaltflächen, eine für die Registrierung und eine zum Scrollen zu den Features. Im Features-Bereich werden verschiedene Features in einer Liste präsentiert. Jedes Feature ist mit einem Symbol, einem Namen und einer Beschreibung versehen.

Die Registrierungsseite dient dem Hochladen von Werken und befindet sich in der Komponentendatei "RegisterWorkPage.jsx". Da der Registrierungsprozess im Allgemeinen nicht leicht verständlich ist, werden Hilfetexte verwendet, um die Nutzer:innen über bestimmte Bereiche zu informieren. Mit dem useState-Hooks in der Komponente wird überprüft, ob bestimmte Parameter in der URL vorhanden sind. Basierend darauf wird der Datei-Upload-Bereich deaktiviert, falls erforderlich. Dies ist insbesondere für das nächste Arbeitspaket, die "nextcloud App", von Bedeutung.

Die View-Komponente besteht aus mehreren Unterkomponenten, darunter der Datei-Upload-Bereich (registerworkFileUpload.jsx), der Informationsbereich (registerworkInformation.jsx), der Lizenzbereich (registerworkLicence.jsx), die Urheberrechtsbestätigung (registerworkConfirmation.jsx) und die Registrierungsüberprüfung (RegisterCheck.jsx). Um die Übersichtlichkeit zu gewährleisten, sind all diese Komponenten wiederum in kleinere Komponenten unterteilt.

In der Hauptkomponente "RegisterWorkPage.jsx" werden alle Fäden zusammengeführt und der Registrierungsprozess wird über den ApiService.js gestartet. Vor dem Senden des Requests finden jedoch einige Validierungsmechanismen statt, um sicherzustellen, dass keine fehlerhaften Eingaben gemacht wurden. Diese Validierungen beschränken sich darauf, sicherzustellen, dass alle erforderlichen Werte korrekt ausgefüllt sind, um fehlerhafte Requests zu vermeiden. Die Funktion "checkValidation()" überprüft verschiedene Bedingungen, um festzustellen, ob alle erforderlichen Felder korrekt ausgefüllt wurden, und gibt einen Boolean-Wert zurück, der angibt, ob die Validierung erfolgreich war. Wenn alle Eingaben korrekt sind, wird von der Komponente eine Übersichtsseite gerendert, auf der alle Eingaben noch einmal übersichtlich für den Benutzer dargestellt werden. Dadurch wird eine abschließende Überprüfung der Daten ermöglicht, um sicherzustellen, dass sie korrekt sind.

Der/Die Benutzer:in muss sich mittels dem IDM Keycloak anmelden, um eine Datei registrieren zu können. Die Anmeldung erfolgt im Frontend über den UserService, und bietet also Funktionen zum Initialisieren und Verwalten der Keycloak-Authentifizierung, zum Abrufen von Benutzerinformationen und zur Überprüfung von Benutzerrollen. Er verwendet die Keycloak-Bibliothek, um mit einem Keycloak-Server zu interagieren.

Der ApiService ist für die Kommunikation mit einem API-Server zuständig und bietet Funktionen zum Abrufen und Verarbeiten von Daten. Er erleichtert die Kommunikation mit dem API-Server und stellt Funktionen für Datenabfragen und -verarbeitung in der Anwendung bereit. Eine wichtige Funktion des ApiService ist das Hochladen von Dateien. Dafür wird der Endpunkt "api-service/files/upload/presigned-url" genutzt, um eine eindeutige vorab signierte URL für das Hochladen einer Datei zu erhalten. Der Dateiname und die Keycloak-ID werden verwendet, um die URL zu generieren. Anschließend wird die Datei mithilfe dieser URL an den API-Server gesendet. Nachdem die Datei erfolgreich hochgeladen wurde, kann der Benutzer den Registrierungsprozess abschließen, indem er auf den Registrierungsbutton klickt. Dafür wird der Endpunkt "api-service/works/register" verwendet. Dadurch erfolgt die endgültige Registrierung mit allen zusätzlichen Metadaten. Diese Informationen werden dann im Backend weiterverarbeitet.

Der Registrierungsprozess wurde wie folgt umgesetzt: Zu Beginn gibt es den Datei-Upload-Bereich, der in der Komponente "RegisterWorkFileUpload.jsx" zu finden ist. Diese Komponente bietet eine Drag-and-Drop-Funktionalität, um das Benutzererlebnis für Desktop-Anwender:innen zu verbessern. Die Bibliothek "react-dropzone" wird verwendet, um die Drag & Drop-Funktionalität für das Hochladen von Dateien bereitzustellen. Die Komponente enthält auch Zustandsvariablen wie "fileUploaded" und "loading", um den Upload-Status der Datei und den Ladezustand anzuzeigen. Wenn eine Datei erfolgreich hochgeladen wurde, wird ein Container angezeigt, der den Dateinamen und die Dateigröße enthält. Zusätzlich wird ein Schließen-Symbol angezeigt, mit dem die hochgeladene Datei gelöscht werden kann.

Nach dem Hochladen des Werks werden detaillierte Informationen über das Werk abgefragt. Die Benutzer:innen werden aufgefordert, einen Werkstyp auszuwählen, der Dokumente, Bilder, Audio, Video, interaktive Werke, Hardware und Daten umfasst. Um die Benutzerfreundlichkeit zu verbessern und den Werkstyp visuell darzustellen, werden große Schaltflächen und Icons verwendet.

Im Registrierungsprozess können die Benutzer:innen einen Titel, eine Beschreibung und Tags für das Werk hinzufügen. Außerdem besteht die Möglichkeit, die Sprache und das Erstellungsdatum anzugeben. In der Werkinformationssektion werden die Urheber:innen des Werks angegeben. Mindestens ein:e Urheber:in muss angegeben werden, es können jedoch auch mehrere Urheber:innen für das Werk verantwortlich sein. Dabei wird auch die prozentuale Beteiligung angegeben. Die Eingabefelder für die Urheberinformationen folgen der Reihenfolge: E-Mail-Adresse, Vorname, Nachname, prozentuale Beteiligung, Pseudonym und URL der Website des Urhebers. Zusätzliche Felder können für den Herausgeber und/oder eine Identifikationsnummer bereitgestellt werden.

Im Anschluss sind die Lizenzbestimmungen für das Werk zu finden. Dabei werden die SPDX (Software Package Data Exchange)-Kurzbezeichnungen verwendet, die standardisierte Lizenzen darstellen. Es gibt Hunderte von verschiedenen SPDX-Lizenzen mit unterschiedlichen Bedingungen und Einschränkungen. Um die Auswahl zu vereinfachen, wurde beschlossen, zwei zusätzliche Registerkarten hinzuzufügen: "Häufig verwendet" und "CC-Lizenz-Kit". Dadurch ergeben sich insgesamt drei Registerkarten: "Häufig verwendet", "CC-Lizenz-Kit" und "Alle Lizenzen". Die Komponente "registerworkLicence.jsx" rendert eine HTML-Struktur, die die Registerkarten und die entsprechenden Inhalte für jede Registerkarte enthält. Es gibt drei Zustandsvariablen: "shortList", "ccKit" und "longlist", die den aktuellen Status der Registerkarten "Häufig verwendet", "CC-Lizenz-Kit" und "Alle Lizenzen" speichern. Abhängig von der ausgewählten Registerkarte werden die entsprechenden Komponenten ("CommonlyUsedLicences", "CCKitHelper" und "AllLicences") gerendert. In der Komponente "CommonlyUsedLicences" werden basierend auf dem Werktyp verschiedene Lizenzen angezeigt und gefiltert. Die Komponente "CCKitHelper" ist eine React-Komponente, die eine Benutzeroberfläche für die Auswahl von Creative Commons-Lizenzen bereitstellt. Sie zeigt verschiedene Lizenzoptionen an, abhängig von den Benutzereingaben zur Freigabe und kommerziellen Nutzung. Die Komponente importiert

auch Bilder, die die Logos der Creative Commons-Lizenzen repräsentieren. Diese Bilder werden später in der Benutzeroberfläche angezeigt, um die ausgewählte Lizenz visuell darzustellen.

Zu Beginn wird eine schreibgeschützte Checkbox angezeigt, die den Benutzer:innen signalisiert, dass die Metadaten standardmäßig in der fairregister-Datenbank und im InterPlanetary File System (IPFS) bei fairregister gespeichert werden. Das IPFS wird verwendet, um die Metadaten zu sichern und sicherzustellen, dass diese als unveränderliche Daten gespeichert werden. Darunter befinden sich die Optionen für die Speicherung des Werks. Standardmäßig ist das fairregister File System ausgewählt. Es besteht auch die Möglichkeit, das Werk im IPFS zu speichern, wodurch es von dort heruntergeladen werden kann. Es ist wichtig, den Benutzer:innen diese Optionen durch einen Hilfetext zu erklären. Des Weiteren haben Benutzer:innen auch die Möglichkeit, selbst für die Speicherung des Werks verantwortlich zu sein. Wenn die entsprechende Checkbox aktiviert wird, sollte dies deutlich mit einer roten Farbe signalisiert werden. Die letzte Checkbox gibt den Benutzer:innen die Option, das Werk in der Suche für andere nicht anzuzeigen.

Abschließend wird noch eine rechtliche Bestätigung verlangt, in der der/die Benutzer:in bestätigt, das Urheberrecht an dem Werk zu besitzen oder dazu berechtigt ist, es zu registrieren.

Die Seiten für eigene Werke und die Werkssuche sind intuitiv verständlich und einfach zu nutzen. Auf der Seite für eigene Werke haben Benutzer:innen die Möglichkeit, ihre eigenen Werke und Zertifikate herunterzuladen. Auf der Seite für die Suche nach veröffentlichten Werken steht den Benutzer:innen eine leistungsstarke Suchfunktion zur Verfügung, mit der sie gezielt nach bestimmten Werken suchen können. Zusätzlich können Filteroptionen angewendet werden, um die Suche weiter einzuschränken. Dadurch können Benutzer:innen beispielsweise nach einem bestimmten Werktyp filtern, um ihre Ergebnisse zu präzisieren. Um die Benutzerfreundlichkeit und Übersichtlichkeit zu verbessern, wurde eine praktische Paginierung implementiert. Dadurch werden die Suchergebnisse in kleinere Abschnitte aufgeteilt und auf mehrere Seiten verteilt. Benutzer:innen können zwischen den Seiten navigieren, um weitere Werke anzuzeigen, ohne dass alle Ergebnisse auf einer einzigen Seite dargestellt werden müssen. Diese Paginierung trägt dazu bei, die Ladezeiten zu optimieren und eine flüssige Benutzererfahrung zu gewährleisten.

Die Einstellungsseite befindet sich im Benutzerbereich und wird nach einer erfolgreichen Anmeldung per Klick auf den Benutzername in der Navigation sichtbar. Auf der Einstellungsseite haben die Benutzer:innen derzeit die Möglichkeit, die Sprache anzupassen. Zudem gibt es eine Weiterleitung zum Keycloak IDM (Identity Management). Weitere Funktionen auf der Einstellungsseite können ergänzt werden.

3.6 Arbeitspaket 6 - nextcloud App

Die Nextcloud-App "fairregister" ist eine speziell für die Nextcloud-Plattform entwickelte Erweiterung. Mit dieser App können Benutzer:innen Dateien direkt aus der Nextcloud-

Dateiansicht an das fairregister senden. Wenn der/die Benutzer:in die Aktion "Sent to fairregister" auswählt, wird die Datei-ID abgerufen und eine URL generiert. Anschließend wird der Benutzer zur fairregister-App weitergeleitet, wobei die URL die Datei-ID als Parameter enthält. Innerhalb der fairregister-App wird der FairregisterApiController verwendet, um den API-Service anzusprechen. Der Endpunkt "api-service/files/upload/presigned-url" wird genutzt, um eine eindeutige vorab signierte URL für das Hochladen der Datei zu erhalten. Dabei werden der Dateiname und die Keycloak-ID verwendet. Die Datei wird dann mithilfe dieser generierten URL an den API-Server hochgeladen. Sobald die Datei erfolgreich hochgeladen wurde, erfolgt eine Weiterleitung des Nextcloud-Benutzers mittels eines iFrames zur Registrierungsseite. In der fairregister-App kann der/die Benutzer:in die verbleibenden Daten eingeben und den Vorgang abschließen. Da der Dateiapload bereits innerhalb des FairregisterApiController behandelt wird, wird die Dateiapload-Sektion im fairregister Frontend ausgeblendet.

3.7 Arbeitspaket 7 - Migration

Das Projekt faircommons sollte den Service RegisteredCommons.org ablösen. Dieser wurde 2006 bei der WizardOfOs Konferenz in Berlin präsentiert und diente insbesondere der internationalen Creative Commons Community, um tausende Werke zu hinterlegen und für mehr Rechtssicherheit zu sorgen. Der Service RegisteredCommons.org musste im Herbst 2022 aus Sicherheitsgründen aufgrund einer veralteten Software-Bibliothek deaktiviert werden. Der damalige Entwickler wurde beauftragt, einen statischen Dump herzustellen, um weiterhin allfällige Rechtsauskünfte beantworten zu können, auch wenn der ursprüngliche Service schon offline ist. Nun ist es aber auch gelungen, sämtliche Werke und Nutzeraccounts auf den neuen Service fairregister.net zu migrieren. Bisherige Nutzer:innen können sich mit ihrem bestehenden Account anmelden. Nach einer Testphase werden sie über den neuen Service per E-Mail informiert.

3.8 Arbeitspaket 8 - Projektende

Die Projektlaufzeit war ursprünglich bis Ende Januar 2023 angesetzt und wurde dann um drei Monate bis Ende April 2023 verlängert.

Besonders der Prozess der Registrierung sowie die Migration der Werke von RegisteredCommons nach fairregister hat mehr Zeit in Anspruch genommen, als ursprünglich veranschlagt. Im Mai 2023 konnte das Projekt dann zu einem guten Ende geführt werden.

Das Projekt ist ausführlich im fairkom GitLab dokumentiert und alle erforderlichen Unterlagen werden an die netidee übermittelt.

Bei der fairkom Gesellschaft zeichnet sich ab, dass innerhalb des Aktivitätsfeldes Commons weiter an dem Projekt gearbeitet und entsprechende Pakete für die Nutzung des Services zur Finanzierung ausgearbeitet werden.

4 Umsetzung Förderauflagen

In der Fördervereinbarung wurden keine speziellen Förderauflagen festgelegt.

5 Liste Projektergebnisse

1	<p><i>Web-Service: Relaunch von registeredcommons.org als fairregister.net</i></p> <p><i>Der Web-Service soll den Urheber:innen von digitalen Werken ermöglichen, die Eigentümerschaft zu einem bestimmten Zeitpunkt zu belegen. Für das digitale Werk wird eine entsprechende Lizenz gewählt und dieses kann so beispielsweise als Creative Commons von der Community gemäß den Nutzungsbedingungen verwendet werden.</i></p>	AGPL 3.0	http://fairregister.net/
2	<i>Blockchain Technologie / IPFS</i>	MIT	https://git.fairkom.net/hosting/faircommons/-/tree/main
3	<i>Software-Modul: nextcloud App</i>	GPL 3.0	https://git.fairkom.net/hosting/faircommons/-/tree/main
4	<i>Daten-Projektergebnisse als Open Data</i>	CC-BY- 3.0 AT	https://git.fairkom.net/hosting/faircommons
5	<p><i>Dokumentation</i></p> <ul style="list-style-type: none"> • <i>Projektzwischenbericht</i> • <i>Projektendbericht</i> • <i>Entwickler:innen-Dokumentation</i> • <i>Anwender:innen-Dokumentation</i> 	<p>CC-BY- 3.0 AT</p> <p>CC-BY- 3.0 AT</p> <p>CC-BY- SA-3.0 AT</p> <p>CC-BY- SA-3.0 AT</p>	<p>https://www.netidee.at/faircommons</p> <p>https://www.netidee.at/faircommons</p> <p>https://git.fairkom.net/hosting/faircommons/-/wikis/developer-documentation</p> <p>https://git.fairkom.net/hosting/faircommons/-/wikis/user-documentation</p>
6	<i>Zusammenfassung als veröffentlichungsfähiger Einseiter</i>	CC-BY- 3.0 AT	https://www.netidee.at/faircommons
7	<i>Externkommunikation</i>	CC-BY- 3.0 AT	https://www.fairkom.eu/fairregister

6 Verwertung der Projektergebnisse in der Praxis

Der Web-Service fairregister.net steht Urheber:innen zur Verfügung, um dort digitale Werke zu registrieren und somit die Eigentümerschaft am jeweiligen Werk zu einem bestimmten Zeitpunkt anhand des Zertifikats belegen zu können.

Nutzer:innen können nach lizenzierten Werken suchen und diese unter Berücksichtigung der jeweiligen Lizenz rechtssicher für eigene Arbeiten verwenden.

Im Fokus stehen die Lizenzierung von digitalen Werken und die Suche nach lizenzierten Werken. Die Verwendung von fairregister lässt sich aber auch auf weitere Use Cases ausweiten. So kann beispielsweise das Design eines handgefertigten Produktes durch das Hochladen von Fotos oder Plänen geschützt werden. Auch für eine Defensivpublikation kann fairregister genutzt werden, um damit die dauerhafte freie Nutzung einer Erfindung zu sichern.

Die Projektergebnisse lassen sich somit auf vielfältige Weise in der Praxis verwenden und können dank der Mikroservice-Architektur beliebig erweitert werden.

7 Öffentlichkeitsarbeit/ Vernetzung

Austausch mit dem European Cultural Backbone Project, SDEPS (<https://www.sdeps.eu/>). Innerhalb des Projekts soll ein Metadaten-Austausch-Netz unter nichtkommerziellen Plattformen entstehen. Europäische Plattformen sollen sich für die Bereitstellung von freiem Content zusammenschließen und vernetzen. Eine Option könnte sein, freie Inhalte bei fairregister zu registrieren, damit diese dauerhaft frei zugänglich sind und entsprechend der Lizenz verwendet werden.

Vernetzung mit Open Source Ecology Germany e.V., OSEG (<https://www.ose-germany.de/>). Der gemeinnützige Verein hat die Entwicklung und Verbreitung von nachhaltigen Open Source Hardware Projekten zum Ziel. Die Open Source Community unterstützt sich hierbei gegenseitig und Open Source Hardware könnte beispielsweise anhand von Abbildungen bei fairregister lizenziert werden. Eine Kooperationsvereinbarung zwischen fairkom und OSEG wurde aufgelegt.

Weiter fand ein Austausch mit DALICC (<https://www.dalicc.net/>) statt. Zum Thema Rechtklärung von Software- und Daten-Lizenzen lassen sich Synergien zwischen DALICC und fairregister bilden. Wie bereits erläutert, soll für die Zuordnung der Lizenzen SPDX als internationaler, offener Standard verwendet werden. SPDX ermöglicht die Verwendung eines Identifikators für standardisierte Lizenzen, während DALICC auch nicht-standardisierte Lizenzen umfasst. Mit dem License Composer von DALICC können Lizenzen individuell zusammengestellt werden. DALICC könnte als Erweiterung für nicht-standardisierte Lizenzen in fairregister integriert werden.

Zusätzlich hat die netidee Förderung zu einem Austausch mit anderen Projekten beigetragen. Das Projekt Educloud Austria behandelt ein Themengebiet, welches auch bei fairkom Anwendung findet. Gemeinsam wurde diskutiert, wie der Einsatz von freier

Software im Bildungsbereich gestärkt und eine umfassende Lösung angeboten werden kann.

Vom 30. September bis zum 2. Oktober 2022 fand die Bits und Bäume Konferenz 2022 an der Technischen Universität (TU) in Berlin statt. Passend zu den Themen Digitalisierung und Nachhaltigkeit wurde dort das faircommons Projekt vorgestellt und der aktuelle Stand von fairregister präsentiert. Der Fokus der Präsentation lag auf dem Frontend von fairregister. Gemeinsam wurde der Registrierungsprozess diskutiert und weitere Möglichkeiten evaluiert.

fairregister im Schubert Theater Wien - im Andenken an den Internetaktivisten Aaron Swartz.

Aaron Swartz war ein Programmierer und Internetaktivist. Sein Einsatz galt der Netzneutralität sowie dem freien und offenen Zugang zu Wissen. Sein Tod liegt nun zehn Jahre zurück und das Schubert Theater Wien erinnerte mit einer Hommage an sein agiles Leben. Für eine gesellschaftspolitische Aufwertung des Netzes setzte er sich als Co-Founder der Social-News-Website Reddit ein. Im Creative-Commons-Team war er als technischer Entwickler dabei und an der Umsetzung der CC-Lizenzen beteiligt.

Im Andenken an seine Arbeit hat fairkom am 01.02.2023 zum Auftakt des Future Labs 2023 das faircommons Projekt vorgestellt. An dem Abend startete die Projektreihe "Aaron's law", begleitet von einem Future Talk zu "Kunst und Kultur als Menschenrecht und Gemeingut im digitalen Zeitalter".

Im Rahmen dessen wurde ein Video über das faircommons Projekt aufgenommen. Dieses ist hier zu sehen: <https://fair.tube/w/thG3ibDN3Mig3UCjfKYipx>.

8 Eigene Projektwebsite

<https://www.fairkom.eu/fairregister>

9 Geplante Aktivitäten nach netidee-Projektende

Nachdem es uns gelungen ist, die bisherigen Werke auf RegisteredCommons.org und wenn wir es schaffen, eventuell eine weitere österreichische Datenbank mit CC Inhalten in fairregister zu hinterlegen, dann könnten wir versuchen im Herbst 2023 beim Preis des Freien Wissens einzureichen <https://freieswissen.net/>.

Nachdem wir etwas Erfahrungen gesammelt haben mit der Möglichkeit, Werke direkt von einer nextcloud in fairregister zu hinterlegen, wollen wir in der nextcloud Community die fairregister App aktiv bewerben und ggf. bei einschlägigen Events oder etwa bei Linuxdays den Service vorführen.

Weiters wollen wir in der Creative Commons Community fairregister vorstellen.

10 Anregungen für Weiterentwicklungen durch Dritte

Die Mikroservice-Architektur ermöglicht es, auf einfache Weise ergänzende Services einzubinden. Dies können Dienste sein, die etwa Werke aus bestehenden Repositories oder Bibliotheken holen und bei fairregister registrieren. Wir werden in entsprechenden Communities das Projekt vorzeigen und erläutern, wie die Schnittstellen funktionieren. Dadurch kann die Rechtssicherheit für jene, die Werke zur Verfügung stellen, aber auch für jene, die sie nutzen, erhöht werden.

Von NLnet wurden wir bereits angesprochen, fairregister im Rahmen deren NCI Zero Entrust Aktivitäten vorzustellen <https://nlnet.nl/entrust/> um andere Entwickler:innen auf die Möglichkeiten aufmerksam zu machen.