HAW HAMBURG

BACHELOR THESIS
Jazib Ahmed

# Entwurf eines rollenbasierten Berechtigungskonzept für die KI-Plattform Data Fusion Generator

FAKULTÄT TECHNIK UND INFORMATIK
Department Informatik

Faculty of Engineering and Computer Science
Department Computer Science

HOCHSCHULE FÜR ANGEWANDTE WISSENSCHAFTEN HAMBURG
Hamburg University of Applied Sciences

**Jazib Ahmed**

**Thema der Arbeit**

Entwurf eines rollenbasierten Berechtigungskonzept für die KI-Plattform Data Fusion Generator

**Stichworte**

IT-Sicherheit, Identitätsmanagement, Berechtigungsmanagement, Berechtigungskonzept, Rollenbasiertes Berechtigungskonzept, RBAC, Zugriffskontrolle, Künstliche Intelligenz, Datengenerierung, Datensynthese

**Kurzzusammenfassung**

In den letzten Jahren hat der Fortschritt im Bereich des maschinellen Lernens zu zahlreichen Weiterentwicklungen geführt. Allerdings sind hochwertige Datensätze oft knapp. Mit der wachsenden Anwendung von künstlicher Intelligenz in verschiedenen Bereichen steigt der Bedarf an Trainingsdaten für Machine-Learning-Modelle stetig, doch sind diese Daten häufig nur begrenzt verfügbar. Die Künstliche Intelligenz Plattform - Data Fusion Generator soll das Problem lösen, indem es die Generierung solcher Daten für verschiedene Bereiche ermöglicht, insbesondere wenn vorhandene Daten quantitativ unzureichend sind oder bestimmte Eigenschaften wie seltene Anomalien fehlen. Um diese Funktionalität für verschiedene Anwender bereitstellen zu können, legt DaFne fest, welche Benutzer welche Berechtigungen haben und auf bestimmte Ressourcen und Funktionen zugreifen können, um bestimmte Aktionen auszuführen. Dies ist wichtig, da DaFne einen speziellen Anwendungskontext hat, in dem Rechte und Rollen neu ausgearbeitet werden müssen. Es ist daher wichtig zu betonen, dass Rollen im speziellen Kontext von DaFne ein fundiertes Konzept darstellen, um unterschiedliche Anwenderinteressen zu verstehen und zu vereinen. Zudem wird geklärt, in welchen Anwendungsszenarien verschiedene Anwender auf welche Daten und Funktionen zugreifen können. Das Berechtigungskonzept trägt dazu bei, unterschiedliche Benutzer in derselben Rolle voneinander abzugrenzen, damit ihre eigenen Daten jeweils geschützt werden können. Auch die Übertragbarkeit des Konzepts auf ähnliche Probleme, wie zum Beispiel andere Plattformen mit ähnlichen Herausforderungen oder andere datenintensive Anwendungen, wird betont. Diese Arbeit zielt darauf ab, die Konzeption eines entsprechendes rollenbasiertes Berechtigungskonzept innerhalb des spezifischen Kontexts von DaFne zu konzipieren.

**Jazib Ahmed**

**Title of Thesis**

Design of a role-based access control concept for the AI platform Data Fusion Generator

**Keywords**

IT Security, Identity Management, Authorisation Management, Authorisation Concept, Role- Based Authorisation Concept, RBAC, Access Control, Artifcial Intelligence, Data Generation, Data Synthesis

**Abstract**

In recent years, progress in the field of machine learning has led to numerous further developments. However, high-quality data sets are often in short supply. With the growing application of artificial intelligence in various fields, the need for training data for machine learning models is constantly increasing, but this data is often only available in limited quantities. The Artificial Intelligence Platform – Data Fusion Generator aims to solve the problem by enabling the generation of such data for different domains, especially when existing data is quantitatively insufficient or lacks certain features such as rare anomalies. In order to provide this functionality for different users, DaFne defines which users have which authorisations and can access certain resources and functions in order to perform certain actions. This is important because DaFne has a specific application context in which rights and roles need to be reworked. It is therefore important to emphasise that roles in the specific context of DaFne represent a sound concept for understanding and uniting different user interests. It also clarifies in which application scenarios different users can access which data and functions. The authorisation concept helps to differentiate between different users in the same role so that their own data can be protected. The transferability of the concept to similar problems, such as other platforms with similar challenges or other data-intensive applications, is also emphasised. This thesis aims to conceptualise a corresponding role-based authorisation concept within the specific context of DaFne.

# Inhaltsverzeichnis

# Abbildungsverzeichnis

# Tabellenverzeichnis

# 1 Einleitung

Dieses Kapitel führt in die grundlegenden Aspekte und Ziele der Arbeit ein. Die Künstliche Intelligenz Plattform - Data Fusion Generator wird im Folgenden als DaFne-Plattform bezeichnet.

## 1.1 Einführung in das Thema

Angesichts der stetig steigenden Bedrohungen durch cyberkriminelle Angriffe stellt die Bekämpfung von Cyberkriminalität und die Gewährleistung der IT-Sicherheit eine der größten Herausforderungen der heutigen Zeit dar ([40]: S. 1; [17]: S. 5; [22]: S. 1). In diesem Zeitalter der Digitalisierung befindet sich auch die Welt in einem inkrementellen Wandel (vgl. [50]: S. 18). Hierbei steht die digitale Transformation im Mittelpunkt (vgl. [37]: S. 38). Aufgrund der globalen Vernetzung und voranschreitenden Globalisierung stehen Unternehmen und Bildungseinrichtungen mehr denn je unter Druck, die Sicherheit, Vertraulichkeit, Integrität und Verfügbarkeit ihrer IT-Systeme zu gewährleisten. Dies ist von entscheidender Bedeutung, um sich vor digitalen Angriffen, Datenmissbrauch und unbefugtem Zugriff zu schützen sowie im internationalen Markt wettbewerbsfähig zu bleiben. Die in den vergangenen zehn Jahren gestiegenen gesetzlichen Anforderungen an die Datensicherheit haben den Handlungsdruck für Unternehmen weiter erhöht, sodass eine kontinuierliche Anpassung und Verbesserung der Sicherheitsmaßnahmen unerlässlich ist. Auch Bildungseinrichtungen sind von diesen Herausforderungen betroffen, allerdings unterliegen sie spezifischen Einschränkungen und Anforderungen (vgl. [40]: S. 1; [22]: S. 1; [49]: S. 1). Die Einhaltung von Sicherheitsstandards und Normen spielt dabei eine wesentliche Rolle, damit potenzielle Risiken und Bedrohungen minimiert werden können (vgl. [22]: S. 1-4). Diesen Herausforderungen stellt sich unter anderem das Identitäts- und Berechtigungsmanagement, das dazu dient, die Datensicherheit zu gewährleisten und die Sicherheitsstandards in IT-Systemen zu verbessern sowie den Zugang und die Funktionen innerhalb der Systeme zu regeln(vgl. [12]: S. 1; [49]: S. 1,35).

Das Identitätsmanagement befasst sich mit der Verwaltung und Kontrolle von Identitäten in einem IT-System, sodass nur autorisierte Benutzer berechtigt sind Zugriff auf Ressourcen und Informationen zu erhalten (vgl. [12]: S. 1; [49]: S. 35). Dieser Prozess umfasst die Identifizierung des Benutzers durch die Angabe des Benutzernamens, gefolgt von der Authentifizierung durch die Angabe des Benutzerkennworts. Danach ist das Authentifizierungsverfahren essenziell, was für die Überprüfung der Identität des Benutzers zuständig ist (vgl. [49]: S. 31-35, 129-134). Die beiden bekanntesten Mechanismen sind das Single-Sign-On-Prinzip, bei dem der Benutzer mit einem Satz von Anmeldeinformationen den Zugang zum gesamten System erhält (vgl. [8]: S. 157, 291), sowie die 2-Faktor-Authentifizierung, die im Falle von sensiblen Daten in Betracht gezogen wird und nach Vorlage von zwei Beweisen den Zugang zum System gewährt. Beide Nachweise setzen sich aus dem Wissen und dem Besitz zusammen, d.h. aus dem Benutzerkennwort und einem zweiten Faktor (vgl. [42]: S. 155). Zum Schluss folgt die Autorisierung, die den Zugang zu den zustehenden Privilegien und Berechtigungen des erfolgreich nachgewiesenen Benutzers gewährt (vgl. [49]: S. 161-164). Hierfür finden Berechtigungskonzepte Anwendung, da die richtige Vergabe und Verwaltung von Berechtigungen einen grundlegenden Aspekt für die Sicherheit und den Schutz sensibler Informationen und Ressourcen eines Systems darstellen.

## 1.2 Problemstellung und Motivation

Eine weitere Aufgabe, die Unternehmen in Angriff nehmen müssen, ist die Auswahl einer geeigneten Berechtigungsmanagementlösung, da sich vor dem Hintergrund der Digitalisierung und der damit verbundenen verschiedenen Reifegrade der am Markt vorhandenen Wahlmöglichkeiten für viele die Frage stellt, welches Berechtigungskonzept zur Unterstützung am besten geeignet ist (vgl. [37]: S. 1; [49]: S. 176). Berechtigungen begleiten uns ständig in unserem täglichen Leben und sind essentiell, da wir eine Vielzahl von ihnen besitzen. Es sind Berechtigungen, um bestimmte Funktionen und Aufgaben am Arbeitsplatz und im täglichen Leben ausführen zu können (vgl. [49]: S. 7). Bei der Vergabe und Zuweisung solcher Berechtigungen in einem IT-System kommt das Berechtigungsmanagement zum Einsatz, um die Zugriffsrechte auf Ressourcen und Funktionen in einem IT-System zu steuern und zu kontrollieren (vgl. [49]: S. VII). Das Berechtigungsmodell bildet die Grundlage für die Umsetzung des Berechtigungsmanagements. Es stellt sicher, dass Zugriffsrechte gemäß der erforderlichen Sicherheitsrichtlinien vergeben werden, und

beschreibt die Regeln und Mechanismen, nach denen Zugriffsrechte zugewiesen und kontrolliert werden. Darüber hinaus legt es fest, welche Benutzer welche Berechtigungen haben und damit auf bestimmte Ressourcen und Funktionen zugreifen und bestimmte Aktionen ausführen dürfen (vgl. [49]: S. 83).

Die herkömmlichen rechte- und regelbasierten Berechtigungskonzepte haben jedoch ihre Grenzen. In rechte- und regelbasierten Berechtigungsmodellen werden Berechtigungen und Regeln auf individueller Basis vergeben bzw. erstellt, was zu einer unübersichtlichen, komplexeren und schwer zu verwaltenden Berechtigungsstruktur führt. Diese Komplexität kann es erschweren, die erforderliche Flexibilität zu gewährleisten, den Überblick über die Berechtigungen zu behalten und diese effektiv zu verwalten (vgl. [49]: S. 41-44, 81). Die Folgen sind Fehler, Inkonsistenzen und das Entstehen von Sicherheitslücken. Dies bietet potenziellen Angreifern mehr Möglichkeiten, Zugang zu sensiblen Daten und Systemressourcen zu erlangen, was zu erheblichen finanziellen Schäden und sogar rechtlichen Konsequenzen führen kann, in Form von Identitätsdiebstahl durch den Verlust vertraulicher und personenbezogener Kundendaten (vgl. [22]: S. 1). Aus diesen Gründen ist es notwendig, alternative Berechtigungskonzepte zu verwenden, die die Komplexität reduzieren, die Skalierbarkeit verbessern und eine übersichtlichere und flexiblere Verwaltung von Berechtigungen ermöglichen.

Die Technologie im Bereich des Berechtigungsmanagements hat sich weiterentwickelt, um den Anforderungen einer datenabhängigen Wirtschaftswelt gerecht zu werden (vgl. [49]: S. 41-44). In diesem Zusammenhang sind Rollenkonzepte relevant, um den Benutzern anhand ihrer Rollen die notwendigen Verantwortlichkeiten zuzuweisen (vgl. [49]: S. 41-44), insbesondere wenn es im Fall von DaFne noch nicht sicher ist, welche Art von Anwendern letztendlich Zugriff bekommen sollen. Mit Blick auf die Vielfalt der Anwenderinteressen in DaFne bilden Rollen ein fundiertes Konzept, um unterschiedliche Anwenderinteressen zu verstehen und zu bündeln, während sie gleichzeitig die Verwaltung von Berechtigungen vereinfachen, Transparenz gewährleisten und die Effizienz steigern (vgl. [49]: S. 41-44).

Das rollenbasierte Berechtigungskonzept bietet eine flexiblere und anpassungsfähigere Methode zur Verwaltung von Berechtigungen, da die Zuweisung von Berechtigungen auf der Grundlage von Rollen erfolgt. Benutzern werden eine oder mehrere Rollen zugeordnet, die bestimmte Aufgaben oder Funktionen repräsentieren. Dies vereinfacht die Verwaltung von Berechtigungen und verbessert die Konsistenz und Kontrolle. Es ermöglicht eine feinere Granularität bei der Zuweisung von Berechtigungen und bietet eine bessere Kontrolle über den Zugriff auf sensible Informationen und Systemressourcen (vgl. [49]: S.

167-168; [8]: S. 157-165). Zudem ist die Anpassungs- und Wartungsfähigkeit insbesondere für DaFne relevant, wo eine Neuausarbeitung und Anpassung von Rechten und Rollen erforderlich ist, um dem speziellen und neuen Anwendungskontext gerecht zu werden.

Diese Bachelorarbeit wird an der Hochschule für angewandte Wissenschaften Hamburg in der Fakultät Technik und Informatik mit dem Studienschwerpunkt Wirtschaftsinformatik innerhalb des Forschungsprojektes DaFne bei Frau Prof. Dr. Ulrike Steffens verfasst. Der Gegenstand dieser Bachelorarbeit ist die Konzeption eines rollenbasierten Berechtigungsmodells für die DaFne-Plattform und die Erläuterung des ausgewählten Konzepts im Vergleich zu anderen Konzepten. Damit soll aufgezeigt werden, welche Lösung bei welchen Anwendungsszenarien besser geeignet ist und ob bzw. wieso das konzipierte Berechtigungsmodell für die DaFne-Plattform nützlich und sinnvoll ist. Ferner soll aufgezeigt werden, inwieweit das konzipierte Modell auf ähnliche Probleme oder Anwendungskontexte übertragbar ist.

Die Generierung synthetischer Daten ist von praktischer Bedeutung in der Forschung und Entwicklung von KI-Methoden, insbesondere dann, wenn die vorhandenen Daten quantitativ unzureichend sind oder bestimmte Eigenschaften, wie seltene Anomalien, fehlen. Zwar existieren in der Wissenschaft bereits verschiedene Ansätze zur Datengenerierung, diese sind jedoch häufig stark auf einen bestimmten Kontext, wie einen spezifischen Anwendungsfall oder ein KI-Modell, optimiert, um eine hohe Datenqualität zu gewährleisten. Dadurch ist die Anwendbarkeit dieser Methoden in der Praxis oft eingeschränkt und ihre Wirkung begrenzt. Das Forschungsprojekt DaFne zielt darauf ab, die Nutzbarkeit von Datengenerierungsmethoden für KI-Forscher und -Entwickler durch die Entwicklung einer innovativen und flexiblen Datengenerierungsplattform zu verbessern. Diese Plattform soll es ermöglichen, die Ergebnisse der Datengenerierung in verschiedenen Kontexten nutzbar zu machen und ihre Effizienz zu steigern (vgl. [46]: S. 3).

Im weiteren Verlauf werden die Aufgabenstellung und Zielsetzung sowie der geplante Aufbau und Gegenstand der Bachelorarbeit vorgestellt.

## 1.3 Aufgabenstellung und Zielsetzung

Das Ziel dieser Bachelorarbeit ist die Konzeption eines rollenbasierten Berechtigungsmodells für die DaFne-Plattform, um den Zugriff auf Ressourcen und Funktionen auf der Plattform zu regulieren. Durch die Verwendung eines Berechtigungsmodells wird geklärt,

in welchen Anwendungsszenarien welche Anwender auf welche Daten und Funktionen zugreifen können. Zugleich soll das Berechtigungskonzept helfen, unterschiedliche Benutzer in der gleichen Rolle voneinander abzugrenzen, um ihre eigenen Daten zu schützen, soweit das gewünscht oder erforderlich ist. Außerdem wird die Sicherheit des Systems bzw. der Plattform, insbesondere im Sinne von Verfügbarkeit gesteigert und verbessert.

Zunächst wird mithilfe des Requirements Engineering Prozesses eine Anforderungsanalyse durchgeführt, um die Ausgangssituation zu analysieren und die Anforderungen zu ermitteln. Dies dient dazu, ein Verständnis für den aktuellen Stand sowie für die zugrunde liegenden Bedürfnisse und Anforderungen zu erlangen.

Anschließend wird die Auswahl eines geeigneten Berechtigungskonzepts unter Berücksichtigung der ermittelten Anforderungen im Requirements Engineering Prozess getroffen und das entsprechende Berechtigungsmodell konzipiert.

Abschließend lassen sich aus dem konzipierten Berechtigungsmodell Erkenntnisse und Empfehlungen bezüglich der Übertragbarkeit auf ähnliche Probleme oder Anwendungskontexte ableiten sowie Fragen identifizieren, die in der Zukunft behandelt werden sollten.

Um die Zielstellung dieser Arbeit zu erreichen, orientiert sich der Aufbau der Bachelorarbeit an folgenden Kern- bzw. Forschungsfragen:

o Wie können die Anforderungen für das rollenbasierte Berechtigungsmodell der Künstlichen Intelligenz Plattform - Data Fusion Generator mithilfe bewährter Methoden des Requirements Engineerings effektiv erfasst werden?

o Wie ist ein rollenbasiertes Berechtigungsmodell für die Künstliche Intelligenz Plattform - Data Fusion Generator zu gestalten und welche Elemente sowie Aspekte sind bei der Gestaltung erforderlich bzw. zu berücksichtigen?

## 1.4 Aufbau der Arbeit

Im ersten Kapitel der Bachelorarbeit wird die Fachliteratur zum Thema Berechtigungsmanagement und Identitätsmanagement sowie die Projektdokumentation zum Projekt DaFne herangezogen. Die Erläuterung der Themenfelder sowie die Vorstellung des DaFne-Projekts zu Beginn dienen der elementaren Grundlagenbildung für die Bachelorarbeit und sollen den Leser und die Leserin an die Thematik heranführen.

Im zweiten Kapitel der Bachelorarbeit soll die Analysephase mittels Requirements Engineering für die DaFne-Plattform, die im Zusammenhang mit dem Berechtigungsmanagement steht, in zwei Phasen durchgeführt werden. In der Analysephase werden bewährte Anforderungsanalysetechniken wie Personas, User Roles und User Stories eingesetzt. Zunächst wird anhand der vorhandenen Projektunterlagen und Dokumentationen die Ausgangssituation dargestellt. Im nächsten Schritt erfolgt die Anforderungsanalyse, bei der die spezifischen Anforderungen und Bedürfnisse aus dem Projektkontext identifiziert werden. Die Erhebung erfolgt in Zusammenarbeit mit relevanten Stakeholdern aus dem Projektteam.

Im dritten Kapitel soll ein Konzept für das Berechtigungsmodell der DaFne-Plattform erstellt werden, um den Zugriff auf Plattformressourcen und Funktionen gemäß den Benutzerrollen zu regeln. Dies trägt dazu bei, die Daten der Benutzer zu schützen sowie die Sicherheit in Bezug auf die Verfügbarkeit der Plattform zu erhöhen. Die im ersten Kapitel ausgearbeiteten Arten von Berechtigungskonzepten sowie die in diesem Kapitel ausgearbeitete Berechtigungsmatrix anhand der Anforderungen aus dem zweiten Kapitel bilden die Grundlage für die Auswahl eines geeigneten Berechtigungskonzepts in Kapitel drei, was ebenfalls einen Schritt im Vorgehen der Konzeption darstellt.

Abschließend wird im letzten der vier Kapitel eine Schlussfolgerung aus der Konzeption gezogen, die das Vorgehen zur Erfüllung der Anforderungen und Bewältigung der Herausforderungen darlegt, sowie gegebenenfalls Empfehlungen, Erweiterungen, Optimierungen und Verbesserungen für den weiteren Prozess ableitet. Darüber hinaus zeigen die Ergebnisse auf, welche offenen Fragen und Aspekte in Zukunft behandelt werden müssen. Interessant ist auch die Betrachtung der Übertragbarkeit auf ähnliche Probleme oder Anwendungskontexte, beispielsweise auf andere Plattformen mit ähnlichen Herausforderungen oder auf datenintensive Anwendungen.

# 2 Theoretische Grundlagen

In diesem Kapitel werden die Grundlagen dargelegt, die zum Verständnis der Arbeit erforderlich sind.

## 2.1 „DaFne“ - Plattform Data Fusion Generator für die Künstliche Intelligenz

Im folgenden Unterkapitel wird die DaFne-Plattform näher vorgestellt.

### 2.1.1 Projektbeschreibung, Motivation und Aufgabenstellung

Die Entwicklung synthetischer Daten ist besonders wichtig, insbesondere wenn für KI-Methoden keine ausreichend großen Trainingsdaten zur Verfügung stehen. Bisherige Methoden zur Datengenerierung sind oft nur begrenzt übertragbar, da sie auf spezifische Anwendungsbereiche zugeschnitten sind und erhebliche Expertise oder Anpassungen erfordern. Das Ziel von DaFne ist es, die Nutzbarkeit von Datengenerierungsmethoden für KI-Forscher und -Entwickler zu verbessern. Hierfür wird eine innovative, flexible und erweiterbare Plattform zur Datengenerierung geschaffen. Diese Plattform ermöglicht die Entwicklung robuster KI-Modelle, die sich in verschiedenen Anwendungsszenarien einsetzen lassen. Die Generierungsmethoden werden systematisch parametrisiert. DaFne definiert überprüfbare Qualitätskriterien für Methoden und generierte Daten, um ihre verlässliche Anwendung in KI-Anwendungen zu gewährleisten. Die Plattform wird im Anwendungsgebiet der Smart Cities getestet und steht nach Abschluss des Projekts der Öffentlichkeit zur Verfügung, um die Datengenerierung für unterschiedlichste Projekte zu unterstützen. Diese Methoden sind besonders wertvoll, wenn Echtzeitdaten knapp sind oder seltene Anomalien fehlen. Derzeit sind vorhandene Datengenerierungstechniken meist nur in spezifischen Szenarien einsetzbar und daher in ihrer Effektivität begrenzt.

Das Projekt verfolgt drei Hauptziele: Erstens sollen anpassungsfähige Generierungsmethoden entwickelt werden, die robuste und breit einsetzbare Modelle hervorbringen. Zweitens sollen Qualitätsstandards für die generierten Daten festgelegt und überprüft werden, um ihre Verlässlichkeit zu garantieren. Und drittens sollen diese Methoden systematisch angepasst werden, um sie für eine Vielzahl von Anwendungen nutzbar zu machen. (vgl. [46]: S. 3-4).

### 2.1.2 Vorgehensweise

In jüngerer Vergangenheit hat der Bereich des maschinellen Lernens Fortschritte verzeichnet. Allerdings ist der Zugang zu den dafür notwendigen hochwertigen Daten oft beschränkt. Dies gilt auch für Daten, die zur Realisierung von Smart-City-Konzepten beitragen könnten. Obwohl über mobile Anwendungen viele Mobilitätsdaten gesammelt werden, ist es oft erforderlich, aggregierte Daten zur Analyse der Bewegungsmuster in Städten zu verwenden, um beispielsweise die Privatsphäre der Nutzer zu wahren. Bei der Auslegung der Analyseergebnisse ist besondere Sorgfalt geboten, um diesen Herausforderungen zu begegnen (vgl. [46]: S. 4).

Das Projekt nutzt den Fortschritt im Bereich der synthetischen Datengenerierung, um diese Herausforderungen zu meistern. Synthetische Daten haben sich als vielversprechend erwiesen, um KI-basierte Forschung und Entwicklung voranzutreiben, insbesondere in Situationen, in denen keine ausreichenden realen Daten verfügbar sind, Datensätze unausgewogen sind oder der Datenschutz die Weitergabe von Daten einschränkt. Fortschritte in der generativen Modellierung eröffnen das Potential, qualitativ hochwertige Datensätze zu schaffen, die beispielsweise die Entwicklung von KI für Smart-City-Anwendungen erleichtern (vgl. [46]: S. 4).

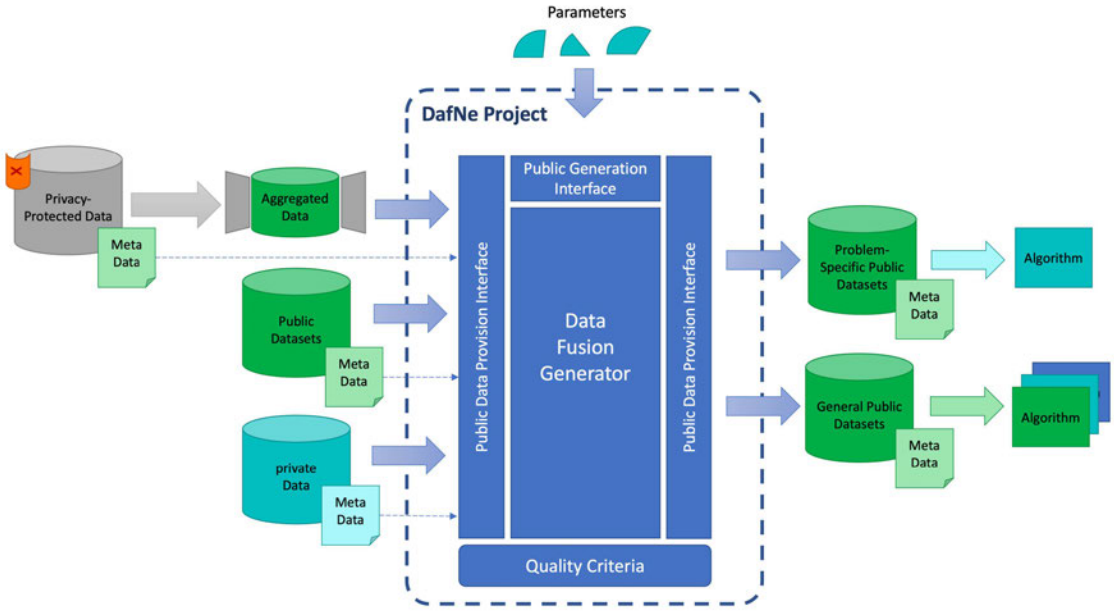



Abbildung 2.1: Übersicht der Architektur und Einsatzkontexte von DaFne (vgl. [46]: Abbildung 1)

Das Projekt untersucht und entwickelt zunächst Methoden zur Erstellung von repräsentativen, datenschutzfreundlichen synthetischen Mobilitätsdaten. Anschließend bewertet es die generierten Daten sorgfältig statistisch, um ihre Nützlichkeit im Vergleich zu realen Daten zu ermitteln. Dies geschieht insbesondere im Kontext von Smart Cities. Das Projekt hat zum Ziel, modernste KI-Verfahren einer breiten Nutzerbasis zugänglich zu machen, einschließlich KMU und Start-Ups. Hierfür wird eine offene Plattform entwickelt, die die Methoden enthält und parametrisiert sowie öffentlich verfügbar macht. Die Plattform wird zukünftig um weitere Methoden erweitert. Eine Übersicht der Plattformarchitektur ist in Abbildung 2.1 dargestellt (vgl. [46]: S. 5).

Die entwickelten Methoden und die Plattform werden durch praktische Anwendungsfälle im Bereich Smart Cities getestet. In einem Szenario wird beispielsweise die Personenmobilität auf der Grundlage synthetischer Daten analysiert, um öffentliche Verkehrsmittel und Verkehrsströme effizienter zu gestalten. Dies geschieht, da individuelle Mobilitätsdaten aus Datenschutzgründen normalerweise nicht verfügbar sind. Ein weiterer Anwendungsfall unterstützt die Planung von Infrastrukturen in neuen Stadtvierteln im Kontext von Smart Cities. Hierbei könnte KI die Vorhersage zukünftiger Nutzungsszenarien für Wohn- und Geschäftsräume verbessern. Im Verlauf des Projekts werden diese Anwendungsfälle

weiter verfeinert, um den Datenbedarf und das Potenzial der Datengenerierungsmethoden besser zu bestimmen (vgl. [46]: S. 5).

### 2.1.3 Funktionalitäten

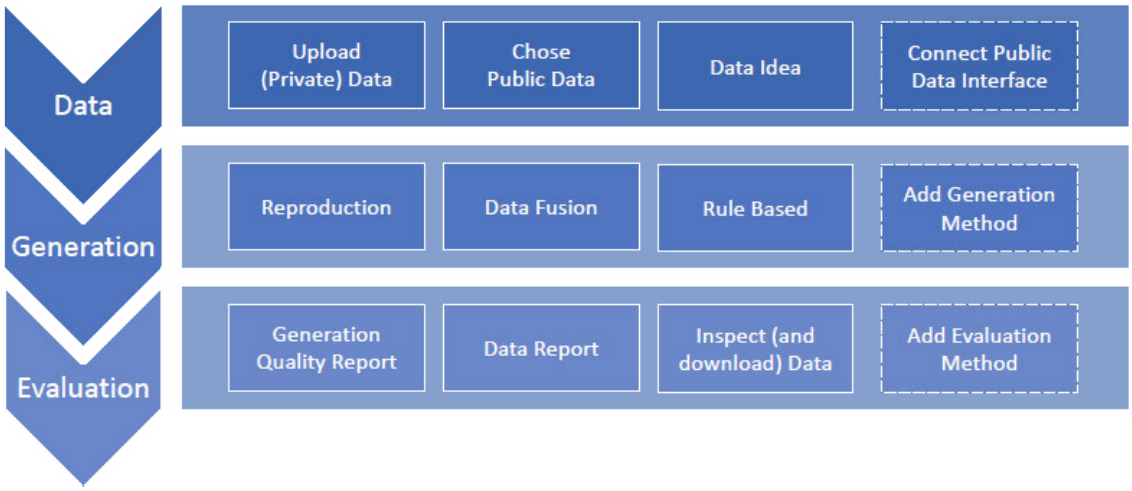
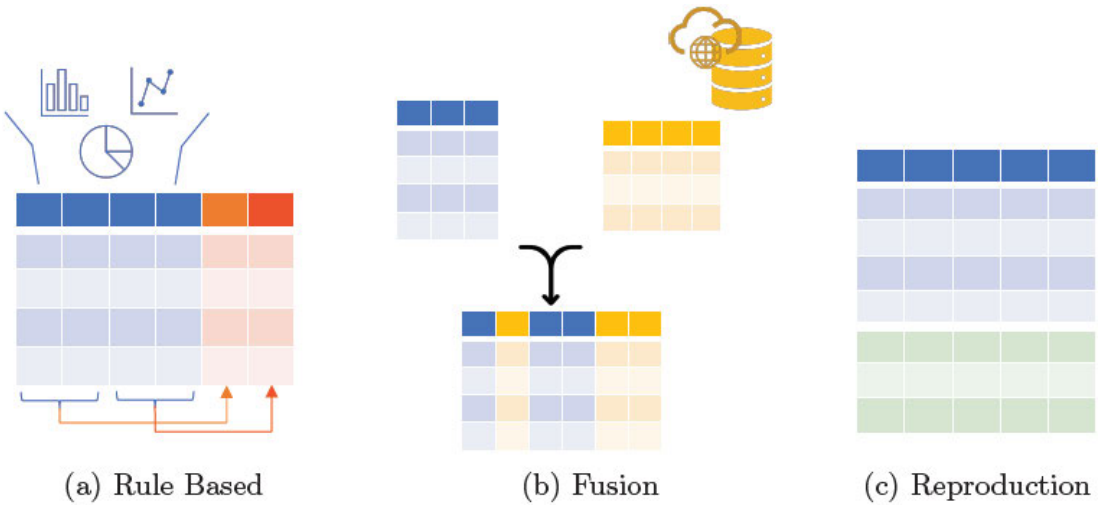



Abbildung 2.2: Überblick über die Funktionalitäten der Plattform (vgl. [33]: Abbildung 1)

Die Plattform bietet Nutzern eine breite Palette an Funktionen zur Synthese von tabellarischen Daten, wie in Abbildung 2.2 ersichtlich. Der Prozess beginnt mit dem Hochladen eigener Daten oder der Auswahl bereits vorhandener öffentlicher Daten aus dem Smart-City-Kontext. Es sind Funktionen zur Bearbeitung, Erweiterung und Qualitätssicherung von echten und synthetischen Daten integriert. Zusätzlich wird sie die synthetischen Daten bereitstellen und die trainierten Generierungsmodelle speichern, um in Zukunft ähnliche Datensätze zu generieren. Eine grafische Übersicht veranschaulicht alle Funktionalitäten in den Segmenten Datenmanagement, Generierung und Evaluierung (vgl. [33]: S. 3-5).

Die Plattform ermöglicht Nutzern die Verarbeitung von Daten. Hierzu können eigene tabellarische Daten hochgeladen werden, und es besteht die Möglichkeit auf eine Auswahl von öffentlich verfügbaren Smart-City-Daten zuzugreifen. Ein integriertes Datenmodell unterstützt eine Vielzahl von Datenarten spezifisch für Smart Cities, einschließlich räumlicher und zeitbezogener Daten. Dadurch wird die Realisierung spezifischer Datenprojekte erleichtert. Berechtigte Nutzer können die Plattform um neue Datenquellen erweitern, indem sie neue Schnittstellen hinzufügen (vgl. [33]: S. 3-5).



(a) Rule Based (b) Fusion (c) Reproduction

Abbildung 2.3: Varianten der Datengenerierung auf der Plattform (vgl. [33]: Abbildung 2)

Die Datensynthese auf der Plattform umfasst verschiedene Funktionalitäten, die Nutzer auch in Kombination nutzen können. Im Folgenden werden diese Funktionalitäten in Abbildung 2.3 beschrieben (vgl. [33]: S. 3-5).

(a) **Rule-based data generation:** Die Plattform ermöglicht es den Nutzern, eigene Datenspalten nach individuellen Vorgaben zu erstellen. Dabei können sie mithilfe regelbasierter Generierung den Datentyp, die Verteilung und spezifische Werte festlegen. Abbildung 2.3 zeigt dies in blauer Farbe. Zusätzlich können Spalten generiert werden, die auf bestimmten Regeln basieren und von anderen Spalten abhängen. Diese sind in orange und rot dargestellt. Auf diese Weise können tabellarische Daten erstellt werden, ohne auf vorher existierende Daten angewiesen zu sein. Benutzer können diese Methode nutzen, um bestehende Datensätze spaltenweise zu erweitern und zu ergänzen (vgl. [33]: S. 3-5).

(b) **Data fusion:** Datenfusion bezeichnet das Zusammenführen verschiedener Datensätze, wie in Abbildung 2.3b dargestellt. Benutzer können ihre eigenen Daten (in Blau) mit öffentlich zugänglichen Informationen (in Gelb), wie Wetter- oder Verkehrsflussdaten, kombinieren, um den Informationsgehalt ihrer ursprünglichen Daten zu erhöhen (vgl. [33]: S. 3-5).

(c) **Reproduction/ Data generation based on existing data:** Durch Vervielfältigung oder Generierung von Daten auf Basis vorhandener Informationen, wie in Abbildung 2.3c dargestellt, können neue Zeilen (grün) zu einem bestehenden tabellarischen

Datensatz (blau) hinzugefügt werden. Diese neuen Zeilen ähneln den vorhandenen Daten in ihren Merkmalen und Beziehungen. Die Plattform bietet verschiedene Bewertungsmethoden an, mit denen der Nutzer die Ähnlichkeit zwischen den generierten und den tatsächlichen Daten bewerten und auswählen kann (vgl. [33]: S. 3-5).

Es ist wichtig, dass die Generierungs- und Bewertungsmethoden der Plattform stets auf dem neuesten Stand der Technik bleiben. Sie sollten leicht um neue Methoden erweitert oder ausgetauscht werden können, wobei nur autorisierte Nutzer dazu befugt sind. Nach der Datengenerierung kann der Nutzer die Qualität der synthetischen Daten anhand verschiedener Qualitätsmetriken überprüfen, insbesondere bei der Anwendung generativer Machine Learning-Modelle. Hierbei werden reale und synthetische Daten hinsichtlich statistischer Ähnlichkeit, Wahrscheinlichkeit und Machine-Learning-Effektivität verglichen. Ein Datenreport bietet dem Nutzer eine Übersicht über die Daten, einschließlich Merkmale, Werte und deren Verteilungen oder fehlende Werte. Nutzer können die Daten herunterladen und eigenständig überprüfen. Obwohl es in Abbildung 2.2 nicht explizit gezeigt wird, ist die Speicherung der Generierungsmodelle Teil der Plattform. Dadurch können eingeloggte Nutzer später darauf zugreifen und bei Bedarf weitere Daten desselben Typs generieren (vgl. [33]: S. 3-5).

## 2.2 Identitäts- und Berechtigungsmanagement

Im nachfolgenden Unterkapitel wird das Identitäts- und Berechtigungsmanagement einer näheren Betrachtung unterzogen.

### 2.2.1 Begriffserklärung

Das Identitäts- und Berechtigungsmanagement ist ein wesentlicher Bestandteil der IT-Sicherheit, dessen Ziel es ist, die Sicherheit sowie die Vertraulichkeit, Integrität und Verfügbarkeit von IT-Systemen zu gewährleisten. Es definiert, wer Zugang zu IT-Systemen hat und welche Rechte diesen Nutzern zugewiesen werden. Das Identitätsmanagement fokussiert sich auf die Verwaltung von Identitäten und die Identifizierung von Nutzern, wobei es teilweise auch die Authentifizierung beinhaltet. Diese Authentifizierung überprüft die Identität der Nutzer und bildet die Grundlage für die nachfolgende Autorisierung (vgl. [12]: S. 1; [49]: S. 23-35; [16]: S. 393; [26]: S. 163; [41]: S. 67; [48]: S. 1; [19]: S. 78-79; [13]: S. 189-204).

Nur autorisierte Benutzer und IT-Komponenten sollten Zugang zu den geschützten Ressourcen einer Organisation erhalten. Dabei ist es wichtig, dass sowohl Benutzer als auch IT-Komponenten eindeutig identifiziert und authentifiziert werden. Die Verwaltung dieser Identitätsinformationen fällt unter den Bereich des Identitätsmanagements. Die Authentifizierung, gefolgt von der Autorisierung und der Zugriffskontrolle, sind Kernelemente des Berechtigungsmanagements, welches bestimmt, inwieweit Nutzer oder IT-Systeme Informationen einsehen oder Dienste nutzen dürfen. Dies schließt alle Methoden zur Vergabe, Entziehung und Überwachung von Zugriffsrechten ein (vgl. [12]: S. 1; [49]: S. 1, 23-35, 83; [16]: S. 393; [26]: S. 163; [41]: S. 67; [48]: S. 1; [19]: S. 78-79; [13]: S. 189-204; [23]: S. 289).

Im Folgenden wird die Struktur des Kapitels dargelegt, welches die einzelnen Komponenten des Identitäts- und Berechtigungsmanagements in der Reihenfolge präsentiert, wie sie in Abbildung 2.4 dargestellt ist.

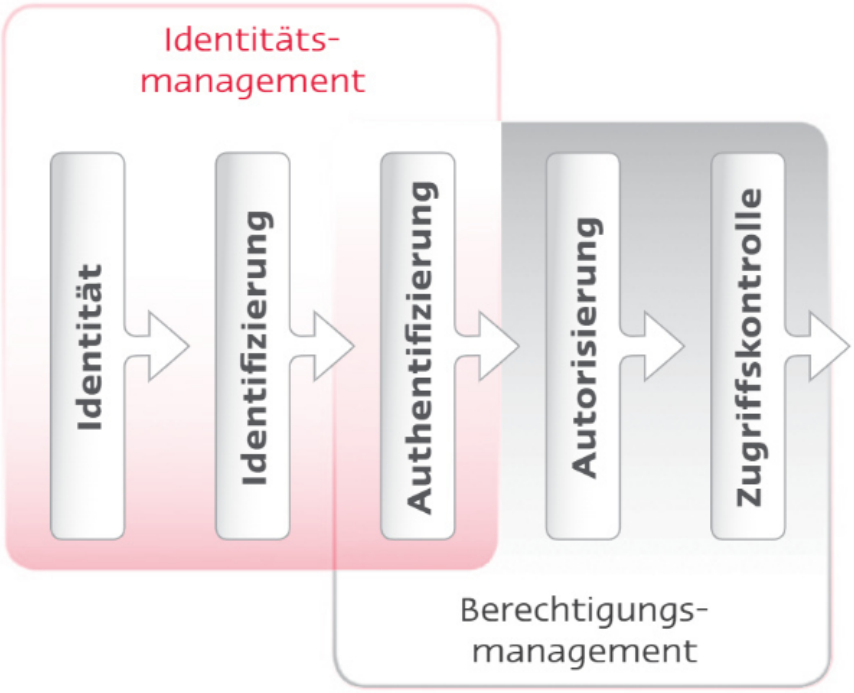



Abbildung 2.4: Elemente des Identitäts- und Berechtigungsmanagements (vgl. [16]: Abbildung 15.1)

### 2.2.2 Identität

Die Identität eines Menschen setzt sich aus einer Kombination einzigartiger Merkmale zusammen, die ihn von anderen unterscheiden. Dazu gehören persönliche Daten, Gesundheitsinformationen, Präferenzen und Online-Verhalten. Je nach Kontext sind unterschiedliche Merkmale zur Identifizierung einer Person notwendig, weshalb nur ausgewählte Informationen preisgegeben werden sollten (vgl. [16]: S. 394).

Identitätsdiebstahl bezeichnet den Missbrauch einer fremden Identität durch Dritte, oft durch Techniken wie Phishing, bei denen durch betrügerische E-Mails vertrauliche Daten abgefangen werden. Solche gestohlenen Identitäten können für betrügerische Handlungen missbraucht werden (vgl. [16]: S. 394).

Im Bereich der IT-Sicherheit stützen sich Authentifizierung und Autorisierung fundamental auf die Identität einer Person. Dabei können die genauen Begriffsdefinitionen variieren. Identität bezieht sich im Allgemeinen auf die Merkmale, die eine Entität eindeutig definieren. Im digitalen Umfeld kann eine Person mehrere Identitäten annehmen, einschließlich rollen- und gruppenbezogener Merkmale sowie Namen und Adressen (vgl. [47]: S. 23-24).

Der Umgang mit Identitätsdaten, einschließlich Stammdaten wie Kontaktdaten und Daten zur Autorisierung, ist rechtlich relevant. Identitätsdaten umfassen auch solche, die speziell für Identitätsmanagementprozesse erstellt werden, wie beispielsweise Nutzernamen und Passwörter (vgl. [47]: S. 23-24).

Zusammenfassend basiert die Identität eines Individuums auf einer einzigartigen Zusammensetzung von Attributen und kann je nach Kontext verschiedene Formen annehmen. Das Management dieser Identitäten umfasst die Auswahl und Authentifizierung unter verschiedenen Identitäten, abhängig vom benötigten Szenario. Dabei müssen auch Datenschutzaspekte berücksichtigt werden, um zu verhindern, dass verschiedene Identitäten einer Person miteinander verknüpft werden können (vgl. [49]: S. 24; [41]: S. 12).

### 2.2.3 Identifizierung

Identifizierung umfasst die Vorlage spezifischer Merkmale eines Individuums, die es in einem bestimmten Kontext eindeutig ausweisen, ohne jedoch die Echtheit dieser Identität

zu beweisen. Nutzer oder Systeme werden anhand einzigartiger Kennzeichen, wie Benutzername, E-Mail-Adresse oder biometrische Daten, erkannt. Es ist möglich, zur Identifizierung eine Kombination dieser Merkmale zu nutzen. Biometrische Identifikationsmethoden umfassen physische Eigenschaften wie Fingerabdrücke oder Irismuster sowie Verhaltensmuster wie die Stimme oder Tastatureingaben. In der Online-Kommunikation ist eine genaue Identifizierung aufgrund des oft fehlenden persönlichen Kontakts besonders wichtig, um Identitätsdiebstahl zu vermeiden. Dieser kann entstehen, wenn Dienstanbieter bei der Feststellung der Identität nachlässig sind. Die Genauigkeit der Identifizierung spielt eine Rolle. Die Bestätigung der Authentizität der dargestellten Identität erfolgt jedoch durch gesonderte Authentifizierungsverfahren. Biometrische Daten können sowohl zur Identifizierung als auch zur Authentifizierung herangezogen werden. Identifizierung beschreibt somit den Prozess der Feststellung, um welche Entität es sich anhand bestimmter Merkmale handelt (vgl. [16]: S. 395; [26]: S. 161).

### 2.2.4 Authentifizierung

In der deutschen Sprache gibt es zwei Begriffe, *Authentifizierung* und *Authentisierung*, die beide den Prozess der Identitätsüberprüfung behandeln. Im Englischen existiert nur *authentication*, was zu Verwechslungen führen kann. *Authentifizierung* bezieht sich auf den Vorgang, in dem eine Partei (Relying Party) die behauptete Identität einer Person überprüft. Bei der *Authentisierung* hingegen weist eine Person oder ein System ihre/seine Identität nach (vgl. [16]: S. 395-396).

Es gibt fünf grundlegende Methoden zur Authentifizierung: Wissen (z.B. ein Passwort), Besitz (z.B. ein Token) und physische oder verhaltensbasierte Merkmale (z.B. Fingerabdruck, Stimmregister), Fähigkeiten (z.B. eine Unterschrift leisten) oder ein bestimmter Standort. Die ersten drei Methoden sind am gebräuchlichsten. Zur Erhöhung der Sicherheit können auch Kombinationen dieser Merkmale verwendet werden. Die Klassifikation von Authentifizierungsmaßnahmen erfolgt anhand der Anzahl der verwendeten Merkmale, wie beispielsweise Ein-Faktor-, Zwei-Faktor- oder Drei-Faktor-Authentifizierung. Ein häufiges Beispiel für Zwei-Faktor-Authentifizierung findet sich bei der Nutzung eines Geldautomaten, bei der sowohl eine Bankkarte (Besitz) als auch eine PIN (Wissen) erforderlich sind (vgl. [16]: S. 395-396; [49]: S. 129).

Authentifizierung dient der Überprüfung einer behaupteten Identität, sei es eines Menschen oder Systems, und belegt die Echtheit dieser Identität durch identitätsgebundene

Informationen. Die Authentifizierung basiert historisch auf der Namensnennung, welche in vielen Kulturen spezifische Authentifizierungsmerkmale aufweist. Heutzutage findet dies auch in der Informationstechnologie Anwendung. Nach erfolgreicher Authentifizierung kann einer Person oder einem System eine definierte Rolle innerhalb eines Systems zugewiesen werden (vgl. [26]: S. 162).

### 2.2.5 Identitätsprovider

Identity Provider (IdP) spielen eine wichtige Rolle bei der Verwaltung von Benutzeridentitäten und -zugängen. Sie dienen als zentrale Instanz, die Identitätsdaten speichert und Authentifizierungsdienste für Benutzer bereitstellt. Beispiele dafür sind: Native Azure AD, G Suite, PingFederate (vgl. [38]), Shibboleth (vgl. [49]: S. 218) und Keycloak (vgl. [1]).

Authentifizierungsprotokolle wie SAML (Security Assertion Markup Language) und OpenID sind Standards, die die Interaktion zwischen Identity Provider und Service Provider erleichtern. Diese Protokolle ermöglichen es Benutzern, sich sicher bei verschiedenen Diensten anzumelden, ohne jedes Mal ihre Anmeldeinformationen erneut eingeben zu müssen. Wenn von IdPs die Rede ist, ist der Serviceprovider die Einrichtung, die die digitale Ressource verwaltet, auf die ein Benutzer zuzugreifen versucht. Der Identity Provider stellt dem Service Provider auf Anfrage des Nutzers die Authentifizierungsdaten zur Verfügung. Daher ist ein Identity Provider technisch gesehen auch ein Service Provider (vgl. [38]).

Authentifizierungsmethoden wie Multi-Faktor-Authentifizierung (MFA) und Single Sign-On (SSO) ergänzen diese Protokolle (vgl. [38]).

Insgesamt arbeiten Identitätsprovider, Authentifizierungsprotokolle und Authentifizierungsmethoden zusammen, um eine sichere, benutzerfreundliche und effiziente Authentifizierungsumgebung für Benutzer bereitzustellen. Die Authentifizierungsmethoden und -protokolle werden im nächsten Abschnitt näher vorgestellt.

### 2.2.6 Authentifizierungsmethoden und -protokolle

Authentifizierungsmethoden sind verschiedene Techniken, um die Identität von Personen, Systemen oder anderen Einheiten zu validieren. Diese Verfahren umfassen Passwörter, biometrische Daten wie Fingerabdrücke oder Gesichtserkennung, Tokens, Smartcards

oder Methoden wie die Zwei-Faktor-Authentifizierung (2FA). Das Ziel ist es, sicherzustellen, dass nur berechtigte Nutzer oder Systeme Zugang zu bestimmten Ressourcen erhalten. Einige Beispiele für solche Methoden sind SSO, 2FA und MFA.

Single Sign-On (SSO) ermöglicht Nutzern nach einer einzigen Anmeldung am Arbeitsplatz Zugriff auf mehrere Anwendungen ohne weitere Authentifizierungsprozesse. Die Anmeldung kann über Benutzer-ID und Passwort oder Technologien wie PKI und Smartcards erfolgen, unterstützt durch Verzeichnisdienste. SSO verbessert den Benutzerkomfort und steigert die Produktivität. Es reduziert den Supportaufwand für das Passwort-Management und erhöht die Sicherheit durch den Einsatz starker Passwörter und möglicher Smartcards. Ein Risiko von SSO besteht darin, dass ein einmal kompromittiertes Passwort einem Angreifer Zugriff auf alle Dienste des Nutzers ermöglicht (vgl. [49]: S. 200-201; [16]: S. 418; [25]: S. 174-175).

Eine mögliche Lösung hierfür ist die Zwei-Faktor-Authentifizierung (2FA). 2FA erhöht die Sicherheit, indem es einen zweiten Authentifizierungsfaktor zum herkömmlichen Passwort hinzufügt. Dabei wird die Kombination eines Wissenselements mit einem Besitzelement (z.B. einem Mobilgerät für SMS-Codes) oder einem biometrischen Faktor angestrebt. Die Unabhängigkeit beider Faktoren ist entscheidend, um die Authentifizierung wirksam zu machen. Die Verschiebung eines Faktors in eine andere Kategorie, wie beispielsweise die Speicherung eines Passworts auf einem Gerät, kann die Effizienz von 2FA beeinträchtigen (vgl. [36]: S. 541-542).

Multi-Faktor-Authentifizierung (MFA) kombiniert mehrere Sicherheitsmerkmale aus unterschiedlichen Kategorien (Wissen, Besitz, Biometrie), um die Sicherheit weiter zu erhöhen. Ein Beispiel hierfür ist die Kombination eines Passworts mit einer TOTP-App. Windows Hello for Business nutzt Multi-Faktor-Authentifizierung (MFA), indem es Geräte mit biometrischen Daten des Nutzers verknüpft, um die Identität zu überprüfen. Dieser Ansatz bindet physische Merkmale an ein spezifisches Gerät, wodurch implizit eine zusätzliche Sicherheitsstufe etabliert wird. Bei der Wahl von MFA-Methoden ist es wichtig, Mechanismen zu wählen, die auf verschiedenen Authentifizierungsfaktoren basieren, um die Sicherheit effektiv zu erhöhen (vgl. [25]: S. 172).

Authentifizierungsprotokolle sind eine Sammlung von Richtlinien, Abläufen und Strukturen, die den Informationsaustausch zur Identitätsbestätigung zwischen einem Client und einem Server regeln. Diese Protokolle spezifizieren, wie die Authentifizierungskommunikation abläuft, einschließlich der Überprüfung der Clientidentität und der Entscheidung

über den Zugang zu Diensten. Bekannte Beispiele für solche Protokolle sind SAML, Kerberos, OAuth und OpenID.

SAML (Security Assertion Markup Language) ist ein XML-basierter Standard, der zur Übermittlung von Authentifizierungs- und Autorisierungsinformationen innerhalb von Webservices entwickelt wurde. Der Standard ermöglicht den Austausch verschiedener Arten von Authentifizierungsdaten wie Passwörter, Kerberos-Tickets oder X.509-Zertifikate. Dabei werden SAML-Assertions verwendet, die Authentifizierungs-, Attribut- und Autorisierungsentscheidungen enthalten. Es ermöglicht einer Serviceanwendung, Entscheidungen darüber zu treffen, wie auf eine Zugriffsanfrage reagiert wird, basierend auf den vom Identity Provider bereitgestellten Informationen über den Benutzer. SAML wird weitverbreitet für das Identity- und Access Management in verschiedenen Softwarelösungen eingesetzt (vgl. [49]: S. 283-285; [16]: S. 426-427).

Kerberos basiert auf der Idee der Authentifizierung durch einen vertrauenswürdigen Dritten, dem Key Distribution Center (KDC), das symmetrische Schlüssel vergibt. Die Authentifizierung erfolgt durch Ausstellen von Tickets, die für die Kommunikation zwischen dem Benutzer und dem angeforderten Dienst verwendet werden. Kerberos minimiert das Risiko der Übertragung von Passwörtern über das Netzwerk, indem verschlüsselte Tickets verwendet werden. Es ermöglicht eine effiziente und sichere Authentifizierung in einer verteilten Netzwerkumgebung (vgl. [49]: S. 138-141; [16]: S. 408-410).

OAuth (Open Authorization) erlaubt es Benutzern, Drittanwendungen einen begrenzten Zugriff auf ihre geschützten Ressourcen bei einem anderen Service zu gewähren, ohne dass dafür ihre Zugangsdaten preisgegeben werden müssen. Dies geschieht mithilfe eines Zugriffstokens, das der Dienstanbieter im Namen des Benutzers verwendet, um auf die Daten zuzugreifen. Dadurch wird eine flexible und sichere Methode der Datenfreigabe zwischen Online-Services ermöglicht. (vgl. [16]: S. 425).

OpenID ermöglicht es einer Identität, sich online auf verschiedenen Websites zu authentifizieren. Es gibt zahlreiche OpenID-Anbieter, da es sich um einen offenen Standard handelt, der für Unternehmen und Privatpersonen keine zusätzlichen Kosten verursacht. Ähnlich wie Shibboleth generiert das System eine digitale Identität, die in Form von *Benutzername.Anbieter.tld* oder *Anbieter.tld/ Benutzername* dargestellt wird, und verwendet grundlegende Open-Source-Technologien wie URI, http, SSL, Diffie-Hellmann und andere im Internet verfügbare Standards. Es bietet Zugang zu mehr als 10.000 Websites, bei denen sich Benutzer anmelden können, und wird von mehr als 160 Millionen Benutzern verwendet, die OpenID-URIs verwenden. Die OpenID-Standards werden

von der OpenID Foundation entwickelt und gepflegt. Als Single-Sign-On (SSO) System ersetzt OpenID den herkömmlichen Anmeldeprozess mit Benutzername und Passwort. Der Benutzer gibt seine OpenID-Identität an, um sich auf einer OpenID-Website anzumelden, wobei die primäre Authentifizierung durch den OpenID-Provider erfolgt. Eine erfolgreiche Anmeldung beim OpenID-Provider bestätigt auch die Anmeldung auf der Ursprungsseite. Bei der OpenID-Anmeldung auf einer Website wird der Benutzer auf die Anmeldeseite des OpenID-Providers umgeleitet. Dort erfolgt die Anmeldung und Authentifizierung nach dem gleichen Prinzip wie bei Shibboleth, bevor der Zugriff auf die eigentliche Seite erlaubt wird. Die Anmeldung wird automatisch bestätigt, sobald die Seite, auf der die Anmeldung erfolgt, als vertrauenswürdig eingestuft wird. Nach der Bestätigung der Anmeldung wird der Benutzer auf die Seite zurückgeleitet, auf der er sich ursprünglich angemeldet hat (vgl. [49]: S. 219; [16]: S. 424-425).

### 2.2.7 Autorisierung

Autorisierung in Computersystemen beschreibt den Prozess, bei dem das System nach der Verifizierung der Identität eines Benutzers prüft, ob dieser die gewünschte Aktion durchführen darf. Dabei wird entschieden, welche Handlungen oder Zugriffe einem Benutzer oder einer Identität innerhalb eines Netzwerks oder auf bestimmte Systemressourcen gestattet werden. Autorisierung beinhaltet die Zuweisung von Rechten an eine Identität, die es ihr ermöglicht, spezifische Aktionen auszuführen. Der gesamte Prozess umfasst zumindest die Schritte der Authentifizierung (Überprüfung der Identität) und der eigentlichen Autorisierung (Überprüfung und Genehmigung des Zugriffs oder der Handlung basierend auf geltenden Regeln) (vgl. [49]: S. 161-163).

Ein analoges Beispiel zur Verdeutlichung der Autorisierung im Alltag ist der Ablauf einer Kreditkartentransaktion. Beim Versuch, mit einer Kreditkarte zu bezahlen, werden die Kartendaten geprüft. Dabei wird die Gültigkeit und der Verfügungsrahmen der Karte ebenso wie die Identität des Karteninhabers überprüft. Eine Transaktion wird nur dann genehmigt, wenn alle Voraussetzungen erfüllt sind und die Regeln des Kreditkarteninstituts dies zulassen. Der Betrag wird entsprechend verbucht (vgl. [49]: S. 161-163).

Zusammenfassend regelt die Autorisierung den Zugang und die Handlungsrechte in einem IT-System auf Basis vorab definierter Berechtigungen, die an die Identität eines Benutzers gebunden sind. (vgl. [26]: S. 162). Es ist wichtig zu beachten, dass die Autori-

sierung lediglich ein genereller Überbegriff für die Zulassung zu Ressourcen in Form einer Zugriffskontrolle ist (vgl. [49]: S. 161-163).

### 2.2.8 Zugriffskontrolle

Zugriffskontrolle in der Informationssicherheit ist ein grundlegendes Konzept, das regelt, wie der Zugang zu Systemen, Netzwerken, Anwendungen, Daten oder Ressourcen verwaltet wird. Das Ziel besteht darin, sicherzustellen, dass nur autorisierte Benutzer, Programme oder Prozesse auf spezifische Informationen oder Funktionen zugreifen können, während unbefugter Zugriff verhindert wird. Um sicherzustellen, dass dies gewährleistet ist, werden Zugriffsrechte individuellen Benutzerkonten zugeordnet, die durch Identifikation und Authentifizierung geschützt sind (vgl. [16]: S. 412; [27]: S. 19-21).

Das Need-to-Know-Prinzip bildet in der Praxis den Ausgangspunkt für die Festlegung spezifischer Regeln zur Vergabe von Zugriffsrechten. Gemäß diesem Prinzip sollen Benutzer nur auf die für ihre Aufgaben erforderlichen Informationen zugreifen können. Dies trägt dazu bei, die Handlungsmöglichkeiten interner Angreifer einzuschränken. Für bestimmte Aufgaben kann auch verlangt werden, dass Zugriffe nur in Kombination von zwei Benutzern möglich sind. Einige Produkte unterstützen bereits das Vier-Augen-Prinzip. Unzureichende Vergabe, Prüfung oder Konfiguration von Zugriffsrechten können zu unentdeckten Sicherheitslücken führen, die den unerlaubten Informationsabfluss oder Daten- und Systemmanipulationen ermöglichen (vgl. [16]: S. 412; [27]: S. 19-21).

Daher ist es entscheidend, dass die Verwaltung von Zugriffsrechten - also die Zuweisung und Rücknahme von Zugriffsrechten für Benutzerkonten - sorgfältig und in strukturierten Prozessen erfolgt, um die Sicherheit sensibler Informationen und Systeme zu gewährleisten (vgl. [16]: S. 412; [27]: S. 19-21).

Es gibt verschiedene Modelle der Zugriffskontrolle, die fünf bekanntesten werden im Abschnitt 2.4 vertieft vorgestellt. Ein Zugriffskontrollmodell bildet die Grundlage dafür, wie ein Subjekt auf ein Objekt zugreift. Zugriffskontrollmodelle und Berechtigungskonzepte bzw. -modelle beziehen sich auf dasselbe Konzept und sind in ihrem Wesen identisch.

### 2.2.9 Zugriffskontrollmatrix

Im gegebenen Beispiel in Tabelle 2.1 wird der Drucke-Auszug als Prozess betrachtet, der von Bob aufgerufen werden kann und somit als Objekt fungiert. Gleichzeitig müssen

diesem Prozess spezifische Ausführungsrechte gewährt werden, was ihn auch als Subjekt darstellt (vgl. [16]: S. 413).

| | Objekte | | |
|---|---|---|---|
| Subjekte | Konto_Bob | Konto_Alice | Drucke-Auszug |
| Bob | Lese-Kontostand | - | ausführen |
| Alice | - | Lese-Kontostand | |
| Drucke-Auszug | Lese-Kontostand | Lese-Kontostand | |

Tabelle 2.1: Zugriffskontrollmatrix (vgl. [16]: Abbildung 15.6)

Eine Zugriffskontrollmatrix ist ein Werkzeug zur visuellen Darstellung und Verwaltung von Berechtigungen. Sie verdeutlicht, welche Akteure (wie Benutzer oder Benutzergruppen) welche Rechte an spezifischen Ressourcen (wie Dateien, Datenbanken oder Systemfunktionen) erhalten. Die Zugriffskontrollmatrix, auch als Access Control Matrix bezeichnet, bietet eine strukturierte Übersicht darüber, wer innerhalb eines IT-Systems welche Aktionen durchführen darf (vgl. [16]: S. 412; [49]: S. 170).

Sie bildet somit die praktische Umsetzung der Grundsätze eines Berechtigungskonzepts und trägt zur Sicherung der Integrität und Vertraulichkeit von Informationen bei. Die Integrität wird durch die Kontrolle von Schreiboperationen und die Vertraulichkeit durch die Kontrolle von Leseoperationen gewährleistet (vgl. [16]: S. 412; [49]: S. 170).

Die Zugriffskontrollmatrix weist Subjekte (S) und Objekte (O) einander zu. Sie besteht aus Einträgen für SxO (einen Eintrag pro Subjekt-Objekt-Paar) und hält die Beziehung zwischen Subjekt, Objekt und Rechten fest (vgl. [16]: S. 412; [49]: S. 170).

Betriebssysteme und Anwendungen nutzen die Zugriffskontrollmatrix, um die Zugriffskontrolle zwischen Subjekten und Objekten zu regeln. Sie spezifiziert, welches Subjekt mit welchem Objekt interagieren darf und wie bzw. welches Subjekt auf welches Objekt zugreifen darf (vgl. [16]: S. 412; [49]: S. 170).

### 2.2.10 Zugriffskontrollliste

Access Control List (ACL) ist eine Methode der Zugriffskontrolle, die in gängigen Betriebssystemen wie Windows, Unix und Linux weit verbreitet ist. Eine ACL speichert

für jedes Objekt eine Liste der autorisierten Benutzer oder Gruppen und deren jeweilige Zugriffsrechte. Beispiele für ACL-Einträge sind (vgl. [16]: S. 414):

*ACL (Konto-Bob) = ((Bob, (Lese-Kontostand)), (Drucke-Auszug (Lese-Kontostand)))*
*ACL (Konto-Alice) = ((Alice, (Lese-Kontostand), (Drucke-Auszug (Lese-Kontostand)))*
*ACL (Drucke-Auszug) = (Bob, (ausführen))*

Diese Einträge entsprechen den Spalten der Zugriffsmatrix, die in Tabelle 2.1 oben gezeigt wurde. Ein praktisches Anwendungsbeispiel für ACL findet sich in der Unix-Zugriffskontrolle, wo für jedes Objekt Schutzbits für den Eigentümer, die Gruppe und andere gesetzt werden. Diese so genannten rwx-Bits stellen eine vereinfachte Form von ACLs dar (vgl. [16]: S. 414).

Das ACL-Verfahren zeichnet sich durch seine Einfachheit und Flexibilität aus. Die für Objekte definierten Zugriffsrechte können sehr effizient ermittelt werden. Die Bestimmung der Rechte eines Subjekts, d.h. auf welche Objekte es zugreifen darf, erweist sich jedoch oft als sehr aufwendig, da das gesamte System durchsucht werden muss, um diese Information zu erhalten (vgl. [16]: S. 414; [49]: S. 171).

## 2.3 Identitätsprovider und Authentifizierungsmechanismus von DaFne

In diesem Abschnitt wird kurz vorgestellt, welcher Identitätsprovider und welche Authentifizierungsmethode im Prototypen der DaFne-Plattform verwendet wird.

Gemäß dem Paper (vgl. [39]) wird für die DaFne-Plattform Keycloak als Identitätsprovider verwendet und der Authentifizierungsmechanismus mit Single-Sign-On über OpenID umgesetzt.

Für den Prototypen wurde die Authentifizierung über OpenID mit Keycloak als IdP gewählt. Keycloak ermöglicht es, verschiedene Benutzergruppen mit Rollen und Berechtigungen in Bereiche zu bündeln, diese über verschiedene Clients mit unterschiedlichen Protokollen zu identifizieren und die Tokens mit spezifischen Werten (Claims) anzureichern. Der Keycloak-Dienst läuft auf verschiedenen Datenbanken, die entsprechend gesichert und verwaltet werden müssen. Insgesamt bietet Keycloak damit ein hohes Maß an Flexibilität.

Für den Prototypen sind JSON Web Tokens (JWT) sinnvoll, da sie die Implementierung von Diensten ohne Sessions ermöglichen. Über diese Tokens können nicht nur Benutzer authentifiziert, sondern auch Werte direkt übertragen werden. JWTs werden in gängigen Programmiersprachen unterstützt und können zur Zuordnung von Benutzern zu Rollen, Gruppen und anderen Werten auf technischer und Anwendungsebene verwendet werden.

Die Authentifizierung über öffentliche Schlüssel erfolgt ausschließlich über die Kommunikation mit dem IdP. Zusätzlich bietet Keycloak Docker-Images an, die nur noch konfiguriert werden müssen. Einmal eingerichtet, kann die Konfiguration über eine API oder ein Webinterface erfolgen. Damit steht für die Plattform eine Authentifizierungskomponente zur Verfügung, die von externen Quellen angesprochen werden kann.

## 2.4 Berechtigungskonzepte und -modelle

In diesem Unterkapitel werden die verschiedenen Berechtigungskonzepte und -modelle vorgestellt, die als theoretische Grundlage für die Konzeption eines geeigneten Berechtigungsmodells dienen sollen.

### 2.4.1 Diskretionäres Berechtigungskonzept (DAC)

Bei diesem Modell entscheidet der Eigentümer selbst über die Vergabe oder den Entzug von Zugriffsrechten an andere Nutzer oder Benutzergruppen, wie in Abbildung 2.5 dargestellt. Dabei ist es nicht notwendig, dass Administratoren in die Vergabe der Zugriffsrechte eingreifen. Innerhalb des DAC-Modells gibt es zwei Hauptvarianten: eine liberale und eine strenge Ausführung. In der liberalen Variante kann der Eigentümer Zugriffsrechte sowie das Eigentum selbst an andere übertragen, was diese ebenfalls zu Eigentümern der Ressource macht. Die strenge Form begrenzt die Zugriffsrechte und das Eigentum auf den ursprünglichen Erzeuger. Die DAC basiert darauf, dass Zugang und Kontrolle nach dem Ermessen des Eigentümers gewährt werden und stützt sich dabei auf drei zentrale Säulen: Eigentum an der Ressource, Benutzeridentität und die Delegierung von Berechtigungen. Allerdings ist das DAC-Modell aufgrund seiner Anfälligkeit für Sicherheitsrisiken, wie Trojaner-Angriffe, und der Komplexität der Rechteverwaltung in kommerziellen und staatlichen Einrichtungen oft weniger geeignet (vgl. [9]: S. 2-3).

Im Rahmen der DAC erteilen Benutzer selbst Zugriffserlaubnisse auf Objekte, basierend allein auf den dem Benutzerkonto zugewiesenen Rechten. Dadurch wird das Besitzer- oder Eigentümerprinzip hervorgehoben, laut dem der Eigentümer eines Objekts vollständig für dessen Schutz verantwortlich ist und Rechte daran weitergeben kann. Es gibt einen Unterschied zwischen Eigentümern, die umfassende Rechte an einer Ressource haben, und Besitzern, die bestimmte Zugriffsrechte vererben können, aber nicht das vollständige Recht am Objekt besitzen. Diese Flexibilität bei der Rechtevergabe bietet Vorteile in Unternehmen. Allerdings kann sie auch zu widersprüchlichen Zugriffsberechtigungen führen (vgl. [16]: S. 412; [49]: S. 164-165).

Die Verwaltung von Zugriffsrechten erfolgt in der Praxis oft über Zugriffskontrolllisten (ACLs). Diese werden vom Ressourcenbesitzer festgelegt, vom Netzwerkadministrator implementiert und vom Betriebssystem ausgeführt. Das Besitzerprinzip sichert, dass der Besitzer einer Datei entscheidet, wer Zugang hat. Dies basiert auf Einträgen im Dateikopf, die die Nutzungsberechtigungen definieren. Das DAC-Modell ermöglicht es individuellen Besitzern, die Kontrolle über den Zugang zu ihren Ressourcen zu behalten (vgl. [16]: S. 412; [49]: S. 164-165).

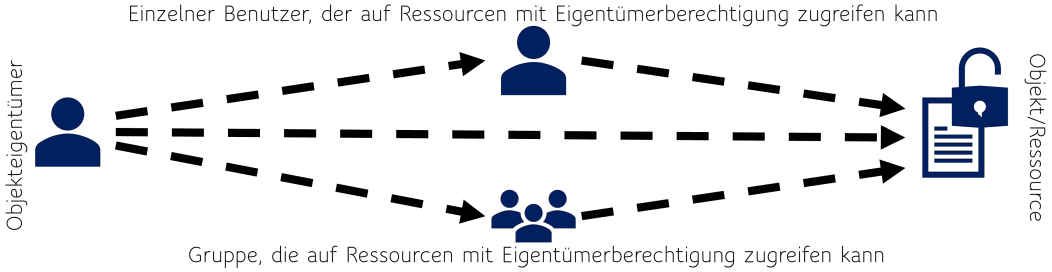



Abbildung 2.5: Abstrakte Ansicht der diskretionären Zugriffskontrolle (DAC) (vgl. [9]: Figure 3)

### 2.4.2 Obligatorisches Berechtigungskonzept (MAC)

Die obligatorische Zugriffskontrolle (Mandatory Access Control, MAC) ist ein Sicherheitsmodell, das auf der Zuweisung von Sicherheitslabels zu Benutzern und Ressourcen basiert. In diesem Modell werden Sicherheitsebenen etabliert, die bestimmen, auf welche Ressourcen Zugriff gewährt wird. Die Rechtevergabe wird streng von einem Administrator überwacht und basiert auf festgelegten Sicherheitsrichtlinien. MAC wird häufig

in hochsicheren Umgebungen wie dem Militär und in bestimmten Industriezweigen eingesetzt. Allerdings bietet es Herausforderungen in Bezug auf Verwaltungsaufwand und Anpassungsfähigkeit, insbesondere in großen Systemen (vgl. [9]: S. 3).

In einem System, das auf ressourcenorientierter Zugriffskontrolle basiert, liegt die Entscheidung über Zugriffsrechte nicht beim Besitzer einer Ressource, sondern wird vom Betriebssystem oder einer zentralen Autorität auf Basis von allgemeinen Sicherheitsregeln getroffen. Dies spiegelt ein strikteres und strukturierteres Modell wider als die benutzerbasierte Zugriffskontrolle (Discretionary Access Control, DAC), bei der Benutzer ihre eigenen Ressourcen kontrollieren. Eine effektive Implementierung dieses Modells erfordert in der Regel eine Datenklassifizierung nach Sicherheitsniveaus, um die Strukturierung der Zugriffsrechte und die Sicherheitsüberwachung zu erleichtern. Die meisten Unternehmen und Organisationen arbeiten hierbei mit vier Stufen: Streng Geheim, Geheim, Vertraulich und Öffentlich, wie in Abbildung 2.6 dargestellt (vgl. [49]: S. 165).

Im Unterschied zur benutzerbestimmten Zugriffskontrolle (DAC), bei der Individuen oder Benutzergruppen eigenständig über die Rechteverteilung entscheiden, legt bei MAC ein zentrales Regelwerk fest, wer auf welche Informationen zugreifen darf. Das Regelwerk orientiert sich an der Klassifizierung der Informationen in verschiedene Sicherheitsgrade und an der zugeordneten Berechtigungsstufe oder Freigabe-Level der Benutzer. Der Zugriff auf Ressourcen wird durch einen Referenzmonitor kontrolliert. Dieser ist eine Schlüsselkomponente, die sicherstellt, dass alle Zugriffe den definierten Sicherheitsrichtlinien entsprechen (vgl. [16]: S. 412-413).

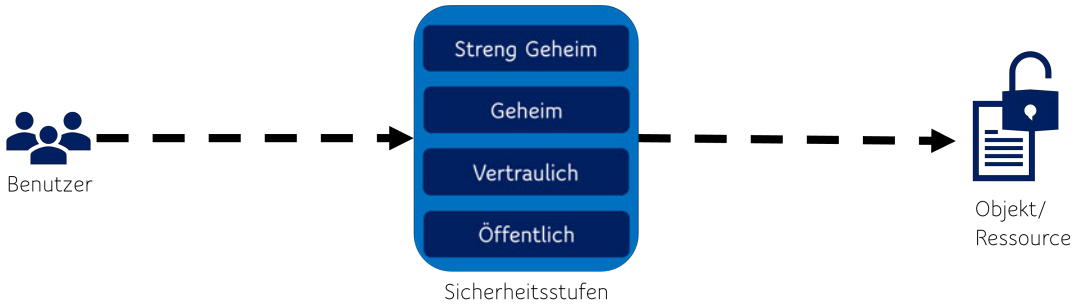



Abbildung 2.6: Abstrakte Ansicht der obligatorischen Zugriffskontrolle (MAC) (vgl. [9]: Figure 4)

### 2.4.3 Regelbasiertes Berechtigungskonzept (RuBAC)

Das regelbasierte Zugriffskontrollmodell wird insbesondere in webbasierten sozialen Netzwerken (WBSN) eingesetzt. Es ermöglicht den Zugang zu Online-Ressourcen abhängig von definierten Regeln. Diese Regeln bestimmen, welche Nutzer aufgrund ihrer Beziehungen, der Verbundenheit und des Vertrauensniveaus zu anderen Nutzern auf bestimmte Inhalte zugreifen dürfen. Der Zugang wird anhand von Kriterien wie Beziehungsart, -tiefe und Vertrauensgrad gesteuert. Die Autorisierung basiert auf Attributen, welche die Eigenschaften der Netznutzer und ihre Verbindung zu anderen beschreiben. Die Bedingungen für den Zugriff werden von den Ressourcenbesitzern festgelegt. Diese formulieren spezifische Zugangsvoraussetzungen, die das Ausmaß der Verbindung zu anderen Nutzern und das damit verbundene Vertrauensniveau betreffen, wie in Abbildung 2.7 dargestellt (vgl. [9]: S. 6).

Im Unternehmenskontext erfolgt die regelbasierte Zugriffskontrolle auf der Netzwerkebene. Dabei regulieren Geräte wie Router, Switches und Firewalls den Zugriff auf Basis von IP-Adressen, Ports und Diensten, die durch explizite Zugriffsregeln bestimmt werden. Die Systemadministratoren verwalten diese Zugriffskontrolle und legen fest, welche Anfragen im Netzwerk zulässig sind und welche abgelehnt werden. Dadurch haben die Nutzer nur begrenzten Einfluss auf die Zugriffsrechte (vgl. [49]: S. 169).

Unter dem regelbasierten Zugriffskontrollansatz, oft als RuBAC abgekürzt, können Nutzerrollen dynamisch zugewiesen werden. Die Zuweisung basiert auf Bedingungen, die von Systemadministratoren definiert werden. Solche Bedingungen können beispielsweise den stundenabhängigen Zugriff auf spezifische Daten beinhalten. Dadurch bietet dieser Ansatz eine dynamische und anpassungsfähige Methode der Zugriffskontrolle. Die Umsetzung dieser Regeln erfordert möglicherweise eine Programmierung, bei der die Zugriffsregelungen als Code vom Administrator in das Netzwerksystem eingefügt werden müssen (vgl. [24]).

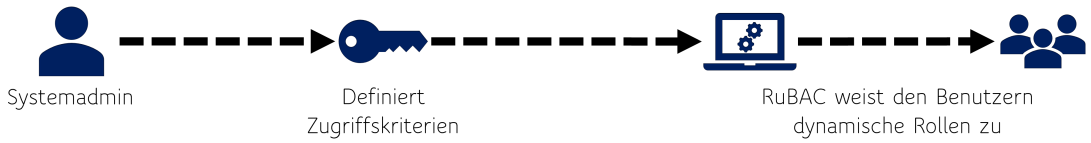



Abbildung 2.7: Abstrakte Ansicht der regelbasierten Zugriffskontrolle (RuBAC) (vgl. [24])

### 2.4.4 Attributbasiertes Berechtigungskonzept (ABAC)

In der Praxis kann die zentralisierte Verwaltung von Benutzerrechten über Entwicklerrollen, Ressourcenrollen und zugehörige Berechtigungen nicht immer reibungslos funktionieren. Ein Beispiel hierfür ist die Situation in einem großen Reiseunternehmen, in dem bestimmte Aufgaben, wie die Stornierung von Buchungen für bestimmte Reisen, nur von leitenden Angestellten mit spezifischer regionaler Verantwortung ausgeführt werden dürfen. Dieser Ansatz führt zu einer großen Anzahl von Rollen innerhalb des Systems, die schnell unübersichtlich werden kann, insbesondere wenn man die vielen Reiseangebote und Reiseziele betrachtet. Die Steuerung der Berechtigungen ausschließlich über Rollen kann jedoch das Gegenteil der ursprünglichen Zielsetzung bewirken, nämlich die Menge der Zuweisungen zwischen Identitäten und Ressourcen zu minimieren (vgl. [49]: S. 79).

Identitätsbasierte Merkmale können zur Steuerung von Zugriffsberechtigungen genutzt werden, wenn sie für spezifische Berechtigungen relevant sind. Ein Beispiel hierfür ist die Zutrittsberechtigung zu einem Gebäude, die durch das Standortattribut gesteuert werden kann, unabhängig von der Funktion oder Abteilungszugehörigkeit einer Person im Unternehmen. Allerdings erfordert die Anwendung solcher Attribute sorgfältige Überlegungen, um sicherzustellen, dass nur wirklich berechtigte Personen die entsprechenden Zugriffsrechte erhalten. Ein Beispiel für eine fehlerhafte Attributsteuerung wäre eine Bank, die über das Standortattribut die Zugriffsberechtigung auf Konten regeln möchte, dadurch aber auch Mitarbeitern, die mit der Kontenverwaltung nichts zu tun haben, Zugriff gewährt. Lösungsansätze könnten darin bestehen, zusätzliche spezifische Attribute oder eine Kombination von Attributen zu verwenden, um die Berechtigungssteuerung präziser zu gestalten (vgl. [49]: S. 94-96).

Das Attributbasierte Zugriffskontrollsystem (ABAC) ermöglicht eine differenzierte Steuerung von Berechtigungen auf Basis von Benutzerattributen, Ressourcen oder Umgebung. Die zugrunde liegenden Prinzipien dieses Systems erlauben eine dynamische Steuerung von Zugriffsberechtigungen, abhängig von Faktoren wie Berufsbezeichnung, Ressourcentyp oder Nutzungskontext, wie in Abbildung 2.8 veranschaulicht. ABAC ermöglicht eine flexible Anpassung der Zugriffsrechte einzelner Personen, je nach deren Standort oder anderen variablen Bedingungen. Subjekte, Objekte und die angestrebten Aktionen werden durch ein System von Bedingungen und Folgerungen definiert, was eine koordinierte und sichere Steuerung des Zugriffs ermöglicht (vgl. [6]).

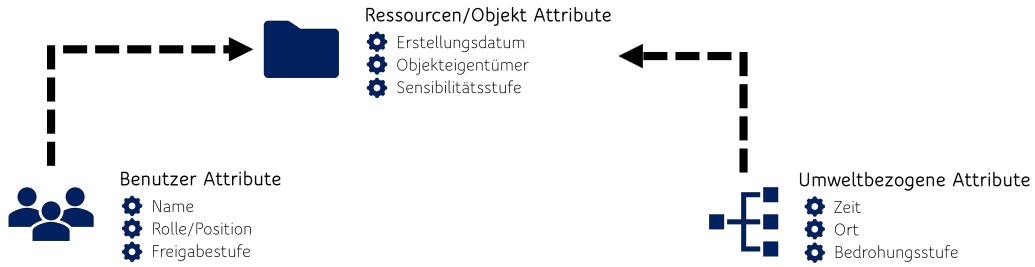



Abbildung 2.8: Abstrakte Ansicht der attributbasierten Zugriffskontrolle (ABAC) (vgl. [6])

### 2.4.5 Rollenbasiertes Berechtigungskonzept (RBAC)

Das Konzept der Role Based Access Control (RBAC) wurde erstmals 1992 von D.F. Ferraiolo und D.R. Kuhn vorgestellt und 2004 als ANSI Standard 359-2004 anerkannt. RBAC hat sich als führendes Modell für die Zugriffskontrolle in verschiedenen Anwendungs- und Betriebssystemen etabliert. Es basiert auf der Zuweisung von Rechten an Benutzer auf der Grundlage ihrer Rollen im Unternehmen. Statt Berechtigungen direkt an Benutzer zu vergeben, werden sie Rollen zugeordnet, was die Administration vereinfacht und mehr Flexibilität ermöglicht. Eine Erweiterung dieses Modells ist das hierarchische RBAC, das Rollenhierarchien verwendet, um Zugriffsrechte zu vererben und die Administration zu erleichtern (vgl. [49]: S. 67-78; [16]: S. 415-416; [9]: S. 3-4).

RBAC definiert fünf Hauptelemente: Objekte, Aktionen, Rechte, Rollen und Benutzer. Objekte sind Ressourcen wie Dateien oder Verzeichnisse, Aktionen sind Operationen wie Schreiben oder Löschen und Berechtigungen kombinieren Objekte mit Aktionen. Rollen verknüpfen Benutzer und Berechtigungen, wobei Benutzer Rollen entsprechend ihrer Position zugewiesen werden, wie in Abbildung 2.9 dargestellt. Dieses Modell minimiert den direkten Zugriff auf Berechtigungen und bietet somit eine effektive Sicherheitslösung. Trotz seiner Vorteile erfordert RBAC eine sorgfältige Planung bei der Erstellung von Berechtigungskonzepten und kann den Arbeitsaufwand für Administratoren erhöhen (vgl. [49]: S. 67-78; [16]: S. 415-416; [9]: S. 3-4).

Core RBAC ist ein wesentlicher Bestandteil des RBAC-Modells und wird als erste Schicht in jeder Organisation implementiert, bevor fortgeschrittene Komponenten eingeführt werden. Ein Benutzer wird als Individuum betrachtet, wobei seine Rolle Funktionalität und Autorität beschreibt. Berechtigungen ermöglichen die Durchführung von Operationen auf verschiedenen Objekten, wie z.B. Lesen oder Schreiben. Objekte können vielfältig sein,

z. B. Zeilen, Tabellen, Verzeichnisse, Ansichten, Dateien oder sogar Hardware wie CPU-Zyklen, Drucker oder Speicherplatz auf Festplatten (vgl. [49]: S. 67-78; [16]: S. 415-416; [9]: S. 3-4).

Das Core-RBAC-Modell konzentriert sich auf die Zuordnung von Benutzern und Berechtigungen zu Rollen in einer flexiblen Many-to-Many-Struktur. Es ist möglich, einer Rolle mehrere Benutzer zuzuordnen und umgekehrt sowie Berechtigungen verschiedenen Rollen zuzuordnen. Es gibt jedoch nur wenig Forschung über die Feinheiten von Rechten, Rollen und deren Interaktion. Einige Experten haben das symmetrische RBAC-Modell vorgeschlagen, das Berechtigungen durch Rollenhierarchien und die Trennung von Aufgaben einschränkt (vgl. [49]: S. 67-78; [16]: S. 415-416; [9]: S. 3-4).

Neben dem Core-Modell gibt es Varianten wie Hierarchical RBAC und Constrained RBAC, die weitere Sicherheitsaspekte berücksichtigen. Diese Modelle sind besonders für Organisationen wie Gesundheitszentren relevant, da sie eine effektive Sicherheit für sensible Daten gewährleisten. Trotz der strengen Sicherheit von RBAC müssen Administratoren die Anforderungen der Aufgabentrennung beachten, um die Effizienz und Sicherheit des Systems zu gewährleisten (vgl. [49]: S. 67-78; [16]: S. 415-416; [9]: S. 3-4).

Hierarchical RBAC ist die zweite Komponente des RBAC-Modells, die auf Core RBAC aufbaut. Es implementiert Rollen mit Hilfe von Rollenhierarchiekonzepten, die auf der Unternehmenshierarchie basieren. Innerhalb des RBAC-Systems werden Rollen oft mit gemeinsamen Standardberechtigungen konfrontiert, was nicht immer die ideale Lösung darstellt. Die Rollenhierarchie verbindet identische Berechtigungen, so dass Sicherheitsadministratoren einen konsolidierten Überblick über gemeinsame Berechtigungen in verschiedenen Rollen erhalten. Einige Rollen werden jedoch separat behandelt, ohne Teil der Rollenhierarchie zu sein (vgl. [49]: S. 67-78; [16]: S. 415-416; [9]: S. 3-4).

Durch Rollenvererbung können alle Berechtigungen von übergeordneten Rollen auf untergeordnete Rollen übertragen werden, wobei untergeordnete Rollen nicht notwendigerweise die gleichen Berechtigungen wie übergeordnete Rollen haben müssen. Dies kann zu Problemen führen, wenn untergeordnete Rollen auf spezifische Berechtigungen von übergeordneten Rollen zugreifen müssen. Ohne Rollenvererbung muss der Sicherheitsadministrator wiederholt Berechtigungen gewähren oder verweigern, was eine anspruchsvolle Aufgabe darstellt. Eine hierarchische Struktur mit mehreren Kategorien wie Senior, Junior, Junior-Most und Senior-Most wird empfohlen, um die Rollenvererbung effizient zu verwalten (vgl. [49]: S. 67-78; [16]: S. 415-416; [9]: S. 3-4).

Eingeschränktes RBAC, auch bekannt als Constrained RBAC, implementiert spezifische Einschränkungen in Bezug auf die Aufgabentrennung. Diese Einschränkungen können sich auf zeit- oder ortsbasierte Zugriffsregeln beziehen. Der Hauptzweck dieser Beschränkungen besteht darin, den Zugriff auf der Grundlage bestimmter Zeitfenster und Orte zu gewähren. Die Einführung von RBAC-Beschränkungen erhöht die Informationssicherheit des Systems und schützt vor internen und externen Bedrohungen. Diese Sicherheitsbedingungen werden wie beim RBAC-Modell für die Zugriffskontrolle bestätigt (vgl. [49]: S. 67-78; [16]: S. 415-416; [9]: S. 3-4).

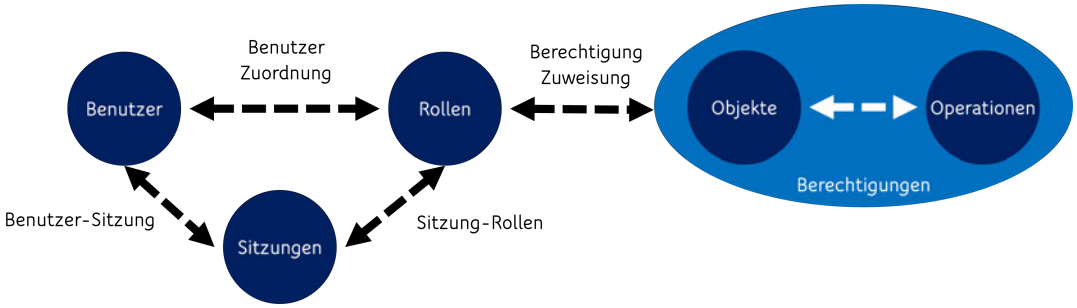



Abbildung 2.9: Abstrakte Ansicht des Core RBAC Modells (vgl. [49]: Abbildung 4.1)

## 2.5 Auswahl eines geeigneten Berechtigungsmodells

In diesem Kapitel wird die Auswahl des passenden Berechtigungsmodells für die DaFne-Plattform behandelt.

Nachfolgend die *Anforderungen* an ein Berechtigungskonzept von DaFne, welche in Kapitel 3.1.1 ausführlich behandelt werden.

**Skalierbarkeit:** Das Berechtigungskonzept muss flexibel genug sein, um Zugriffsrechte und Rollen an wachsende Anforderungen anzupassen.

**Erweiterbarkeit:** Neue Dienste sollten nahtlos in das bestehende Authentifizierungs- und Autorisierungssystem integriert werden können, um das Berechtigungskonzept zu erweitern.

**Flexibilität:** Das Berechtigungskonzept sollte eine feingranulare Steuerung der Zugriffsrechte für unterschiedliche Benutzerrollen ermöglichen.

**Überwachung/Monitoring:** Die Protokollierung von Berechtigungsänderungen und Zugriffsvorgängen ist entscheidend für die Sicherheit und Transparenz des Berechtigungskonzepts.

**Sicherheit:** Das Berechtigungskonzept muss robust sein, um sensible Daten zu schützen und unberechtigte Zugriffe zu verhindern.

**Kleines Entwicklungsteam:** Ein einfaches Berechtigungskonzept hilft dabei, unnötige Komplexität zu vermeiden und die Umsetzung für das kleine Entwicklungsteam effizient zu gestalten.

**Umgang mit großen Datenmengen:** Das Berechtigungskonzept muss sicherstellen, dass große Datenmengen effizient genutzt und verwaltet werden können, um die Nutzererfahrung zu optimieren.

Nachfolgend die *Gegebenheiten* an ein Berechtigungskonzept von DaFne, welche in Kapitel 3.3.2, 3.3.3 und 3.3.4 ausführlich behandelt werden.

In DaFne sind diverse Module und Services vorhanden, die nur für bestimmte Nutzer zugänglich sind. Dies gilt auch für die Datenquellen und Ressourcen, die nicht allen Benutzern offenstehen.

Nachfolgend die *Nutzerstruktur* an ein Berechtigungskonzept von DaFne, welche in Kapitel 3.4 ausführlich behandelt wird.

DaFne verfügt über unterschiedliche Nutzertypen zur Abgrenzung. Aufgrund seiner innovativen und flexiblen Plattformstruktur unterliegt DaFne einer sich wandelnden Nutzerstruktur, die sich in der Zukunft verändern kann.

Die verschiedenen Zugriffskontrollmodelle haben jeweils ihre eigenen Vor- und Nachteile im Hinblick auf die Anforderungen, Gegebenheiten und Nutzerstruktur. Hier sind die Vor- und Nachteile der genannten Modelle:

**Diskretionäre Zugriffskontrolle (DAC)** (vgl. [43])

*Vorteile:*

- Einfach in der Handhabung und Implementierung.

- Erlaubt benutzerdefinierte Zugriffskontrollen.

*Nachteile:*

- Sicherheitslücken, z. B. Risiko durch Malware.

- Verwaltung von Zugriffskontrollisten kann aufwändig werden.

- Begrenzt in der negativen Authentifizierung.

**Obligatorische Zugriffskontrolle (MAC)** (vgl. [14]: S.20; [7])

*Vorteile:*

- Erhöht die Sicherheit.

- Klare Regeln, wie Arbeitsprozesse abzulaufen haben.

*Nachteile:*

- Erfordert gründliche Vorbereitung und hohen Verwaltungsaufwand.

- Fortlaufende Überprüfung und Aktualisierung der Rechte sind nötig.

- Kann die Flexibilität der Nutzer einschränken.

**Regelbasierte Zugriffskontrolle (RuBAC)** (vgl. [20])

*Vorteile:*

- Erlaubt detaillierte Regeln auf unterschiedlichen Ebenen.

- Kann spezielle Beziehungen zwischen Ressourcen, Personen oder Prozessen abbilden.

*Nachteile:*

- Zeitintensive Erstellung detaillierter Regeln.

- Umständliche Anpassungen außerhalb von Standardprozeduren.

- Komplexität kann die Verwaltung erschweren.

**Attributbasierte Zugriffskontrolle (ABAC)** (vgl. [34])

**Vorteile:**

- Feiner Kontrolle über den Zugriff auf Ressourcen.

- Flexibel und anpassungsfähig an verschiedene Situationen.

*Nachteile:*

- Schwierig zu implementieren.

- Erfordert detaillierte und umfangreiche Policies.

- Verwaltung von Konten und Ressourcen kann komplex werden.

**Rollenbasierte Zugriffskontrolle (RBAC)** (vgl. [16]: S. 415-416; [49] S. 41-44 [3]; [29])

*Vorteile:*

- Geringste Privilegien (PoLP) werden berücksichtigt.

- Einfache Verwaltung durch direkte Zuordnung von Rechten zu Rollen.

- Flexibilität und effiziente Verwaltung über Rollenzuweisungen.

*Nachteile:*

- Gefahr der Überberechtigung durch Rollenzuweisungen.

- Manuelle Anpassung bei Rollen- oder Mitarbeiteränderungen erforderlich.

- Verwaltung bei wachsenden Unternehmen kann ineffizient werden.

Die Entscheidung für das adäquate Berechtigungsmodell für DaFne ist aufgrund der vielfältigen Anforderungen an eine Skalierbarkeit, Erweiterbarkeit, Flexibilität, Überwachung, Sicherheit, effizientes Handling großer Datenmengen sowie der spezifischen Nutzerstruktur eine komplexe Abwägung. Die genannten Aspekte sind von entscheidender Bedeutung, um ein effektives und gut verwaltbares Berechtigungskonzept zu gewährleisten. Da es keine universellen Richtlinien gibt, die festlegen, wann welches Modell angemessen ist, ist eine Einzelfallbetrachtung erforderlich. Jedes Modell weist spezifische Stärken und Schwächen auf, und die Vielzahl verfügbarer Lösungen erschwert die Auswahl zusätzlich.

Nach einer ausführlichen Diskussion mit dem Entwicklerteam von DaFne wurde die Entscheidung zugunsten des rollenbasierten Zugriffskontrollmodells (RBAC) getroffen, das im Jahr 2004 als ANSI-Norm 359-2004 etabliert wurde. RBAC bietet klare Vorteile wie die intuitive Replikation von Geschäftsprozessen und Organisationsstrukturen, wodurch Berechtigungen direkt an Benutzerrollen angepasst werden können. Dies ermöglicht eine effiziente Verwaltung der Benutzerrechte und eine flexible Anpassung bei Rollenwechseln

oder neuen Nutzertypen. Jedoch besteht das Risiko einer Überberechtigung bei uneingeschränkter Rollenzuweisung, das durch sorgfältige Planung und regelmäßiges Monitoring abgemildert werden muss (vgl. [16]: 415-416; [49] S. 41-44, [29]).

Die nutzerbasierte Zugriffskontrolle, wie sie in rollenbasierten Modellen angewendet wird, bietet klare Vorteile hinsichtlich der direkten Zuordnung von Identitäten zu Ressourcen. Dies bildet eine transparente und effiziente Struktur, erfordert jedoch eine umfassende Verwaltung, insbesondere in expandierenden Systemen. Das Abwägen zwischen Sicherheit, Verwaltbarkeit und Nutzerfreundlichkeit bleibt eine essentielle Herausforderung (vgl. [16]: S. 415-416; [49] S. 41-44).

Die integrierte Berücksichtigung des Prinzips der geringsten Privilegien (PoLP) im RBAC-Modell hilft, Sicherheits- und Compliance-Risiken zu minimieren. Dieses Prinzip gewährleistet, dass Nutzer nur die für ihre jeweilige Rolle erforderlichen Rechte erhalten, was eine wichtige Maßnahme gegen Rechteakkumulation und daraus resultierende Sicherheitsrisiken darstellt (vgl. [3]).

Die Diskretionäre Zugriffskontrolle (DAC) bietet Einfachheit in der Handhabung und ermöglicht benutzerdefinierte Zugriffskontrollen. Dennoch existieren Sicherheitslücken, wie das Risiko durch Malware, sowie eine herausfordernde Verwaltung von Zugriffskontrollisten und Begrenzungen in der negativen Authentifizierung (vgl. [43]).

Mandatorische Zugriffskontrollen (MAC) und regelbasierte Zugriffskontrollen (RuBAC) können spezifischere Zugriffsregeln ermöglichen, jedoch aufgrund ihrer Komplexität und intensiven Verwaltung besondere Anforderungen an das Entwicklungsteam und Überwachungsprozesse stellen (vgl. [7]; [14]: S.20; [20]). Attributebasierte Zugriffskontrolle (ABAC) bietet eine feinere Kontrolle über Zugriffsrechte, erfordert jedoch eine umfangreiche Policy-Verwaltung und Implementierungsaufwand (vgl. [34]).

## 2.6 Berechtigung

Im nachfolgenden Unterkapitel erfolgt eine Auseinandersetzung mit den auf der DaFne-Plattform vorhandenen Berechtigungen.

### 2.6.1 Erklärung

Basierend auf Tsolkas (vgl. [49]: S. 1-4) wird folgende Erklärung verwendet.

„Ein wesentliches Ziel von unternehmensweiten Berechtigungs- und Rollenkonzepten ist die effiziente und elegante Steuerung von Berechtigungen. Dies basiert auf einem systematischen Ansatz zur Organisation und Strukturierung von Berechtigungen unter Verwendung von Design- oder Steuerungselementen, die Identitäten und Berechtigungen zugeordnet werden."

Das Hauptziel der Organisation von Berechtigungen besteht darin, einzelne Zugriffsrechte in übergeordneten Gruppen zusammenzufassen, um eine schnellere und eindeutigere Zuordnung zu ermöglichen. Durch eine sorgfältige Planung dieser Elemente können die gewünschten Steuerungsziele auch dann erreicht werden, wenn eine große Anzahl von Berechtigungen verwaltet, miteinander verknüpft oder verschachtelt werden muss.

Das Grundelement für die Strukturierung von Berechtigungen ist die Berechtigung selbst. Da sowohl die Ausführung von IT-Funktionen als auch der Zugriff auf IT-Inhalte Formen des IT-Zugriffs sind, spricht man in diesem Zusammenhang von Zugriffsberechtigungen. Wenn Berechtigungen diskutiert werden, die sich auf IT-Funktionen beziehen, wird häufig der Begriff Ausführungsberechtigung verwendet. Beispielsweise erfordert das Starten einer ausführbaren Datei eine solche Berechtigung.

Jede Berechtigung besteht aus zwei Hauptkomponenten:

**1. der Ressource, auf die zugegriffen werden soll.**

Diese Komponente gibt an, welche Ressource durch die Berechtigung geschützt wird. Eine Ressource kann eine funktionale Ressource, eine inhaltliche Ressource oder beides sein und sowohl global als auch spezifisch definiert werden.

**2. die Operation, die an dieser Ressource ausgeführt werden darf.**

Die zweite Komponente definiert, welche Operation in Bezug auf die Ressource erlaubt oder verboten ist.

Wenn ein Subjekt mit einem Objekt interagiert, muss es dazu berechtigt sein. Eine Berechtigung gibt dem Subjekt die Erlaubnis, mit dem Objekt zu interagieren und spielt eine Schlüsselrolle beim Schutz des Objekts. Berechtigungen reichen von physischen Zugangsrechten zu einem realen Objekt über physische Nutzungsrechte an einem Objekt bis

hin zu Zugriffsrechten auf Funktionen und Inhalte in einer IT-Umgebung. Beispielsweise ist das Starten einer bestimmten Software eine Ausführungsberechtigung.

**Zutrittsberechtigungen:** Diese Art von Berechtigung erlaubt den physischen Zugriff auf ein reales Objekt.

**Nutzungsberechtigungen:** Legen fest, wie ein physisches Objekt genutzt werden darf.

**Zugriffsberechtigungen:** Dieser Begriff umfasst den Umstand, dass der Zugriff auf Funktionen und Inhalte innerhalb der IT-Infrastruktur gewährt wird, was den Eintritt in eine virtuelle Umgebung bedeutet.

**Ausführungsberechtigungen:** Erlauben die Nutzung einer bestimmten IT-Funktion, z.B. das Starten eines Textverarbeitungsprogramms.

Die Stelle im Arbeitsprozess, an der die Prüfung einer Berechtigung stattfindet, wird als *Berechtigungspunkt* bezeichnet. Nach erfolgreicher Prüfung erhält das anfragende Subjekt Zugriff auf die gewünschte Operation am Objekt. Diese Prüfung kann unmittelbar vor dem Start einer Anwendung oder während des Zugriffsversuchs auf ein Objekt erfolgen. Die Abbildung 2.10 zeigt die Beziehung zwischen einem anfragenden Subjekt und einem Objekt.

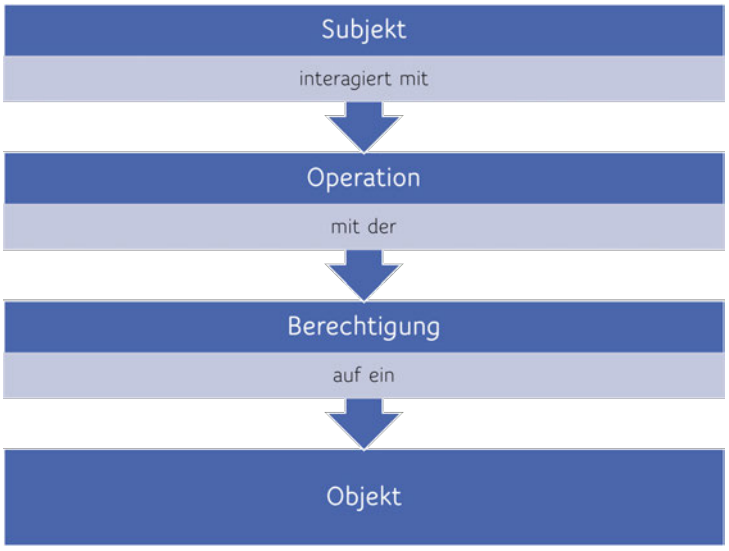



Abbildung 2.10: Berechtigung als Vermittler zwischen dem Subjekt und dem Objekt der Operation

### 2.6.2 Operationen

In der Informatik werden Interaktionen von Subjekten mit Objekten als Operationen bezeichnet, für die bestimmte Berechtigungen erforderlich sind. Inhaltsressourcen können durch verschiedene Operationen beeinflusst werden. Es gibt verschiedene Berechtigungsebenen, da weitergehende Berechtigungen (z.B. zum Löschen) oft auch grundlegendere Berechtigungen (z.B. zum Lesen) implizieren. Hierfür wird auch der Begriff Berechtigungsstufen verwendet. Der folgende Abschnitt stellt die grundlegenden Operationen vor gemäß (vgl. [49]: S. 3-7):

**Entdecken** (*engl. Detect*):

Ermöglicht das Feststellen der Existenz einer Ressource ohne weitere Zugriffsrechte.

**Suchen** (*engl. Search/Find*):

Ermöglicht die Suche nach einer Ressource, auch ohne genaue Kenntnis ihres Speicherorts. Die Suche kann auch breit angelegt sein.

**Vergleichen** (*engl. Compare*):

Erlaubt das Vergleichen, z.B. das Überprüfen von Passwörtern oder das Vergleichen von Dateigrößen. Diese Berechtigung baut auf der Berechtigung Entdecken auf.

**Anzeigen** (*engl. Show*):

Ermöglicht die Auflistung einer Ressource als Ganzes (Black Box), ohne sie zu öffnen. Dieses Recht erweitert die Such- und Vergleichsmöglichkeiten.

**Lesen** (*engl. Read*):

Ermöglicht den Zugriff auf den Inhalt einer Ressource sowie die Anzeige von Attributen und Verwaltungsinformationen.

**Hinzufügen** (*engl. Add*):

Ermöglicht das Hinzufügen von Inhalten zu einer Ressource, wodurch der Zustand der Ressource verändert wird. Dies kann insbesondere für die IT-Sicherheit von Bedeutung sein.

**Ändern** (*engl. Change/Modify*):

Erlaubt das Ändern von Inhalten, was das Hinzufügen neuer und das Entfernen bestehender Inhalte beinhaltet. Diese kritische Operation kann Inhalte manipulieren oder zerstören.

**Löschen** (*engl. Delete*):

Erlaubt das vollständige Entfernen eines Objekts. Meistens beinhaltet diese Berechtigung auch das Recht, das Objekt zu entdecken, zu lesen und zu verändern.

**Ausführen** (*engl. Execute*):

Erlaubt das Starten ausführbarer Inhalte wie Programme oder Skripte und baut auf den Rechten Entdecken und Anzeigen auf.

### 2.6.3 Stufen

Der folgende Abschnitt ist adaptiert aus (vgl. [49]: S. 10-11).

Die Vergabe von Berechtigungen kann durch die Verwendung von Berechtigungsstufen erleichtert werden. Diese Stufen reichen von keiner Berechtigung bis zu allen Berechtigungen.

Sobald die Berechtigungsstufen definiert sind, kann die Berechtigung durch Angabe der Nummer der entsprechenden Stufe erteilt werden, die der Identität über ein berechtigungssteuerndes Element (z. B. eine Rolle oder eine Gruppe) zugewiesen wird. Wenn verschiedene Kombinationen von Operationen erlaubt sein sollen, muss der resultierende Wert eindeutig angeben, welche Operationen enthalten sind. Für jede Identität wird festgehalten, welche Berechtigungsstufe ihr zugeordnet ist. Der Vorteil der Verwendung von Berechtigungsstufen liegt darin, dass einzelne Berechtigungen nicht mehr separat behandelt werden müssen. Durch die Zuordnung einer Stufe ist bereits klar, welche Berechtigungen die Identität hat, was die Vergabe von Berechtigungen vereinfacht.

Einige der dargestellten Berechtigungen beinhalten weitere Berechtigungen und stellen somit eine erste Bündelung dar. Die Berechtigungsstufen setzen an diesem Punkt an und fassen Gruppen von Berechtigungen zusammen, siehe Abbildung 2.11. Dies erleichtert eine schnelle und übersichtliche Zuordnung von Berechtigungen zu Objekten. Eine weitere optionale Eigenschaft von Berechtigungsstufen ist, dass Funktionen und Inhalte integriert werden können.

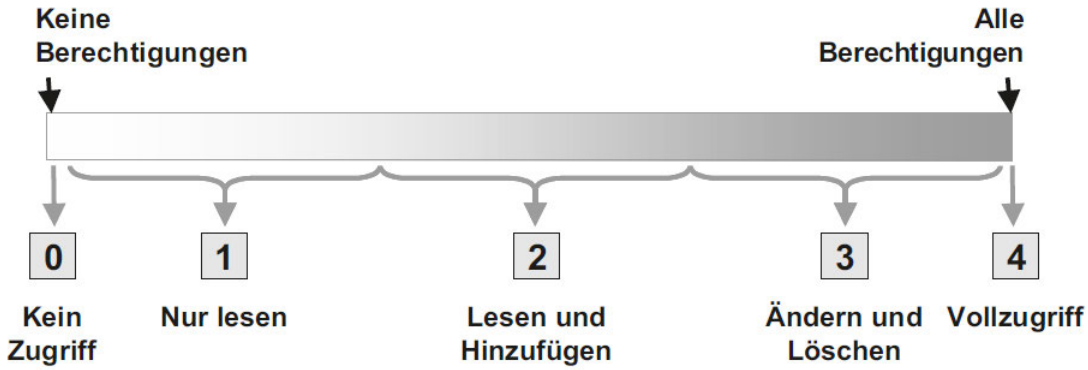



Abbildung 2.11: Berechtigungsstufen (vgl. [49]: Abbildung 1.5)

**Stufe 0**

Gewährt einem anfragenden Subjekt keinerlei Zugriff auf ein zu schützendes Objekt. Ob dennoch Detect-, Search-, Compare- oder Show-Berechtigungen bestehen, hängt von der Schutzstufe des Objekts ab.

**Stufe 1**

Erlaubt den Lesezugriff auf ein Objekt und vergibt gleichzeitig die Read- und Show-Berechtigungen, die Detect-, Search- und Compare-Berechtigungen implizieren.

**Stufe 2**

Erlaubt zusätzlich zu Stufe 1 die Berechtigung Add und kann je nach Schutzanforderungen bereits die Berechtigung Execute enthalten.

**Stufe 3**

Erweitert alle Berechtigungen der Stufe 2 um die Berechtigungen Modify und Delete.

**Stufe 4**

Hat keine Einschränkungen mehr und enthält alle Berechtigungen für die Funktionen, Eigenschaften und Inhalte des Objekts, aufbauend auf den Stufen 1 bis 3.

In Abbildung 2.12 sind die Operationen der Berechtigungsstufen zugeordnet, um die Gruppierung zu verdeutlichen. Die bisher vorgestellten Stufen können beliebig schwach oder stark granuliert und weiter angepasst werden.


Abbildung 2.12: Berechtigungsstufen inklusive der jeweiligen Operationen (vgl. [49]: Abbildung 1.5)

**Nur Lese- und Ausführungsrechte:**

Da die Berechtigungen zum Ausführen und Hinzufügen unabhängig voneinander sind, ermöglicht dies eine feinere Unterscheidung zwischen Level 1 und Level 2.

**Ändern ohne Löschen:**

Diese Unterscheidung wird wichtig, wenn bestimmte Benutzer nur die Berechtigung haben sollen, Objekte zu ändern, aber nicht zu löschen. Dies hilft, das versehentliche oder absichtliche Löschen von Ordnern, Dokumenten oder anderen Dateien zu verhindern.

**Verfeinerung der Ebenen 3-4:**

Eine Verfeinerung innerhalb dieser Stufen kann sich auf zusätzliche Operationen beziehen, die vom Benutzer ausgeführt werden können. Auf Stufe 3 sind nur Lese- und Ausführungsrechte erlaubt, während auf Stufe 4 auch Änderungs- und Löschrechte vergeben werden können.

## 2.7 Zugriffsregeln

Im nachfolgenden Unterkapitel erfolgt eine Auseinandersetzung mit den wesentlichen Zugriffsregeln.

### 2.7.1 Definition

Beim Konzept der rollenbasierten Zugriffskontrolle verknüpft eine Zugriffsregel eine bestimmte Kombination aus Domäne, Objekttyp, Lebenszyklusstatus dieser Objekte und Benutzergruppe mit definierten Berechtigungen. Sie legt fest, welche Zugriffsrechte einzelne Benutzer, Gruppen, Rollen oder ganze Organisationsstrukturen auf Objekte eines bestimmten Typs in einem bestimmten Zustand innerhalb einer Domäne haben. Darüber hinaus ermöglicht das Konzept, zusätzlich zu den Regeln, die für bestimmte Teilnehmer definiert sind, Berechtigungen für bestimmte Szenarien unter Verwendung definierter Methoden zu definieren, um eine flexible und dynamische Zugriffsverwaltung im Rahmen der rollenbasierten Zugriffskontrolle zu gewährleisten (vgl. [5]).

### 2.7.2 Beschreibung der drei wichtigsten Zugriffsregeln

Die drei wichtigsten Zugriffsregeln für ein rollenbasiertes Berechtigungskonzept (vgl. [4]).

**Zuweisung von Rollen:** Benutzer haben nur dann Zugriff auf bestimmte Rechte, wenn ihnen explizit eine Rolle zugewiesen wurde.

**Rollenbasierte Autorisierung:** Es muss bestätigt werden, dass ein Benutzer berechtigt ist, eine bestimmte Rolle einzunehmen, um sicherzustellen, dass Benutzer nur Zugriff auf die Rollen haben, für die sie explizit autorisiert wurden.

**Autorisierung von Privilegien:** Ein Benutzer ist berechtigt, bestimmte Rechte auszuüben, vorausgesetzt, dass diese Rechte durch die zugewiesene Rolle und die entsprechende Autorisierung legitimiert sind.

### 2.7.3 Überwachungsmechanismus

Nach (vgl. [49]: S. 187, 228, 254, 312-314; [27]: S. 149-150; [28]: S. 248-249; [11]: S. 161-162) dient ein detailliertes Logging-Konzept als wirksamer Kontroll- bzw. Überwachungsmechanismus, um sicherzustellen, dass ein rollenbasiertes Berechtigungskonzept den Anforderungen des Datenschutzes und der Datensicherheit genügt. Im Folgenden sind einige Aspekte aufgeführt, wie das Logging-Konzept dazu beitragen kann.

**Compliance:** Durch das Logging werden alle Zugriffe, Berechtigungsänderungen und Aktionen protokolliert. Diese Informationen können genutzt werden, um zu überprüfen, ob die Vergabe von Rollen und Berechtigungen den Compliance-Anforderungen entspricht. Beispielsweise kann mit Hilfe von Audit-Logs überprüft werden, ob nur autorisierte Benutzer auf geschützte Daten zugreifen.

**Richtlinien und Standards:** Das Logging-Konzept ermöglicht die Überwachung der Einhaltung interner Richtlinien und externer Standards. Protokolle von Berechtigungsänderungen können helfen, sicherzustellen, dass Rollenzuweisungen den definierten Richtlinien und Best Practices entsprechen. Beispielsweise können regelmäßige Berichte über Zugriffe und Änderungen Aufschluss darüber geben, ob Standards eingehalten werden.

**Normen:** Durch die Überwachung von Logs können Organisationen sicherstellen, dass Normen wie ISO 27001 oder NIST oder DSGVO/GDPR etc. eingehalten werden. Logs können verwendet werden, um Zugriffe und Berechtigungen zu verfolgen und sicherzustellen, dass Datenschutz- und Sicherheitsanforderungen eingehalten werden.

Durch die Überwachung und Analyse der Logs im Rahmen des Logging-Konzepts können potenzielle Sicherheitsvorfälle identifiziert, Schwachstellen aufgedeckt und die Einhaltung von Richtlinien, Standards und Normen sichergestellt werden. Das Logging-Konzept fungiert somit als zentraler Überwachungsmechanismus, um die Sicherheit und Compliance des rollenbasierten Berechtigungskonzeptes zu gewährleisten.

## 2.8 Provisioning

Im nachfolgenden Unterkapitel erfolgt eine Auseinandersetzung mit den wesentlichen Aspekten der Provisoning.

### 2.8.1 Definition

Nach (vgl. [49]: S. 115) wird Provisioning vorgestellt.

Provisioning bezieht sich in der IT und Telekommunikation auf die Versorgung von Anwendern oder IT-Systemen mit Daten und Ressourcen durch einen Prozess. Dies umfasst die Verwaltung von Benutzerkonten, Benutzeridentitäten, Zugriffskontrollen und Ressourcenzuweisungen für IT-Systeme. Früher wurden Daten manuell auf Laufzetteln

erfasst und dann in IT-Systeme eingegeben, was zu Unterbrechungen führte. Durch Provisioning-Prozesse und -Systeme werden Daten nahtlos geladen und Ressourcen bereitgestellt, um Datenbrüche zu vermeiden.

Der Begriff Provisioning tauchte erstmals im Zusammenhang mit Identitätsmanagement, Verzeichnisdiensten und Single Sign-On auf. Diese Technologien automatisieren IT-Prozesse wie die Benutzerverwaltung durch Provisioning. Die Kernelemente des Provisioning sind das User-Provisioning und das Ressource-Provisioning. Im Telekommunikations- und Mobilfunkbereich hat sich Provisioning zu komplexen Service-Delivery-Plattformen entwickelt. De-Provisioning bezieht sich auf das Entfernen von Benutzerinformationen und Systemressourcen.

Es gibt verschiedene Arten von Provisioning, wie z.B. User- und Ressource-Provisioning, Server-Provisioning, Service-Provisioning, Mobile Subscriber-Provisioning, Mobile Content-Provisioning und Cloud-Provisioning. Diese dienen der technischen Verwaltung von Benutzern, Ressourcen und Verbindungen in Netzwerk- und Systemumgebungen. Ziel eines Provisioning-Systems ist es, autorisierten Benutzern die benötigten Ressourcen in Netzwerk- oder Systemumgebungen zur Verfügung zu stellen, einschließlich der Verwaltung von Konten, der Zuweisung von Zugriffsrechten und der Netzwerkverbindung. Für diese Arbeit sind nur die folgenden Arten relevant: User- und Ressource-Provisioning, Server-Provisioning und Service-Provisioning, die auch in den folgenden Abschnitten kurz vorgestellt werden.

### 2.8.2 Benutzer-Provisioning

Das User-Provisioning-System, auch Benutzerverwaltungssystem genannt, verwaltet das Hinzufügen oder Löschen von Benutzer- und Servicekonten. Es setzt die Anforderungen der einzelnen Prozesse zur Erstellung, Deaktivierung und Löschung von Identitäten um. Darüber hinaus ordnet das User-Provisioning-System die dokumentierten Eigenschaften einer Identität dem entsprechenden Account auf Systemebene zu (vgl. [49]: S. 116, 117-120).

### 2.8.3 Ressource-Provisioning

Das Resource-Provisioning-System konfiguriert die Zugriffsrechte der Accounts auf die Ressourcen unter Berücksichtigung der Berechtigungsprofile und setzt gleichzeitig die notwendigen Sicherheitsanforderungen technisch um (vgl. [49]: S. 116, 120-123).

### 2.8.4 Server-Provisioning

Server-Provisioning beschreibt die Konfiguration eines Servers auf Basis der Geschäfts- und Sicherheitsanforderungen sowie des Verwendungszwecks. Darüber hinaus wird der Server über seinen gesamten Lebenszyklus hinweg betreut, was Audits, die Überprüfung des gesetzeskonformen Betriebs (Compliance), Sicherheitsbewertungen in Bezug auf Schwachstellen und Aktualisierungen durch Updates umfasst. Server Provisioning ist die zielgerichtete Einrichtung und Verwaltung eines Servers, die entweder durch ein automatisiertes System oder manuell durchgeführt wird (vgl. [49]: S. 123-124).

### 2.8.5 Service-Provisioning

Beim Service-Provisioning werden standardisierte Servicepakete zwischen einem Service Provider und einem Kunden bereitgestellt. Dazu gehören Aufgaben wie das zentrale Sammeln von Firewall-Logfiles oder das Betreiben eines Internet-Bewerberportals. Die Dienste können entweder extern von einem Dienstleister oder intern von der IT-Abteilung eines Unternehmens erbracht werden. Typische Dienste sind DNS- und E-Mail-Dienste, Antivirenlösungen, Patch-Management, Monitoring, Zugangskontrollen und vieles mehr. Das Prinzip des Service Provisioning wird auch innerhalb von Unternehmen für interne IT-Dienste angewandt (vgl. [49]: S. 124-125).

# 3 Anforderungsanalyse - Requirements Engineering

Im Rahmen dieses Kapitels erfolgt eine Anforderungsanalyse, welche die Darstellung aller wesentlichen Anforderungen zum Ziel hat, die für die Konzeption von Relevanz sind.

## 3.1 Nichtfunktionale Anforderungen

Im vorliegenden Unterkapitel erfolgt eine Auseinandersetzung mit den nichtfunktionalen Anforderungen.

### 3.1.1 Anforderungen

Die folgende Zusammenfassung gibt einen Überblick über die identifizierten Anforderungen an die Plattform und beschreibt für jede Anforderung kurz, wie sie sich auf das Berechtigungskonzept auswirken kann und worauf ggf. zu achten ist.

Im vorliegenden Paper (vgl. [33]: S. 7) wurden folgende übergreifende Anforderungen identifiziert, die es zu berücksichtigen gilt:

**Skalierbarkeit:** Die Plattformarchitektur sollte skalierbar gestaltet sein, um die Parallelisierung von Trainingsverfahren für maschinelles Lernen zu ermöglichen. Wichtig ist, dass die Plattform auf gängigen Cloud-Plattformen einfach erweiterbar ist und sowohl GPU- als auch CPU-Ressourcen effizient nutzt. In Bezug auf das Berechtigungskonzept sollte die Skalierbarkeit berücksichtigt werden, um sicherzustellen, dass Zugriffsrechte und Rollen flexibel an wachsende Anforderungen angepasst werden können. Die Veränderung kann durch eine Zunahme der Anzahl an Benutzern, den Bedarf an verschiedenen Berechtigungen oder die Notwendigkeit umfassenderer Zugriffskontrollen bedingt sein.

**Erweiterbarkeit:** Die Plattformarchitektur sollte so gestaltet sein, dass die Integration weiterer Dienste ohne großen Aufwand möglich ist. Im Hinblick auf das Berechtigungskonzept ist sicherzustellen, dass neue Dienste problemlos in das bestehende Authentifizierungs- und Autorisierungssystem integriert werden können. Es ist wichtig sicherzustellen, dass die Personen, die für die Erweiterung berechtigt sein müssen, entsprechend zugewiesen werden.

**Flexibilität:** Flexibilität bezieht sich auf die Möglichkeit der Nutzung unterschiedlicher Hardware-Ressourcen sowie auf die Unterstützung unterschiedlicher Benutzerrollen mit spezifischem Know-how im Umgang mit Generierungsmethoden. Bei der Planung des Berechtigungskonzepts sollte die Möglichkeit einer feingranularen Steuerung der Zugriffsrechte für unterschiedliche Benutzerrollen berücksichtigt werden.

**Überwachung bzw. Monitoring:** Eine zentrale Anforderung an die Plattformarchitektur ist die Implementierung eines umfassenden Überwachungs- und Monitoringsystems. Dieses ermöglicht die transparente Erfassung von Statusinformationen der Dienste, was wiederum die Plattform-Performance verbessern und die User Experience optimieren kann. Im Zusammenhang mit dem Berechtigungskonzept ist es wichtig, dass auch die Protokollierung von Berechtigungsänderungen und Zugriffsvorgängen gewährleistet ist.

**Sicherheit:** Die Sicherheit der Plattform gegenüber externen Bedrohungen hat höchste Priorität. Dazu gehören Maßnahmen wie die Authentifizierung der Nutzer sowie die Gewährleistung eines sicheren Umgangs mit Daten, sei es bei der Datengenerierung oder -speicherung. Das Berechtigungskonzept muss robust genug sein, um sensible Daten zu schützen und unberechtigte Zugriffe zu verhindern.

Konkrete Anforderungen, die sich aus der Anforderungsanalyse ergeben:

**Kleines Entwicklungsteam:** Da die Entwicklung dieses Projekts mit einem Team von ca. 5 Personen geplant ist, ist es wichtig, unnötige Komplexität und Verwicklungen zu vermeiden. Dieser Ansatz zielt darauf ab, den Entwicklungsaufwand für das bestehende Team zu minimieren. Es wurde bewusst auf zusätzliche Komplexität verzichtet, um die Umsetzung des Projektes für das Team effizient und gut handhabbar zu gestalten.

**Umgang mit großen Datenmengen:** Ein weiterer wichtiger Aspekt ist die Fähigkeit der Plattform, mit großen Datenmengen umzugehen. Diese Daten können entweder von den Nutzern hochgeladen werden, um ML-Modelle zu trainieren, oder durch generative

Verfahren erzeugt und in der Plattform verwendet werden. Die Daten müssen von verschiedenen Diensten genutzt und effizient zugänglich gemacht werden, um eine optimale Nutzererfahrung zu gewährleisten.

Aufgrund des kleinen Entwicklungsteams und der Anforderung an eine effiziente Datenverarbeitung wurde bewusst auf eine übermäßige Komplexität des Berechtigungskonzepts verzichtet. Diese strategische Entscheidung zielt darauf ab, die Prozesse schlank und überschaubar zu halten, um die Projektdurchführung für das Team so reibungslos wie möglich zu gestalten. Aspekte wie die Implementierung von Sicherheitspaketen und spezielle Standards im Berechtigungskonzept wurden in Anbetracht der Teamgröße und der Projektanforderungen bewusst nicht berücksichtigt.

### 3.1.2 Identifikation der Services

Diese Arbeit bezieht sich auf die im Paper vorgestellte Architektur (vgl. [33]: S. 7-8) und verwendet diese als Grundlage für die Entwicklung des Berechtigungskonzepts. Da sich das Projekt zum Zeitpunkt der Erstellung dieser Arbeit noch in der Entwicklung befindet, sind zukünftige Änderungen an der Architektur möglich, die in dieser Arbeit nicht berücksichtigt werden können.

Bei der Entwicklung einer geeigneten Architektur wurden folgende Komponenten für die Umsetzung der Plattform als Microservice-Architektur identifiziert, welche in Abbildung 3.1 dargestellt sind.

Die Architektur von DaFne ist Microservice-orientiert und umfasst folgende Dienste:

**GUI - Graphical User Interface:** Diese Schnittstelle gewährleistet die Parametrisierung der Algorithmen und ermöglicht dem Benutzer den Zugriff und die Visualisierung der Ergebnisse der Datengenerierung. Die Benutzeroberfläche muss benutzerfreundlich gestaltet sein, um eine flexible Nutzung der Datengenerierung zu gewährleisten.

**Evaluation Manager:** Diese Komponente ermöglicht die Auswertung der Ergebnisse der Datengenerierung. Metriken wie Korrelationen, bivariate Verteilungen und Genauigkeit können verwendet werden, um den synthetischen Datensatz zu bewerten. Die Ergebnisse werden in einem Qualitätssicherungsreport zusammengefasst.

**Data Manager:** Diese Komponente umfasst das Datenmanagement, die Datenintegration, die Qualitätsbewertung sowie die Datenharmonisierung und das Metadatenmanagement.

**Generation Manager:** Diese Komponente für das Modellmanagement ermöglicht das Hinzufügen, Parametrisieren, Entwickeln und ggf. Trainieren von Modellen des maschinellen Lernens. Die Modelle werden im Modellmanagement gespeichert und können später wiederverwendet werden.

**Orchestrator:** Die Orchestrierungskomponente stellt die geordnete Ausführung der einzelnen Dienste unter Berücksichtigung ihrer Abhängigkeiten sicher. Dabei koordiniert sie die einzelnen Schritte der beschriebenen Komponenten wie Daten- und Modellmanagement, so dass die Generierungsengine schließlich die reine Datensynthese erfolgreich durchführen kann.

**Authentication:** Diese Komponente ist für die Überprüfung der Identität der registrierten Benutzer verantwortlich.

**Personalisation:** Nach erfolgreicher Authentifizierung können Benutzer ihre personalisierten Ressourcen über einen Personalisationsservice verwalten und einsehen. Diese Ressourcen können z.B. Datensätze enthalten, die als Input für die Datensynthese dienen.

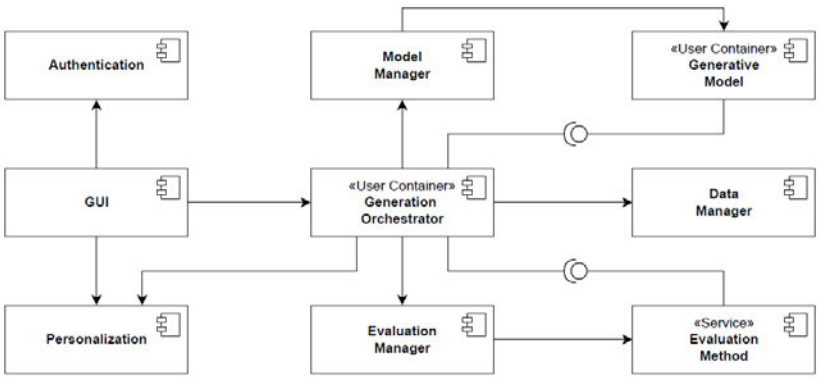



Abbildung 3.1: Komponenten der DaFne-Plattform, die den prototypischen Anwendungsfall zur Reproduction von tabellarischen Daten implementieren (vgl. [33]: Figure 3)

### 3.1.3 Kommunikation der Services

Im Folgenden wird begründet, warum jeder einzelne Service mit den anderen Services kommunizieren muss, basierend auf dem Paper (vgl. [33]: S. 9-10).

Der Benutzer authentifiziert sich über die grafische Benutzeroberfläche (GUI) mittels der Authentifizierungskomponente. Jeder Benutzer erhält seine eigene Docker-Container-Generierungskomponente, um Skalierbarkeit und Konsistenz zu gewährleisten. Wie bereits beschrieben, koordiniert diese Komponente die Kommunikation mit anderen Diensten und führt den Trainingsalgorithmus aus. Um ein generatives ML-Modell zu verwenden, fordert das Generierungssystem das Modell vom Model Manager an. Dieser verwaltet Docker-Images verschiedener implementierter Modelle und startet für den Benutzer den entsprechenden Docker-Container des angeforderten Modells. Über definierte REST-Schnittstellen können Parameter und Gewichtungen des Modells abgefragt und gesetzt werden, um das Modell zu starten und zu trainieren.

Evaluationsmetriken werden vom Evaluation Manager angefordert, der auf die einzelnen Docker Container der entsprechenden Metriken verweist. Auf diese Weise implementiert das Generierungssystem den Trainingsprozess wie folgt: Das Modell wird über x Epochen trainiert, die synthetischen Daten werden mit den ausgewählten Metriken gegen reale Daten verglichen und die Ergebnisse zwischengespeichert. Dies wird y-mal wiederholt, so dass das Modell insgesamt über x*y Epochen trainiert wird. Anschließend wählt das Generierungssystem die Epoche aus, in der die besten Ergebnisse für die angegebene Zielmetrik erzielt wurden, und gibt die entsprechenden (besten) synthetischen Daten an den Benutzer zurück. Die Personalisierungskomponente speichert das beste Modell als Kombination von Parametern und Gewichten.

Die Kommunikationspfade zwischen den verschiedenen Komponenten sind wie folgt definiert:

**GUI und Authentification:** Die GUI muss mit dem Authentifizierungsdienst kommunizieren, um Benutzer zu authentifizieren.

**GUI und Personalisation:** Die GUI muss mit dem Personalisierungsdienst interagieren, um private Benutzerdaten zu erhalten und benutzerspezifische Daten abzurufen.

**GUI und Orchestrator:** Die GUI muss mit dem Orchestrator-Dienst kommunizieren, um den Generierungsprozess zu starten und die Ergebnisse nach Abschluss des Prozesses zu erhalten.

**Orchestrator und Data Manager:** Der Orchestrator muss die Datenpositionen in der Datenbank vom Datenmanager erhalten, um sie an die Container für Generierung und Auswertung weiterzuleiten.

**Orchestrator und Generation Manager:** Der Orchestrator muss mit dem Generierungsmanager kommunizieren, um das Container-Image für die vom Benutzer ausgewählte Generierungsmethode zu erhalten.

**Orchestrator und Evaluation Manager:** Der Orchestrator muss mit dem Evaluation Manager-Dienst kommunizieren, um das Container-Image für die vom Benutzer ausgewählte Evaluationsmethode zu erhalten.

**Generation Container und Data Manager:** Der Generierungscontainer benötigt Daten vom Datenmanager für die Generierung, und der Datenmanager benötigt die generierten Daten für Aggregationszwecke und Ähnliches.

**Generation Container und Evaluation Container:** Der Generationscontainer muss mit dem Evaluationscontainer kommunizieren, um die generierten Daten zur Evaluation zu übermitteln.

**Evaluation Container und Data Manager:** Der Evaluationscontainer muss die Evaluationsberichte in der Datenbank speichern und kann sie an den Datenmanager zur Speicherung senden.

**Data Manager und Orchestrator:** Der Orchestrator benötigt Ergebnisse vom Datenmanager, um diese über die GUI für den Benutzer sichtbar zu machen.

### 3.1.4 Beispielhafter Ablauf einer Datengenerierung

Gemäß dem DaFne-Papier (vgl. [33]) läuft die Datengenerierung wie folgt ab:

Der Benutzer meldet sich über die grafische Benutzeroberfläche (GUI) an oder registriert sich ggf. Der GUI-Dienst muss dazu mit dem Authentifizierungsdienst kommunizieren.

Der GUI-Dienst ruft die gespeicherten Benutzerdaten vom Personalisierungsdienst ab.

Über die GUI kann der Benutzer dann Eingaben vornehmen, um z. B. einen Datengenerierungsprozess zu starten. Dazu sind verschiedene Eingaben erforderlich, wie die Auswahl einer Generierungsmethode, ggf. mehrerer Auswertemethoden und Daten für die Generierungsmethode.

Wenn der Benutzer den Generierungsprozess startet, wird der Orchestrator-Dienst vom GUI-Dienst darüber informiert.

Der Orchestrator-Dienst erhält vom Datenverwaltungsdienst die Position der vom Benutzer ausgewählten Daten in der Datenbank.

Zusätzlich benötigt der Orchestrator-Dienst die Daten für die ausgewählte Generierungsmethode vom Generierungsmanager-Dienst, z.B. Docker-Images, um Container für die Implementierung der Methode zu starten.

Der Orchestrator-Dienst startet dann einen Container mit dem entsprechenden Docker-Image. Dieser Generierungscontainer verwendet die erforderlichen Daten aus der Datenbank, um sie zu verarbeiten oder neue Daten zu generieren.

Parallel dazu erhält der Orchestrator Service vom Evaluation Manager Service die notwendigen Informationen über die Evaluationsmethoden und startet einen Evaluationscontainer mit dem entsprechenden Docker-Image, der die generierten Daten aus dem Generierungscontainer evaluiert.

Nach Abschluss der Generierung und Evaluation speichert der Evaluationscontainer die Berichte in der Datenbank. Der Orchestrator Service informiert den Evaluation Container, wenn der Generierungsprozess abgeschlossen ist.

Der Datenmanager informiert den Orchestrator-Dienst über den Abschluss des Prozesses, und der Orchestrator-Dienst liefert die Daten und Berichte über den GUI-Dienst an den Benutzer.

Der GUI-Dienst speichert nun die Benutzerdaten im Personalisierungsdienst, z.B. ML-Modellinformationen wie Gewichte und Parameter.

## 3.2 Übertragbarkeit und Erweiterbarkeit

Im Folgenden wird die Übertragbarkeit und Erweiterbarkeit des konzeptionierten Berechtigungsmodells für zukünftige ähnliche Projekte behandelt.

Das konzipierte Berechtigungsmodell sowie der zugehörige Entwicklungsprozess bieten eine robuste Grundlage, die sich effektiv auf ähnliche Probleme oder Anwendungskontexte

übertragen lässt. Dies trifft insbesondere auf andere Plattformen zu, die ähnliche Herausforderungen bewältigen müssen, sowie auf datenintensive Anwendungen. Die breite Anwendbarkeit des entwickelten Modells basiert auf der Nutzung bewährter Techniken des Requirements Engineerings. Der detaillierte Ausarbeitungsprozess diente als Grundlage für die Auswahl eines geeigneten Modells aus den anfänglich vorgestellten Grundlagen. Nach dieser Auswahl wurden die Berechtigungsmatrizen konzipiert und gemäß der User Stories für verschiedene Benutzergruppen ausgefüllt. Sie dienten als Entwurf.

Der Konzeptionsprozess gliedert sich in fünfzehn detaillierte Schritte:

**Schritt 1:** Analyse der IT-Landschaft, um ein klares Bild der vorhandenen Infrastruktur zu erhalten.

**Schritt 2:** Festlegung der Anforderungen und Ziele, die das Berechtigungskonzept erfüllen soll.

**Schritt 3:** Identifikation der relevanten Services bzw. Module, die in das Berechtigungssystem integriert werden sollen.

**Schritt 4:** Eindeutige Identifikation der Ressourcen bzw. Objekte, die durch das System geschützt werden sollen.

**Schritt 5:** Definition der Berechtigungsoperationen, die auf die Ressourcen angewendet werden können.

**Schritt 6:** Konsolidierung bzw. Zusammenfassung dieser Berechtigungsoperationen, um Übersichtlichkeit und Verwaltbarkeit zu verbessern.

**Schritt 7:** Erstellung von Personas, die typische Nutzerprofile darstellen und die Entwicklung von benutzerzentrierten Berechtigungen ermöglichen.

**Schritt 8:** Ableitung der User Roles aus den Personas, um sicherzustellen, dass jede Rolle spezifisch auf User-Anforderungen eingeht.

**Schritt 9:** Zusammenfassung dieser Rollen in Benutzergruppen zur verbesserten Übersichtlichkeit.

**Schritt 10:** Erstellung der User Stories, die spezifische Anwendungsfälle und Zugriffsrechte der Nutzer beschreiben.

**Schritt 11:** Analyse des Prozesses eines Standardnutzers, um die Nutzerinteraktion mit dem System zu verstehen.

**Schritt 12:** Entwicklung eines Prozesses für die Validierung und Zuordnung der Rollen an die jeweiligen Nutzer.

**Schritt 13:** Festlegung der spezifischen Zugriffsregeln, die im Berechtigungsmodell angewendet werden.

**Schritt 14:** Konzeption der Berechtigungsmatrix, die alle Rollen und ihre jeweiligen Berechtigungen visualisiert.

**Schritt 15:** Entwurf der finalen Berechtigungsmatrix als handlungsleitendes Dokument.

Für die Implementierung des erarbeiteten Konzepts werden folgende fünf weitere Schritte vorgesehen:

**Schritt 16:** Praktische Umsetzung und Einführung des Berechtigungssystems.

**Schritt 17:** Schaffung von Kontrollmöglichkeiten, um die Einhaltung der Berechtigungen zu überprüfen.

**Schritt 18:** Durchführung von Tests, um die Funktionalität und Effizienz des Systems zu validieren.

**Schritt 19:** Abschluss der vollständigen Inbetriebnahme des Berechtigungssystems.

**Schritt 20:** Etablierung eines Prozesses für kontinuierliche Verbesserungen, basierend auf Nutzerrückmeldungen und systematischen Überprüfungen.

Der hier vorgestellte Ansatz ist sorgfältig strukturiert und basiert auf Methoden des Requirements Engineering. Dadurch wird sichergestellt, dass keine wesentlichen Anforderungen übersehen werden und das System flexibel auf Veränderungen reagieren kann. Dies gewährleistet ein effektives und sicherheitsorientiertes Ergebnis.

## 3.3 Objekte/Ressourcen

Im nachfolgenden Unterkapitel erfolgt eine Auseinandersetzung mit den auf der DaFne-Plattform vorhandenen Objekten und Ressourcen.

### 3.3.1 Definition

In Anlehnung an Tsolkas (vgl. [49]: S. 2) wird folgende Definition verwendet.

Objekte werden oft als Ressourcen bezeichnet, da sie verschiedene Funktionen und Inhalte bereitstellen können, die genutzt werden können. Ressourcen können in Funktionsressourcen und Inhaltsressourcen unterteilt werden, je nachdem, welche Dienste sie anbieten. Ein Beispiel für eine Funktionsressource wäre eine Konvertierungssoftware, die die Funktion *Datei konvertieren* anbietet, z.B. von Microsoft Word nach PDF. Eine Inhaltsressource wäre dagegen eine Datenbank, die Dateninhalte bereitstellt. Es ist auch möglich, dass eine Ressource sowohl eine Funktions- als auch eine Inhaltsressource ist. Beispielsweise kann ein Smartphone die Funktion *SMS versenden* bereitstellen und gleichzeitig den Zugriff auf Fotos oder Internetseiten ermöglichen.

In der Vergangenheit lag der Schwerpunkt häufig auf der Interaktion zwischen Subjekten. Ressourcen stellen Inhalte und Funktionen zur Verfügung und werden in Funktions- und Inhaltsressourcen unterteilt:

**Funktionsressourcen**:

- Repräsentieren funktionale Aktivitäten, wie z.B. ein Administrationstool oder ein Messenger-Dienst.

**Inhaltsressourcen**:

- Ermöglichen den Zugriff auf Inhalte, z.B. Dokumentenverzeichnisse oder Kundendaten.

**Funktions- und Inhaltsressourcen**:

- Ermöglichen die Modifikation von Inhalten, z.B. die Benutzeroberfläche einer Kundendatenbank oder eines Mailprogramms.

Weiterhin wird die Umgebung, in der diese Ressourcen existieren oder genutzt werden, unterschieden in:

**Physische Umgebung**:

- Hier sind die Ressourcen reale Gegenstände, Räume oder andere Objekte in der physischen Welt.

**Nicht-reale oder virtuelle Umgebung**:

- Hier stehen die Ressourcen im Kontext der IT-Welt, wie z.B. Desktop-Computer, Clients, Server, Betriebssysteme, Programme oder Dienste.

### 3.3.2 Identifikation der Funktionsressourcen: Module

Im Rahmen des Projekts DaFne wird eine Hauptkategorie von Anwendungsfällen (Use Cases) betrachtet: Solche, die spezifische generative Algorithmen auf der Plattform nutzen, um synthetische Daten im Kontext von Smart Cities zu generieren. Diese Anwendungsfälle und ihre zugehörigen Module bzw. deren Dienste/Services sind in Abbildung 3.2 dargestellt.

Die auf der Plattform bereitgestellten Use Cases zielen darauf ab, spezifische Daten für konkrete Anforderungen in verschiedenen Bereichen, wie z.B. Smart City, zu generieren und sind folgende:

**1. Tabular Data Synthesis:** Dies ist der Hauptanwendungsfall, der den Prozess der Generierung synthetischer Daten beschreibt, die echten tabellarischen Daten ähnlich sind. Benutzer können ihre eigenen Daten hochladen und diese mit Hilfe fortgeschrittener generativer Modelle synthetisieren. Ziel ist es, sowohl den Schutz der Privatsphäre als auch die Nutzung der Daten zu Analysezwecken unter Einhaltung der Datenschutzbestimmungen zu ermöglichen (vgl. [32]: S. 2).

Enthält folgende Module bzw. Services:

*(1.1) Rule-based data generation:* Die Plattform ermöglicht es den Nutzern, eigene Datenspalten nach individuellen Vorgaben zu erstellen. Dabei können sie mithilfe regelbasierter Generierung den Datentyp, die Verteilung und spezifische Werte festlegen. Zusätzlich können Spalten generiert werden, die auf bestimmten Regeln basieren und von anderen Spalten abhängen. Auf diese Weise können tabellarische Daten erstellt werden, ohne auf vorher existierende Daten angewiesen zu sein. Benutzer können diese Methode nutzen, um bestehende Datensätze spaltenweise zu erweitern und zu ergänzen (vgl. [33]: S. 3-5).

*(1.2) Data fusion:* Datenfusion bezeichnet das Zusammenführen verschiedener Datensätze. Benutzer können ihre eigenen Daten mit öffentlich zugänglichen Informationen, wie z.B. Wetter- oder Verkehrsflussdaten, kombinieren, um den Informationsgehalt ihrer ursprünglichen Daten zu erhöhen (vgl. [33]: S. 3-5).

*(1.3) Reproduction / Data generation based on existing data:* Durch Vervielfältigung oder Generierung von Daten auf Basis vorhandener Informationen, können neue Zeilen zu einem bestehenden tabellarischen Datensatz hinzugefügt werden. Diese neuen Zeilen ähneln den vorhandenen Daten in ihren Merkmalen und Beziehungen. Dieser Service umfasst auch die Evaluation der Daten anhand von Evaluierungsmetriken (vgl. [33]: S. 3-5).

**2. Data Evaluation:** Der zweite Anwendungsfall stellt eine Erweiterung des ersten dar und bietet verschiedene Analysemethoden auf der Plattform an, mit denen die Benutzer die Ähnlichkeit zwischen den synthetisierten Daten und den Originaldaten bewerten können. Unter *Data Evaluation* versteht man den Bewertungs- oder Analyseprozess von Daten, die mit Hilfe des Tabellarischen Datensyntheseverfahrens erzeugt wurden. Ziel ist es, festzustellen, inwieweit die generierten Daten die Eigenschaften und Muster der Originaldaten widerspiegeln, ohne dabei vertrauliche oder persönliche Informationen preiszugeben. In diesem Rahmen werden verschiedene Bewertungskriterien und Methoden verwendet, um die Qualität und Anwendbarkeit der generierten Daten für bestimmte Anwendungsbereiche zu überprüfen (vgl. [30]: S. 1-13; [10]).

Enthält folgende Module bzw. Services:

*(2.1) Generation Quality Report:* Dieser Service kann als *Generierungsqualitätsbericht* im Kontext der Tabellarischen Datensynthese verstanden werden, der eine umfassende Analyse der Qualität und Güte der generierten Daten liefert. Dieser Service gibt Aufschluss darüber, wie ähnlich die generierten Daten zu den Originaldaten sind, wobei verschiedene Aspekte wie statistische Übereinstimmung, Datenschutzmaßnahmen, Datenvielfalt und ihre Eignung für geplante Anwendungen wie maschinelles Lernen und Datenanalyse berücksichtigt werden (vgl. [33]: S. 5,9; [10]).

*(2.2) Data Report:* Dieser Dienst bietet einen ausführlichen Bericht mit detaillierten Informationen und Analysen zu einem oder mehreren Datensätzen. Ziel des Dienstes ist es, den Nutzern einen detaillierten Einblick in die Daten zu geben, indem verschiedene Elemente der Datensätze untersucht werden, einschließlich ihrer Struktur, Qualität, erkennbarer Muster und möglicher Einschränkungen oder Herausforderungen, die sich aus den Daten ergeben können (vgl. [33]: S. 5,9 ; [10]).

**3. Neighbourhood Generation:** Der dritte Anwendungsfall „Neighbourhood Generation“ beschäftigt sich mit der Simulation und Planung von Stadtstrukturen mit Hilfe

von Graphen. Dabei repräsentieren die Knoten des Graphen verschiedene städtische Einrichtungen wie Schulen, Cafés oder Geschäfte. Dieser Ansatz ermöglicht es, Planungsaufgaben im urbanen Kontext zu unterstützen, wie z.B. die optimale Platzierung neuer Einrichtungen oder die Übertragung erfolgreicher Infrastrukturkonzepte von einer Stadt auf eine andere. Die in diesem Zusammenhang generierten Daten sind somit Graphen oder virtuelle Stadtpläne, die beispielsweise dabei helfen können, den idealen Standort für ein neues Café vorherzusagen oder Radwegenetze zwischen Städten anzupassen. So könnte ein erfolgreiches Radwegesystem in Amsterdam als Vorbild für die Planung in Hamburg dienen. Die Parametrisierung dieser Modelle erfolgt durch die Auswahl spezifischer Städte und der zu betrachtenden Entitäten (vgl. [30]: S. 5).

Enthält folgende Module bzw. Services:

*(3.1) Neighbourhood Generator:* Dieser erhält eine grafische Struktur, die einzelne Knoten mit spezifischen Eigenschaften und Funktionen kennzeichnet. Der Benutzer kann durch einen Filter, der als Maske dargestellt wird, bestimmte Knoten für die Analyse auswählen. Der Dienst berechnet dann eine Matrix von Vorhersagen für die Funktion oder Nutzung jedes ausgewählten Knotens im Graphen. Jedem Knoten wird eine Punktzahl zwischen 0 und 1 für jede Kategorie zugewiesen, die den Zweck oder die Funktion definiert, die jedem Knoten oder Gebäude im generierten städtischen Netzwerk zugewiesen wird (vgl. [30]: S. 4).

**4. Reinforcement Learning - Agent based Mobility Data Generation:** Dieser Anwendungsfall basiert auf einem Reinforcement-Learning-Verfahren, das individuell trainierte Agenten simuliert, die sich je nach Rolle oder Ziel unterschiedlich durch urbane Räume bewegen, wie z.B. Touristen, die Sehenswürdigkeiten besichtigen, Rentner, die Cafés besuchen oder Studenten auf dem Weg zur Universität. Die dabei generierten Daten stellen wertvolle Mobilitätsinformationen dar. Eine Anpassung dieser Simulationen ist durch verschiedene Parameter möglich, z.B. durch die Festlegung der Anzahl der Agenten oder durch die Spezifikation ihrer Eigenschaften (vgl. [21]: S. 2-8).

Enthält folgende Module bzw. Services:

*(4.1) Mobilitätsdaten-Generator:* Dieses Modul simuliert das Verhalten von Agenten in einer definierten Umgebung, die auf ihre Aktionen reagiert und diese auswertet. Die Interaktionen der Agenten mit ihrer Umgebung führen zu spezifischen Ergebnissen und Rückmeldungen, die zur Verbesserung der Entscheidungsprozesse der Agenten genutzt werden können. Die erzielten Ergebnisse und das daraus resultierende Feedback werden

in einer CSV-Datei zur weiteren Analyse gespeichert. Die Rahmenbedingungen für die Simulationsumgebung werden ebenfalls in einer CSV-Datei definiert. Der Dienst berechnet dann eine Matrix von Vorhersagen für die Funktion oder Verwendung jedes ausgewählten Knotens im Graphen. Jedem Knoten wird eine Punktzahl zwischen 0 und 1 für jede Kategorie zugewiesen, die den Zweck oder die Funktion definiert, die jedem Knoten oder Gebäude im generierten städtischen Netzwerk zugewiesen wird (vgl. [21]: S. 4-6).

**5. Data Analyzer:** Dieser Anwendungsfall konzentriert sich explizit auf die Analyse eines einzelnen Datensatzes. Im Gegensatz zu anderen Anwendungen, die möglicherweise Vergleiche zwischen mehreren Datensätzen, z.B. zwischen Originaldaten und synthetisierten Daten, anstellen, konzentriert sich dieser Anwendungsfall ausschließlich auf die Visualisierung und das Verständnis eines spezifischen Datensatzes. Diese Funktionalität bietet Nutzern die Möglichkeit, einen tieferen Einblick in ihre Daten zu erhalten, ohne den Kontext der Datengenerierung zu kennen, indem sie ihren Datensatz auf die Plattform hochladen und analysieren lassen (vgl. [18]).

Enthält folgende Module bzw. Services:

*(5.1) Analyzer:* Der primäre Zweck des Analyzers ist es, die Untersuchung eines einzelnen Datensatzes zu erleichtern, indem seine charakteristischen Daten, wie z.B. die Verteilungen innerhalb verschiedener Spalten, grafisch aufbereitet und in einem PDF-Bericht dargestellt werden (vgl. [18]).

**6. Data Idea:** Neben der Bereitstellung von Erkenntnissen durch synthetische Daten soll diese Funktion auch dazu dienen, neue Datenideen zu inspirieren. Zum einen sollen vorgefertigte Use Cases mit entsprechenden Datensätzen bereitgestellt werden. Zum anderen sollen generische Ansätze helfen, Datensätze von Kooperationspartnern durch Replikation nutzbar zu machen (vgl. [21]: S. 1,4 ; [35]: S. 6,11).

Enthält folgende Module bzw. Services:

*(6.1) Use Case Explorer:* Der Use Case Explorer bietet dem Nutzer eine benutzerfreundliche Oberfläche zur Abfrage von vorab trainierten Machine-Learning-Modellen, um Daten für einen spezifischen Anwendungsfall zu generieren (vgl. [21]: S. 1,4; [35]: S. 6,11).

**7. Contribution:** Dieser Anwendungsfall macht die gesamte Plattform flexibler und erweiterbarer, indem neue Modelle, Metriken und Datenquellen hinzugefügt werden können, siehe Abbildung 2.2 (vgl. [45] ; [46]: S. 3,6 ; [21]: S. 4,8 ; [35]: S. 6,11).

Enthält folgende Module bzw. Services:

*(7.1) Add Generative Model:* Dies beinhaltet das Hinzufügen eines generativen Modells.

*(7.2) Add Evaluation Metric:* Dies beinhaltet das Hinzufügen von Evaluationsmetriken.

*(7.3) Add Data Source:* Dies bezieht sich auf die Integration von Datenquellen. Weitere Informationen hierzu finden sich in Kapitel 3.3.3, wo auch die verschiedenen Arten von Datenbanken und Datenquellen behandelt werden.

| Art des Use Cases | Module (Services) |
|---|---|
| Generate Data | Rule-Based |
| Generate Data | Fusion |
| Generate Data | Reproduction (inkl. Evaluation der Daten anhand von Evaluierungsmetriken) |
| Evaluate Generated/Own Data | Generation Quality Report |
| Evaluate Generated/Own Data | Data Report |
| Neighbourhood Generation | Neighbourhood Generator |
| Mobilitätsdatengenerierung mit RL | Mobilitätsdaten Generator |
| Data Analyzer | Analyzer (EIN DATENSATZ) |
| Data Idea | Use Case Explorer |
| Contribution | Add Generative Model |
| Contribution | Add Evaluation Metric |
| Contribution | Add Data Source |

Abbildung 3.2: Übersicht der Use Case Art mit den entsprechenden Services bzw. Modulen

### 3.3.3 Identifikation der Inhaltsressourcen: Daten

Die in (vgl. [46] ; [33]) vorgestellte Architektur sowie die Absprache mit dem DaFne-Team werden als Grundlage für die verschiedenen Arten von Daten bzw. Datenbanken sowie Datenquellen, welche die DaFne-Plattform enthält, verwendet. Eine Übersicht ist in Abbildung 3.3 dargestellt.

**Private Data:** Dies sind privat hochgeladene Datensätze von Nutzern oder von der Plattform selbst, die nur für den jeweiligen Nutzer, dem sie gehören, sichtbar sind. Diese Daten können auf die Plattform hochgeladen werden und sind nur für den Eigentümer zugänglich.

**Public Data:** Dies bezieht sich auf öffentliche Datensätze, die Open Source sind und für Demo- und Testzwecke zur Verfügung gestellt werden. Jeder kann auf diese Daten zugreifen und sie nutzen. Sie können auch parametrisiert werden, da sie aggregierte Daten aus bestimmten problembezogenen Datensätzen enthalten.

**Public Data Interface (Datenbankanbindung):** Ermöglicht die Anbindung weiterer Datenquellen oder Datenbanken an die Plattform. Diese Funktion ist ausschließlich für

Entwickler gedacht, die eigene Modelle integrieren möchten und ist nur für den Nutzer sichtbar, der die Anbindung initiiert hat.

**Own Generated Data:** Dies bezieht sich auf die innerhalb der DaFne-Plattform generierten synthetischen Daten. Diese Daten sind ebenfalls für alle Nutzer sichtbar, da die synthetische Datengenerierung die Probleme des Datenschutzes und der Privatsphäre löst.

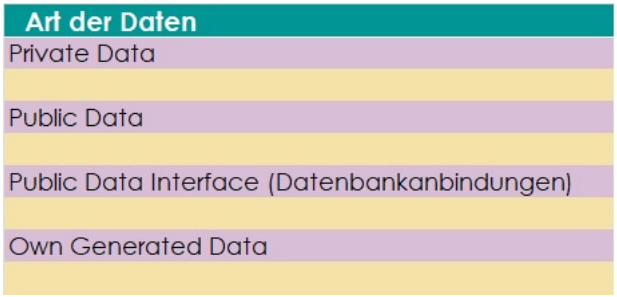



Abbildung 3.3: Übersicht der Art der Daten

### 3.3.4 Identifikation der Funktions- und Inhaltsressourcen

Der nachfolgende Abschnitt basiert auf den Grundlagen, die in Tsolkas (vgl. [12]: S. 2-7) dargelegt sind. Die Verschmelzung von funktionalen und inhaltlichen Ressourcen innerhalb von IT-Systemen, Software oder Web-Anwendungen führt zur Bildung integrierter Ressourcen. Diese sind sowohl für die Überwachung und Steuerung von Funktionalitäten - wie z.B. die Ausführung bestimmter Aktionen - als auch für den Zugriff und die Kontrolle von Inhalten oder Daten verantwortlich. Solche kombinierten Funktions- und Inhaltsressourcen sind für die Umsetzung eines effektiven Berechtigungskonzepts, das die Sicherheit der Plattform gewährleistet und die Vergabe und Verwaltung von Berechtigungen und Rollen optimiert, unerlässlich. Diese Komponenten sind grundlegend für ein umfassendes Berechtigungs- und Identitätsmanagement, das die adäquate Vergabe von Zugriffsrechten unterstützt, die Verwaltung von Benutzeridentitäten vereinfacht und zur Wahrung der Datenintegrität und -sicherheit beiträgt. Durch die Integration dieser Ressourcentypen wird eine solide Grundlage für Berechtigungskonzepte in digitalen Plattformen geschaffen, die Sicherheit, Effizienz und Benutzerfreundlichkeit fördern. Dadurch wird sichergestellt, dass die richtigen Personen angemessenen Zugang zu den benötigten Funktionen und Inhalten erhalten. Daher sind solche integrierten Ansätze im Bereich der funktionalen und inhaltlichen Ressourcen, wie z.B. User Management, Role Management,

Permission Management, Policies and Rule Management, Support Ticket Management und Personal Data Management, für moderne Softwarestrukturen und -systeme, insbesondere unter dem Aspekt der Zugriffskontrolle und Sicherheit, unverzichtbar für die Verwaltung und Steuerung.

**User Management:** Dies beinhaltet die Verwaltung der Benutzer auf der Plattform. Es umfasst Aspekte wie Registrierung, Erstellung, Anpassung / Änderung und Löschung.

**Role Management:** Hier geht es um die Verwaltung der Rollen auf der Plattform. Es ermöglicht die Zuweisung spezifischer Rollen an Benutzer auf der Grundlage ihrer Aufgaben und Verantwortlichkeiten. Es umfasst auch das Erstellen, Anpassen/Ändern und Löschen von Rollen.

**Permission Management:** Dies bezieht sich auf die Verwaltung von Berechtigungen. Dies beinhaltet die Kontrolle darüber, welche Aktionen bestimmte Benutzer oder Rollen ausführen dürfen, den Zugriff auf bestimmte Funktionen oder Daten sowie das Erstellen, Anpassen/Ändern und Löschen von Berechtigungen.

**Policies and Rule Management:** Dies beinhaltet die Verwaltung von Richtlinien und Regeln innerhalb der Plattform. Dies umfasst die Definition von Verhaltensregeln, Sicherheitsrichtlinien, Compliance-Standards und allgemeinen Richtlinien für die Nutzung der Plattform sowie das Erstellen, Anpassen/Ändern und Löschen von Richtlinien und Regeln.

**Support Ticket Management:** Dies bezieht sich auf die Verwaltung von Support-Tickets innerhalb der Plattform. Dies umfasst die Erfassung, Verfolgung, Zuordnung und Bearbeitung von Kundenanfragen oder Problemmeldungen, um einen effizienten und zeitnahen Support zu gewährleisten.

**Personal Data Management:** Dies bezieht sich auf die Verwaltung der persönlichen Daten der jeweiligen Nutzer auf der Plattform, wie z.B. das Bearbeiten, Löschen und Ändern von persönlichen Daten unter Einhaltung des Datenschutzes.

**Contribution - Extend Platform Management (Generative Models):** Dies umfasst die Verwaltung der neu hinzugefügten generativen Modelle durch die Nutzer auf der Plattform, einschließlich Funktionen wie Bearbeiten, Löschen und Ändern dieser hinzugefügten Modelle.

**Contribution - Extend Platform Management (Evaluation Metrics):** Dies beinhaltet die Verwaltung der neu hinzugefügten Bewertungsmetriken durch die Nutzer auf

der Plattform, wie z.B. das Bearbeiten, Löschen und Ändern dieser hinzugefügten Metriken.

**Contribution - Extend Platform Management (Data Sources):** Dies betrifft die Verwaltung der neu integrierten Datenquellen durch die Nutzer auf der Plattform, einschließlich Funktionen wie Bearbeiten, Löschen und Ändern dieser integrierten Datenquellen.

**Generate/Evaluate/Analyze Data Management:** Dies betrifft die Verwaltung der von den Nutzern auf der Plattform generierten, evaluierten und analysierten Daten für die bereits angebotenen Anwendungsfälle und auch für neue spezifische Anwendungsfälle mit dem Use Case Explorer, d.h. die drei Generierungsmodelle, einschließlich der Evaluierung und des Analyzers sowie des Use Case Explorers, und die anderen auf der Plattform angebotenen Anwendungsfälle, die den Kern der Plattform bilden, einschließlich der Funktionen zum Bearbeiten, Löschen und Ändern dieser generierten, evaluierten und analysierten Daten.

Die Abbildung 3.4 gibt einen Überblick über die Funktions- und Inhaltsressourcen

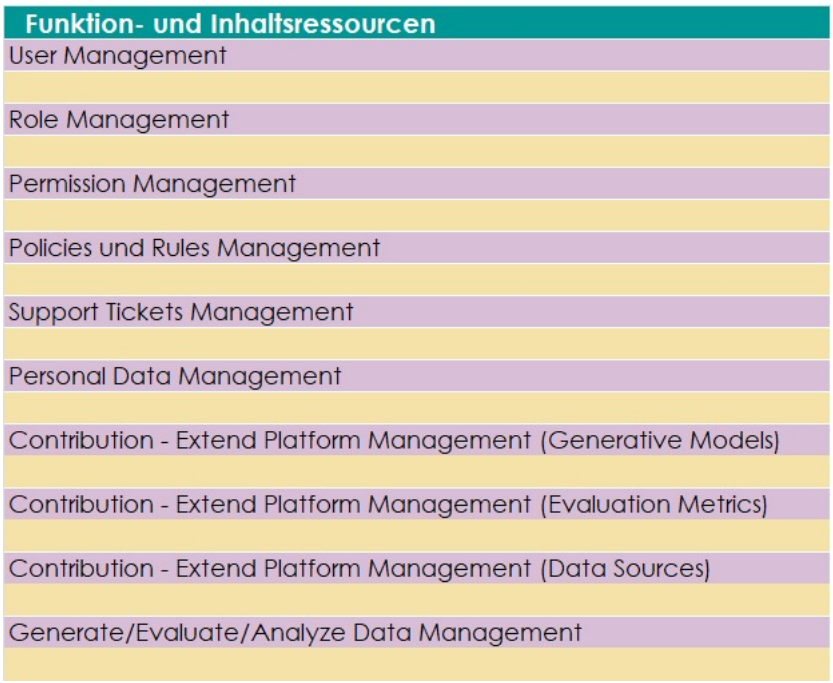



Abbildung 3.4: Übersicht der Funktions- und Inhaltsressourcen

## 3.4 Rollen

Im nachfolgenden Unterkapitel erfolgt eine Auseinandersetzung mit den auf der DaFne-Plattform vorhandenen Rollen.

### 3.4.1 Erklärung

Der nachfolgende Abschnitt stützt sich auf (vgl. [49]: S. 12-14).

Eine wesentliche Komponente für die Verwaltung von Berechtigungen in IT-Systemen ist die Rolle. Das Konzept der rollenbasierten Zugriffskontrolle (Role Based Access Control, RBAC) verdeutlicht die Bedeutung von Rollen für Berechtigungen: Sie ermöglichen es, Berechtigungen von Identitäten zu entkoppeln.

Im Kontext von Berechtigungen beschreibt eine **Geschäftsrolle** die Funktion, die der Rolleninhaber im Unternehmen ausübt. Diese Funktion kann entweder statisch auf einen bestimmten Arbeitsbereich oder dynamisch auf ein bestimmtes Aufgabenspektrum ausgerichtet sein. Für die Benennung von Rollen gibt es vier Varianten:

*Orientierung am Funktionstyp:*

Die Rollenbezeichnung orientiert sich an der Art der Funktion und nicht an dem Bereich im Unternehmen, in dem der Rolleninhaber tätig ist. Beispielsweise könnte eine Rolle mit der Bezeichnung 'Entscheider' unabhängig davon sein, ob die Entscheidungen den Einkauf von Material oder die Vergabe von Dienstleistungen betreffen.

*Orientierung am Verantwortungsbereich:*

Hier wird die Rolle nach dem Verantwortungsbereich des Rolleninhabers benannt, entweder entsprechend der Aufbauorganisation oder daraus abgeleitet. Beispiele sind Rollen wie 'Einkauf' oder 'Facility Management'.

*Orientierung an der Bezeichnung des Rolleninhabers:*

Eine Rolle kann auch nach der Bezeichnung des Funktionsträgers benannt werden, um sie verständlicher zu machen. Beispielsweise wäre die Rolle 'Kassierer' in einer Bank eine solche Bezeichnung.

*Orientierung an einem Geschäftsprozess oder einem Tätigkeitsablauf:*

Stehen Prozesse im Vordergrund, können Rollen auch nach Prozessbezeichnungen benannt werden, z. B. *Datenschutzaudit* oder *Produktionsüberwachung*.

**Abstrakte Rollen** beschreiben in der Regel eine organisatorische Funktion, sind aber für die Berechtigungssteuerung ungeeignet, da sie nicht klar spezifizieren, welche konkreten Tätigkeiten enthalten sind. Für eine wirklich detaillierte Berechtigungssteuerung werden konkrete, operative Rollen benötigt. Diese können durch Unterteilung der abstrakten Rollen zu einem geschlossenen Rollenkonzept konkretisiert werden.

**Primäre Rollen** haben einen direkten Einfluss auf die Geschäftstätigkeit und berühren die Wertschöpfungskette, während sekundäre oder unterstützende Rollen parallel zu den primären Rollen existieren oder diese unterstützen. Das Rollenkonzept legt fest, welche Geschäftsrollen im Unternehmen existieren und wie den Rollen Aktivitäten und Rechte zugeordnet sind.

**Business-Rollen** in der Berechtigungssteuerung dienen der Zuordnung von Berechtigungen zu einer Identität anhand von Tätigkeitsschwerpunkten und sind ein Schlüsselelement für die Gestaltung von Berechtigungsbündeln.

**Technische Rollen** beziehen sich auf zu berechtigende Objekte und nicht auf Identitäten oder Geschäftsaktivitäten. Sie werden in IT-Anwendungen verwendet, um Funktionen zu beschreiben und zu berechtigen. Ein Beispiel hierfür ist die technische Rolle *Systemadministrator* in UNIX-Systemen, die sich ausschließlich auf das System und nicht auf das gesamte Unternehmen bezieht.

Technische Rollen in der Berechtigungssteuerung sind nicht an Identitäten, sondern an Ressourcen gebunden. Sie berücksichtigen ausschließlich die IT-Anwendung und die Funktionen innerhalb dieser Anwendung. Diese Rollen werden durch technische Benutzerkonten in den Systemen abgebildet, um die entsprechenden Berechtigungen zu steuern.

Der Unterschied zwischen Business-Rollen und technischen Rollen besteht darin, dass Business-Rollen eng mit Identitäten und Geschäftsaktivitäten verknüpft sind, während technische Rollen davon losgelöst sind und sich auf Systemressourcen konzentrieren.

### 3.4.2 Persona

Personas sind in Projekten wie DaFne äußerst nützlich, denn sie ermöglichen ein tiefes Verständnis für die verschiedenen Nutzergruppen und ihre spezifischen Bedürfnisse. Im Folgenden werden einige Argumente für die Nützlichkeit von Personas dargelegt.

**Benutzerzentrierte Entwicklung:** Personas helfen dabei, die Anforderungen und Erwartungen der verschiedenen Benutzergruppen zu identifizieren, was zu einer benutzerzentrierten Entwicklung der Plattform führt. Dadurch wird sichergestellt, dass die Plattform für alle relevanten Nutzer funktional, zugänglich und effizient ist.

**Zielgerichtete Funktionalitäten:** Durch die Definition klarer Personas kann das Projektteam gezielte Funktionalitäten entwickeln, die den spezifischen Aufgaben und Verantwortlichkeiten der verschiedenen Nutzergruppen entsprechen. Dies erhöht die Effektivität und Zufriedenheit der Nutzer.

**Effiziente Ressourcenallokation:** Die Kenntnis über die Bedürfnisse und Aufgaben der verschiedenen Personas ermöglicht eine effizientere Zuteilung von Entwicklungsressourcen. Funktionen und Features können priorisiert und optimiert werden, um den größten Mehrwert für die wichtigsten Nutzergruppen zu bieten.

Die Persona Developer Models spielt eine zentrale Rolle im Kontext des Projekts DaFne, da sie direkt an der Generierung der Modelle beteiligt ist, die für die synthetische Datenerzeugung verwendet werden. Diese Persona ist dafür verantwortlich, neue generative Modelle zu entwickeln und zur Plattform hinzuzufügen. Ihre Arbeit ist besonders wichtig, aus den folgenden Gründen:

**Innovation in der Datengenerierung:** Developer Models entwickeln maßgeschneiderte Modelle, die auf spezifische Anforderungen und Anwendungsfälle zugeschnitten sind. Ihre Kreativität und technische Kompetenz treiben die Innovation voran und tragen dazu bei, dass die Plattform stets aktuelle und fortschrittliche Methoden nutzt.

**Optimierung der Datenqualität:** Synthetische Daten müssen von hoher Qualität sein, um in KI-Anwendungen eingesetzt werden zu können. Durch die Entwicklung und Implementierung von robusten generativen Modellen sorgen Developer Models dafür, dass die Daten den erforderlichen Qualitätsstandards entsprechen und verlässliche Ergebnisse liefern.

**Flexibilität und Anpassungsfähigkeit:** Die von Developer Models entwickelten Modelle sind flexibel und können für verschiedene Anwendungsbereiche angepasst werden. Dies fördert eine breitgefächerte Nutzung der Plattform und erhöht deren Attraktivität für eine Vielzahl von Nutzergruppen.

**PERSONA 01: DEVELOPER MODELS**

***Name:*** Marlene Schuster

***Lebensmotto/Leitsatz:*** 'Innovation beginnt mit der Frage: Was wäre wenn?

***Alter:*** 28

***Wohnort:*** Dresden, Deutschland

***Ausbildung:*** Master in Informatik mit Schwerpunkt Künstliche Intelligenz

***Beruf:*** Entwicklerin generativer KI/ML-Modelle

***Branche:*** Softwareentwicklung und KI-Forschung

***Spezialgebiet:*** Künstliche Intelligenz, Maschinelles Lernen, Generative Modelle

***Technische Fähigkeiten:***

- Expertise in der Entwicklung und dem Training von generativen Modellen wie GANs (Generative Adversarial Networks) und VAEs (Variational Autoencoders)

- Fundierte Programmierkenntnisse in Python und Frameworks wie TensorFlow und PyTorch

- Erfahrung in der Anwendung und Optimierung von Deep Learning Architekturen

- Kenntnisse in der Verarbeitung und Analyse großer Datensätze

- Erfahrung in der Implementierung von Algorithmen zur Automatisierung der Datenvorverarbeitung und Modellverfeinerung

- Fähigkeit zur Arbeit mit Cloud Computing Plattformen für skalierbares Modelltraining

***Motivation und Ziele:***

Marlene ist leidenschaftlich daran interessiert, die Grenzen der KI durch die Entwicklung neuer generativer Modelle zu erweitern. Ihr Hauptziel ist es, Modelle zu erstellen, die eine

beispiellose Genauigkeit und Vielseitigkeit bei der Generierung von Inhalten, von Bildern bis hin zu Texten, bieten. Ziel ist es, diese Technologien für kreative Problemlösungen in Bereichen wie Design, Unterhaltung und darüber hinaus nutzbar zu machen.

***Schwächen:***

- Manchmal zu sehr auf Forschung und Entwicklung fokussiert, was dazu führen kann, dass praktische Anwendungsaspekte vernachlässigt werden.

- Neigt dazu, sich in der Komplexität der Modelle zu verlieren und den Überblick über das Gesamtprojekt zu verlieren.

- Unter Druck kann es schwierig sein, Prioritäten zwischen mehreren Projekten zu setzen.

***Stärken:***

- Starke analytische Fähigkeiten und die Fähigkeit, komplexe Probleme methodisch zu lösen

- Kreativität bei der Entwicklung neuer Ansätze und Lösungen

- Ausgeprägte Teamfähigkeit und Erfahrung mit interdisziplinären Projekten

- Schnelle Auffassungsgabe und die Fähigkeit, sich rasch in neue Technologien einzuarbeiten

- Starkes Engagement für Ethik in der KI und die gesellschaftlichen Auswirkungen der eigenen Arbeit

Aus dem Paper (vgl. [33]: S. 6) wurden die bestehenden Rollen adaptiert. Anschließend wurden die Rollen verfeinert und einige neue Rollen hinzugefügt. Zu jeder Rolle gibt es nun Personas, um diese im Detail darzustellen.

Im Anhang A.1 sind alle weiteren Personas aufgeführt, die für den erfolgreichen Betrieb und die Verwaltung der Plattform von essenzieller Bedeutung sind.

### 3.4.3 User Roles - Identifikation der Nutzerrollen und Benutzergruppen

In Anlehnung an das Paper (vgl. [33]: S. 6; [31]) werden folgende Definitionen für die einzelnen Rollen verwendet und anschließend den Benutzergruppen zugeordnet.

**Super-Admin:** Der Super-Admin ist die oberste administrative Rolle mit umfassenden Zugriffsrechten auf alle Dienste der Plattform. Verantwortlich für die Verwaltung und Wartung des Systems, Benutzerverwaltung und die Sicherstellung der Sicherheitsstandards, trägt der Super-Admin die Hauptverantwortung für den reibungslosen Betrieb und die Sicherheit der gesamten Plattform.

**Plattform-Admin:** Der Plattform-Admin ist für die alltägliche Verwaltung und Unterstützung der Benutzer zuständig, hat jedoch eingeschränktere Zugriffsrechte im Vergleich zum Super-Admin. Diese Rolle umfasst die Überwachung der Systemleistung, Durchführung von Updates und direkte Unterstützung der Nutzer, darf jedoch keine grundlegenden Systemeinstellungen ändern.

**Support-User:** Nutzer, die einen eingeschränkten Zugang zu Benutzerkonten und Systemfunktionen haben, um Support-Anfragen zu bearbeiten.

**Auditor:** Benutzer, der überprüft, ob die Prozesse, Anforderungen, Rechte und Richtlinien die geforderten Standards und Normen erfüllen.

**Developer Models:** Benutzer, die berechtigt sind, zur Plattform beizutragen, indem sie neue generative Modelle hinzufügen.

**Developer Metrics:** Benutzer, die berechtigt sind, zur Plattform beizutragen, indem sie neue Evaluierungsmetriken bzw. Modelle hinzufügen.

**Data Engineer:** Benutzer, die berechtigt sind, neue Datenquellen zu integrieren oder zu verbinden.

**Data Scientist:** Nutzer, die Daten für KI-Anwendungsfälle oder Daten im Kontext von Smart Cities benötigen, haben aber keine Kenntnisse über ML-Algorithmen und deren Parameter oder über Datensynthesemethoden.

Diese Nutzerrollen sind verschiedenen Benutzergruppen zugeordnet, wie in der folgenden Abbildung 3.5 dargestellt.

| Benutzergruppen | Nutzerrollen |
|---|---|
| Adminstratoren | |
| | Super-Admin<br>Plattform-Admin |
| Support | |
| | Support User |
| Auditoren | |
| | Auditor |
| Contributor | |
| | Developer Models<br>Developer Metrics<br>Data Engineer |
| Anwender | |
| | Data Scientist |

Abbildung 3.5: Übersicht über Benutzergruppen und Nutzerrollen

### 3.4.4 User Stories - Benutzergruppen

Die folgende Sammlung von User Stories gibt detailliert Auskunft darüber, wie die verschiedenen Benutzergruppen der DaFne-Plattform auf die unterschiedlichen Arten von Datenquellen und Services zugreifen können. Diese Stories sind von entscheidender Bedeutung für die Konzeption des Berechtigungsmodells, da sie einen klaren Überblick darüber geben, welche Benutzergruppen welche Services zur Verfügung stehen und welche Operationen diese auf die verschiedenen Arten von Datenquellen durchführen können.

**Adminstratoren:**

AD1. Als Adminstratoren können wir auf alle Module und Services der Plattform zugreifen, damit wir die volle Kontrolle über die Verwaltung, Wartung und Sicherheit der gesamten Plattform haben und sämtliche administrative Aufgaben effizient ausführen können.

AD2. Als Adminstratoren möchten wir unsere privaten Daten inspizieren, hochladen, verwenden und löschen können, um interne Tests und Anpassungen durchführen zu können.

AD3. Als Adminstratoren möchten wir unsere öffentliche Daten inspizieren, hochladen, verwenden und löschen können, um die Plattform aktuell und relevant zu halten.

AD4. Als Adminstratoren möchten wir unsere Datenbankanbindungen inspizieren, verbinden, verwenden und entfernen können, um eine optimale Integration der Datenquellen zu gewährleisten.

AD5. Als Adminstratoren möchten wir unsere synthetisch generierten Daten inspizieren, herunterladen, verwenden und löschen können, um die Qualität und Anwendbarkeit der Daten zu überprüfen.

AD6. Als Adminstratoren möchten wir die privaten Daten anderer Nutzer nur löschen können, um Datenschutz und Compliance zu gewährleisten.

AD7. Als Adminstratoren möchten wir die öffentlichen Daten anderer Nutzer inspizieren, verwenden und löschen können, um die Integrität und Qualität der auf der Plattform verfügbaren Daten zu sichern.

AD8. Als Adminstratoren möchten wir die Datenbankanbindungen anderer Nutzer inspizieren, verwenden und entfernen können, um Sicherheitsrisiken zu minimieren.

AD9. Als Adminstratoren möchten wir die synthetisch generierten Daten anderer Nutzer inspizieren, herunterladen, verwenden und löschen können, um die Einhaltung der Qualitätsstandards zu überprüfen und zu fördern.

**Support:**

SP1. Als Support-User können wir auf alle generativen Module und Services der Plattform zugreifen, damit wir die Nutzeranfragen zeitnah bearbeiten und Benutzer bei der effizienten Nutzung der Plattform unterstützen können.

SP2. Als Support-User möchten wir unsere private Daten inspizieren, hochladen, verwenden und löschen können, um Support-Aufgaben effektiv zu managen und eigene Analysen durchzuführen.

SP3. Als Support-User möchten wir unsere synthetisch generierten Daten inspizieren, herunterladen, verwenden und löschen können, um deren Qualität sicherzustellen und Optimierungen vorzunehmen.

SP4. Als Support-User möchten wir die öffentlichen Daten anderer Nutzer inspizieren und verwenden können, um Support bei Problemlösungen anzubieten und Nutzerinteraktionen zu verbessern.

SP5. Als Support-User möchten wir die Datenbankanbindungen anderer Nutzer inspizieren und verwenden können, um technische Unterstützung zu bieten und Konnektivitätsprobleme zu beheben.

SP6. Als Support-User möchten wir die synthetisch generierten Daten anderer Nutzer inspizieren, herunterladen und verwenden können, um Nutzern bei der Verbesserung ihrer Modelle und Anwendungen zu assistieren.

**Auditoren:**

AU1. Als Auditoren können wir auf alle generativen Module und Services der Plattform zugreifen, damit wir die Sicherheit und Compliance überwachen, regelkonforme Prozesse sicherstellen und detaillierte Audits durchführen können, um die Integrität der Plattform zu gewährleisten.

AU2. Als Auditoren möchten wir unsere private Daten inspizieren, hochladen, verwenden und löschen können, um Compliance- und Sicherheitsüberprüfungen durchzuführen.

AU3. Als Auditoren möchten wir unsere synthetisch generierten Daten inspizieren, herunterladen, verwenden und löschen können, um die Einhaltung von Qualitätsstandards zu gewährleisten.

AU4. Als Auditoren möchten wir die öffentlichen Daten anderer Nutzer inspizieren und verwenden können, um deren Korrektheit und Konformität mit den Plattformrichtlinien zu überprüfen.

AU5. Als Auditoren möchten wir die Datenbankanbindungen anderer Nutzer inspizieren und verwenden können, um die Sicherheit und ordnungsgemäße Konfiguration zu gewährleisten.

AU6. Als Auditoren möchten wir die synthetisch generierten Daten anderer Nutzer inspizieren, herunterladen und verwenden können, um die Qualität und Angemessenheit dieser Daten sicherzustellen.

**Contributor:**

CO1. Als Contributor können wir auf alle generativen Module und Services sowie auf die Contribution Module der Plattform zugreifen, um neue Modelle, Metriken und Datenquellen hinzuzufügen, damit wir die Plattform kontinuierlich erweitern und verbessern können.

CO2. Als Contributor möchten wir unsere privaten Daten inspizieren, hochladen, verwenden und löschen können, um eigene Forschungs- und Entwicklungsprojekte effektiv durchzuführen.

CO3. Als Contributor möchten wir unsere öffentlichen Daten inspizieren, hochladen, verwenden und löschen können, um zur Plattform-Community beizutragen und Ressourcen für andere Entwickler bereitzustellen.

CO4. Als Contributor möchten wir unsere Datenbankanbindungen inspizieren, verbinden, verwenden und entfernen können, um eine nahtlose Datenintegration und Effizienz bei der Arbeit mit verschiedenen Datenquellen zu gewährleisten.

CO5. Als Contributor möchten wir unsere synthetisch generierten Daten inspizieren, herunterladen, verwenden und löschen können, um die Genauigkeit und Relevanz unserer Arbeit zu überprüfen und zu optimieren.

CO6. Als Contributor möchten wir die öffentlichen Daten anderer Nutzer inspizieren und verwenden können, um Kollaborationen zu fördern und von bewährten Methoden zu lernen.

CO7. Als Contributor möchten wir die Datenbankanbindungen anderer Nutzer inspizieren und verwenden können, um Synergien zu erkennen und gemeinsame Projekte effizienter zu gestalten.

CO8. Als Contributor möchten wir die synthetisch generierten Daten anderer Nutzer inspizieren, herunterladen und verwenden können, um unsere Modelle und Ansätze durch Vergleiche weiterzuentwickeln und zu verbessern.

**Anwender:**

AN1. Als Anwender können wir nur auf die generativen Module und Services der Plattform zugreifen, damit wir synthetische Daten generieren können.

AN2. Als Anwender möchten wir unsere privaten Daten inspizieren, hochladen, verwenden und löschen können, um unsere individuellen Forschungs- und Entwicklungsziele zu unterstützen.

AN3. Als Anwender möchten wir unsere synthetisch generierten Daten inspizieren, herunterladen, verwenden und löschen können, um die Ergebnisse zu validieren und weiter zu optimieren.

AN4. Als Anwender möchten wir die öffentlichen Daten anderer Nutzer inspizieren und verwenden können, um von verfügbaren Ressourcen zu lernen.

AN5. Als Anwender möchten wir die synthetisch generierten Daten anderer Nutzer inspizieren, herunterladen und verwenden können, um Vergleichsanalysen durchzuführen und unsere Forschung zu erweitern.

### 3.4.5 User Stories - Benutzer

Die in diesen User Stories detailliert beschriebenen Anforderungen und Szenarien leisten einen wesentlichen Beitrag. Sie sind essenziell, um die spezifischen Berechtigungen und Zugriffsmöglichkeiten der Benutzerrollen auf die für sie verfügbaren Services der DaFne-Plattform präzise zu definieren. Die User Stories dienen als Grundlage für die Konzeption des rollenbasierten Berechtigungsmodells, welches im späteren Verlauf die Sicherstellung des Zugriffs der Anwender auf die für sie vorgesehenen Services gewährleistet. Infolgedessen leisten sie einen maßgeblichen Beitrag zur Konzeption des Berechtigungsmodells, indem sie die spezifischen Bedürfnisse und Anforderungen der verschiedenen Nutzergruppen abbilden und somit eine gezielte und kontrollierte Zugriffskontrolle ermöglichen.

**Super-Admin:**

SA1. Als Super-Admin habe ich auf alle Funktions- und Inhaltsressourcen bezogen auf meinen eigenen Nutzer vollen Zugriff, damit ich meinen eigenen Nutzer vollumfänglich verwalten kann.

SA2. Als Super-Admin habe ich auf alle Funktions- und Inhaltsressourcen bezogen auf andere Nutzer vollen Zugriff, damit ich die vollumfängliche Verwaltung für alle Nutzer bieten kann.

**Plattform-Admin:**

PA1. Als Plattform-Admin habe ich auf alle Funktions- und Inhaltsressourcen bezogen auf meinen eigenen Nutzer vollen Zugriff, um die Verwaltung für meinen eigenen Nutzer effizient durchführen zu können.

PA2. Als Plattform-Admin habe ich auf alle Funktions- und Inhaltsressourcen bezogen auf andere Nutzer Änderungs- und Löschrechte, um die Verwaltung und Sicherheit der Plattform zu gewährleisten, indem ich Informationen von Nutzern bearbeiten oder Nutzerkonten löschen kann falls erforderlich.

**Support-User:**

SU1. Als Support-User habe ich auf die Funktions- und Inhaltsressourcen: Support Tickets, Personal Data und Generate/Evaluate/Analyze Data Management bezogen auf meinen eigenen Nutzer vollen Zugriff, damit ich umfassende Unterstützung bieten kann.

SU2. Als Support-User habe ich auf die Funktions- und Inhaltsressourcen: User, Role, Permisson und Policies/Rules Management bezogen auf meinen eigenen Nutzer nur Leserechte, damit ich Einsicht in relevante Informationen habe.

SU3. Als Support-User habe ich auf die Funktions- und Inhaltsressourcen: Extend Plattform Management für Generative Models, Evaluation Metrics und Data Sources bezogen auf meinen eigenen Nutzer keinen Zugriff, damit die Integrität der Plattform gewahrt bleibt.

SU4. Als Support-User habe ich auf die Funktions- und Inhaltsressource: Support Tickets Management bezogen auf andere Nutzer vollen Zugriff, um effektiven Support zu gewährleisten.

SU5. Als Support-User habe ich auf die Funktions- und Inhaltsressourcen: Generate/ Evaluate/Analyze Data Management bezogen auf andere Nutzer Änderungs- und Löschrechte, um sicherzustellen, dass Daten korrekt verwaltet werden.

SU6. Als Support-User habe ich auf die Funktions- und Inhaltsressourcen: User, Role, Permission und Personal Data Management sowie Extend Platform Management für

Generative Models, Evaluation Metrics und Data Sources bezogen auf andere Nutzer Lese- und Hinzufügerechte, um Informationen zu überprüfen und ergänzen zu können.

SU7. Als Support-User habe ich auf die Funktions- und Inhaltsressource: Policies und Rules Management bezogen auf andere Nutzer nur Lesenrechte, um die Richtlinien im Blick zu haben.

**Auditor:**

AUD1. Als Auditor habe ich auf die Funktions- und Inhaltsressourcen: Policies/Rules, Personal Data und Generate/Evaluate/Analyze Data Management bezogen auf meinen eigenen Nutzer vollen Zugriff, damit ich umfassend auf alle relevanten Informationen zugreifen und diese verwalten kann.

AUD2. Als Auditor habe ich auf die Funktions- und Inhaltsressource: Support Tickets Management bezogen auf meinen eigenen Nutzer Änderungs- und Löschrechte, damit ich meine Support-Tickets effizient bearbeiten kann.

AUD3. Als Auditor habe ich auf die Funktions- und Inhaltsressourcen: User, Role und Permisson Management bezogen auf meinen eigenen Nutzer nur Leserechte, damit ich die Informationen einsehen, aber keine Änderungen vornehmen kann.

AUD4. Als Auditor habe ich auf die Funktions- und Inhaltsressource: Extend Plattform Management für Generative Models, Evaluation Metrics und Data Sources bezogen auf meinen eigenen Nutzer keinen Zugriff, um sicherzustellen, dass ich nicht unbeabsichtigt Änderungen vornehmen kann.

AUD5. Als Auditor habe ich auf die Funktions- und Inhaltsressource: Policies and Rules Management bezogen auf andere Nutzer vollen Zugriff, damit ich die Regeln und Richtlinien effektiv verwalten kann.

AUD6. Als Auditor habe ich auf die Funktions- und Inhaltsressourcen: Generate/Evaluate/Analyze Management sowie Extend Platform Management für Generative Models, Evaluation Metrics und Data Sources bezogen auf andere Nutzer nur Leserechte, um Einblicke zu erhalten, aber keine direkten Änderungen vornehmen zu können.

AUD7. Als Auditor habe ich auf die Funktions- und Inhaltsressourcen: User, Role, Permisson, Support Tickets und Personal Data Management bezogen auf andere Nutzer keinen Zugriff, um die sensiblen Daten und Einstellungen der anderen Nutzer zu schützen.

**Developer Models:**

DMO1. Als Developer Models habe ich auf die Funktions- und Inhaltsressourcen: Personal Data und Generate/Evaluate/Analyze Data Management sowie Extend Platform Management für Generative Models bezogen auf meinen eigenen Nutzer vollen Zugriff, um effektiv auf diese Ressourcen zuzugreifen.

DMO2. Als Developer Models habe ich auf die Funktions- und Inhaltsressource: Support Tickets Management bezogen auf meinen eigenen Nutzer Änderungs- und Löschrechte, um Support-Tickets effizient zu verwalten.

DMO3. Als Developer Models habe ich auf die Funktions- und Inhaltsressourcen: User, Role, Permission und Policies/Rule Management bezogen auf meinen eigenen Nutzer nur Leserechte, um Einblicke in diese Bereiche zu erhalten.

DMO4. Als Developer Models habe ich auf die Funktions- und Inhaltsressource: Extend Platform Management für Evaluation Metrics und Data Sources bezogen auf meinen eigenen Nutzer keinen Zugriff, um die Integrität und Sicherheit dieser Ressourcen zu wahren.

DMO5. Als Developer Models habe ich auf die Funktions- und Inhaltsressource: Extend Platform Management für Generative Models bezogen auf andere Nutzer Lese- und Hinzufügerechte, um aktiv an der Plattform zu partizipieren.

DMO6. Als Developer Models habe ich auf die Funktions- und Inhaltsressource: Generate/Evaluate/Analyze Data Management bezogen auf andere Nutzer Leserechte, um Einsicht in die von ihnen generierten Daten zu bekommen.

DMO7. Als Developer Models habe ich auf die Funktions- und Inhaltsressourcen: User, Role, Permission, Policies/Rules, Support Ticket und Personal Management sowie Extend Platform Management für Evaluation Metrics und Data Sources bezogen auf andere Nutzer keinen Zugriff, um die Datensicherheit und -integrität zu gewährleisten.

**Developer Metrics:**

DME1. Als Developer Metrics habe ich auf die Funktions- und Inhaltsressourcen: Personal Data und Generate/Evaluate/Analyze Data Management sowie Extend Platform Management für Evaluation Metrics bezogen auf meinen eigenen Nutzer vollen Zugriff, damit ich umfassend arbeiten kann.

DME2. Als Developer Metrics habe ich auf die Funktions- und Inhaltsressource: Support Tickets Management bezogen auf meinen eigenen Nutzer Änderungs- und Löschrechte, damit du effizient Support-Tickets verwalten kannst.

DME3. Als Developer Metrics habe ich auf die Funktions- und Inhaltsressourcen: User, Role, Permission und Policies/Rule Management bezogen auf meinen eigenen Nutzer nur Leserechte, um Einblicke zu erhalten ohne Editiermöglichkeiten.

DME4. Als Developer Metrics habe ich auf die Funktions- und Inhaltsressource: Extend Platform Management für Generative Models und Data Sources bezogen auf meinen eigenen Nutzer keinen Zugriff, um die Integrität und Sicherheit dieser Ressourcen zu gewährleisten.

DME5. Als Developer Metrics habe ich auf die Funktions- und Inhaltsressource: Extend Platform Management für Evaluation Metrics bezogen auf andere Nutzer Lese- und Hinzufügerechte, um Beiträge zu dieser Ressource zu ermöglichen.

DME6. Als Developer Metrics habe ich auf die Funktions- und Inhaltsressource: Generate/Evaluate/Analyze Data Management bezogen auf andere Nutzer Leserechte, um Einblicke zu erhalten.

DME7. Als Developer Metrics habe ich auf die Funktions- und Inhaltsressourcen: User, Role, Permission, Policies/Rules, Support Ticket und Personal Management sowie Extend Platform Management für Generative Models und Data Sources bezogen auf andere Nutzer keinen Zugriff, um Sicherheit und Datenschutz zu gewährleisten.

**Data Engineer:**

DE1. Als Data Engineer habe ich auf die Funktions- und Inhaltsressourcen: Personal Data und Generate/Evaluate/Analyze Data Management sowie Extend Platform Management für Data Sources bezogen auf meinen eigenen Nutzer vollen Zugriff, um effektiv auf alle relevanten Daten und Funktionen zugreifen zu können.

DE2. Als Data Engineer habe ich auf die Funktions- und Inhaltsressource: Support Tickets Management bezogen auf meinen eigenen Nutzer Änderungs- und Löschrechte, um Support-Anfragen effizient bearbeiten zu können.

DE3. Als Data Engineer habe ich auf die Funktions- und Inhaltsressourcen: User, Role, Permission und Policies/Rule Management bezogen auf meinen eigenen Nutzer nur

Leserechte, um Einblicke in die Nutzer- und Rollenverwaltung zu erhalten, ohne Eingriffsmöglichkeiten.

DE4. Als Data Engineer habe ich auf die Funktions- und Inhaltsressource: Extend Platform Management für Generative Models und Evaluation Metrics bezogen auf meinen eigenen Nutzer keinen Zugriff, um die Integrität und Sicherheit dieser spezifischen Funktionen zu gewährleisten.

DE5. Als Data Engineer habe ich auf die Funktions- und Inhaltsressource: Extend Platform Management für Data Sources bezogen auf andere Nutzer Lese- und Hinzufügenrechte, um Datenquellen zu erkunden und gegebenenfalls neue hinzuzufügen.

DE6. Als Data Engineer habe ich auf die Funktions- und Inhaltsressource: Generate/Evaluate/Analyze Data Management bezogen auf andere Nutzer Leserechte, um Einblicke in die von ihnen generierten und evaluierten Daten zu erhalten.

DE7. Als Data Engineer habe ich auf die Funktions- und Inhaltsressourcen: User, Role, Permission, Policies/Rules, Support Ticket und Personal Management sowie Extend Platform Management für Generative Models und Evaluation Metrics bezogen auf andere Nutzer keinen Zugriff, um die Sicherheit und Datenschutzbestimmungen der Plattform zu wahren.

**Data Scientist:**

DS1. Als Data Scientist habe ich auf die Funktions- und Inhaltsressourcen: Personal Data und Generate/Evaluate/Analyze Data Management bezogen auf meinen eigenen Nutzer vollen Zugriff, um umfassend auf entsprechende Daten zugreifen und diese verwalten zu können.

DS2. Als Data Scientist habe ich auf die Funktions- und Inhaltsressource: Support Tickets Management bezogen auf meinen eigenen Nutzer Änderungs- und Löschrechte, um Support-Anfragen effizient bearbeiten zu können.

DS3. Als Data Scientist habe ich auf die Funktions- und Inhaltsressourcen: User, Role, Permission und Policies/Rules Management bezogen auf meinen eigenen Nutzer Leserechte, um relevante Informationen einsehen zu können.

DS4. Als Data Scientist habe ich auf die Funktions- und Inhaltsressourcen: Extend Platform Management für Generative Models, Evaluation Metrics und Data Sources bezogen

auf meinen eigenen Nutzer keinen Zugriff, um die Interaktion mit diesen spezifischen Entwickler-Ressourcen zu beschränken.

DS5. Als Data Scientist habe ich auf die Funktions- und Inhaltsressource: Generate/Evaluate/ Analyze Data Management bezogen auf andere Nutzer Änderungs- und Hinzufügerechte, um Daten anderer Nutzer zu ergänzen und zu analysieren.

DS6. Als Data Scientist habe ich auf die Funktions- und Inhaltsressourcen: User, Role, Permission, Policies/Rules, Support Tickets, Personal Data und Generate/Evaluate/Analyze Data Management sowie Extend Platform Management für Generative Models, Evaluation Metrics und Data Sources sowie bezogen auf andere Nutzer keine Rechte, um die Privatsphäre und Integrität der Daten anderer Nutzer zu wahren.

## 3.5 Benutzerprozess

Im nachfolgenden Unterkapitel erfolgt eine Auseinandersetzung mit den auf der DaFne-Plattform vorhandenen Benutzerprozessen.

### 3.5.1 Definition

Ein Benutzerprozess ist eine grafische Darstellung des Weges, den ein Benutzer innerhalb einer Anwendung zurücklegt, um eine bestimmte Tätigkeit auszuführen. Produktentwicklungsteams entwickeln solche Benutzerpfade mit dem Ziel, die Produktnutzung intuitiv zu gestalten. Der Schwerpunkt liegt darauf, den Benutzern die relevanten Informationen zum richtigen Zeitpunkt zur Verfügung zu stellen und sie in die Lage zu versetzen, ihre Ziele effizient zu erreichen, indem die erforderlichen Schritte auf ein Minimum reduziert werden (vgl. [2]).

### 3.5.2 Benutzerprozess - Standard Nutzer

Im Anhang A.2 ist ein Benutzerprozess für einen Standardbenutzer auf der DaFne-Plattform von Julia Seufert, Masterstudentin der Informatik an der HAW Hamburg, modelliert. Der Prozess wird im Folgenden beschrieben:

Ein Benutzer besucht die DaFne-Website und es wird geprüft, ob dieser Benutzer bereits einen Account hat. Ist dies der Fall, kann sich der Benutzer einloggen. Wenn nicht, muss er

sich registrieren und anschließend ein Onboarding durchführen. In diesem Schritt wird der Rollenzuweisungsprozess angewendet, wie er in Abschnitt 3.5.3 modelliert und dargestellt wird.

Es gibt zwei verschiedene Nutzungsszenarien: *Usage* und *Contribution*. Bei Auswahl von *Contribution* wird geprüft, ob der Benutzer ein Entwickler ist. Ist dies der Fall, erhält er Zugang zum *Contribution*-Workflow, andernfalls wird ihm der Zugang verweigert. Im *Contribution*-Workflow kann der Benutzer eigene Metriken, Modelle oder Demo-Daten hochladen. Die hochgeladenen Ressourcen werden geprüft; unzulässige Ressourcen führen zu einer Fehlermeldung, während zulässige Ressourcen freigegeben und der Workflow abgeschlossen wird.

Bei Auswahl von *Usage* gibt es zwei Hauptbereiche: *Generierung synthetischer Daten* und *Use Case Explorer*. Der Bereich *Generierung synthetischer Daten* ist weiter unterteilt. Im ersten Bereich, *Regelbasiert*, kann der Benutzer eigene oder Demo-Daten auswählen und Regeln für die Datengenerierung festlegen. Im zweiten Bereich, *Data Fusion*, müssen zwei Datensätze ausgewählt und Parameter definiert werden. Im dritten Bereich, *Reproduktion*, wählt der Benutzer eine Metrik, Daten und ein Modell aus und legt Parameter für das Modelltraining fest. Nach der Generierung kann der Benutzer entscheiden, wie die Daten ausgewertet werden sollen.

Schließlich wird der Workflow mit dem Download der Daten abgeschlossen.

### 3.5.3 Rollenvalidierungsprozess

Nach (vgl. [44]; [15]) werden im Kontext der Rollenvalidierung im Bereich des Identitäts- und Berechtigungsmanagements im Wesentlichen zwei Methoden zur Überprüfung von Rollenzuordnungen unterschieden: der manuelle und der automatisierte Ansatz. Beide verfolgen das Ziel, die korrekte Zuordnung von Rollen und Berechtigungen zu Benutzern auf Basis ihrer Aufgaben und Verantwortlichkeiten sicherzustellen, gehen dabei aber unterschiedlich vor. Eine Modellierung dieser Ansätze als BPMN-Prozesse ist im Anhang A.3 dargestellt.

**Manueller Rollenvalidierungsprozess:**

Am Anfang steht in der Regel eine Rollenanfrage eines Benutzers, woraufhin die notwendigen Benutzerinformationen gesammelt werden. Ein wichtiger Schritt im manuellen

Prozess ist, dass der Administrator diese Informationen zur Benutzeridentität mit bestehenden Richtlinien, Regeln und Standards vergleicht, bevor der Validierungsprozess fortgesetzt wird. Dies kann z.B. eine Überprüfung der Branchenzugehörigkeit oder des beruflichen Status des Benutzers beinhalten. Nach dieser Überprüfung wird eine Entscheidung getroffen: Wird der Antrag genehmigt, informiert der Administrator den Nutzer über die Genehmigung und weist ihm die neue Rolle zu, womit der Prozess erfolgreich abgeschlossen ist. Im Falle einer Ablehnung informiert der Administrator den Benutzer über die Gründe, was zur Folge hat, dass der Benutzer die Berechtigungen für die beantragte Rolle nicht erhält.

**Automatischer Rollenvalidierungsprozess:**

Auch dieser Prozess beginnt mit der Beantragung einer Rolle durch einen Benutzer und der Erfassung der erforderlichen Benutzerinformationen. Im Gegensatz zum manuellen Prozess erfolgt hier der Abgleich der Benutzerinformationen automatisch durch das System, basierend auf vordefinierten Regeln und Richtlinien. Dies kann z.B. eine Überprüfung der Branchenzugehörigkeit oder der beruflichen Stellung des Nutzers beinhalten. Wird der Antrag aufgrund dieses automatischen Abgleichs genehmigt, informiert das System den Benutzer über die ihm zugewiesene Rolle und der Prozess ist erfolgreich abgeschlossen. Wird der Antrag abgelehnt, informiert das System den Benutzer entsprechend und der Benutzer erhält keine Zugriffsrechte für die beantragte Rolle.

**Selbstregistrierung:**

Im Kontext der manuellen Rollenvalidierung kann die Rolle des Administrators auch von spezialisierten Benutzerrollen übernommen werden, wie z.B. dem Auditor als 'Power User' oder dem Support User als 'Light Admin'. Diese spezialisierten Rollen können mit zusätzlichen Berechtigungen ausgestattet werden, die es ihnen ermöglichen, ähnliche Aufgaben wie ein herkömmlicher Administrator durchzuführen, jedoch in der Regel mit einem spezifischeren oder eingeschränkteren Verantwortungsbereich.

Beispielsweise könnte der Auditor als Power-User in der Lage sein, zusätzlich die Konformität und Sicherheit der Rollenzuweisungen zu überprüfen, um sicherzustellen, dass diese den höchsten Standards entsprechen. In einer solchen Rolle wäre der Auditor besonders nützlich, um Sicherheitsüberprüfungen durchzuführen oder die Einhaltung strenger regulatorischer Anforderungen sicherzustellen.

Im Gegensatz dazu könnte der Support User als Light Admin hauptsächlich für die erste Ebene der Überprüfung und Genehmigung von Rollenanfragen zuständig sein. Diese

Rolle wäre besonders in einem Umfeld relevant, in dem schnelle Reaktionszeiten bei der Rollenvergabe erforderlich sind, z.B. um temporären Teams oder Projektmitarbeitern kurzfristig Zugriff zu gewähren.

Die Integration dieser spezialisierten Benutzerrollen in den manuellen Rollenvalidierungsprozess kann die Flexibilität und Effizienz des Rollenmanagements erhöhen, ohne die Sicherheit oder Compliance zu beeinträchtigen.

# 4 Konzeption eines rollenbasierten Berechtigungskonzepts

In diesem Kapitel wird die Konzeption des rollenbasierten Berechtigungskonzepts behandelt.

## 4.1 Konzeption der Berechtigungsmatrix

In Anknüpfung an die Darstellung des Requirements Engineering Prozesses im vorangehenden Kapitel wurden drei Berechtigungsmatrizen konzeptioniert, die eine Differenzierung der verschiedenen Arten von Ressourcen bzw. Objekten ermöglichen.

Die in den Kapiteln 4.2, 4.3 und 4.4 dargestellten Entwürfe basieren auf den in den Kapiteln 3.4.4 und 3.4.5 aufgeführten User Stories. Die User Stories wurden mit Absprache des Entwicklungsteams von DaFne erstellt.

### 4.1.1 Eigen- und Fremdberechtigungen

Eigenberechtigungen beziehen sich auf die Zugriffsrechte, die ein bestimmter Benutzer auf sein eigenes Konto, Daten oder andere Ressourcen hat. Diese Berechtigungen beziehen sich ausschließlich auf den jeweiligen Benutzer und dessen Sichtweise auf die Plattform. Als Beispiel seien hier Zugriffsberechtigungen für persönliche Daten genannt.

Fremdberechtigungen hingegen beziehen sich auf die Zugriffsrechte, die ein bestimmter Benutzer auf die Konten, Daten oder Ressourcen anderer Nutzer hat. Diese Berechtigungen beziehen sich auf andere Benutzer und deren Sicht auf die Plattform. Ein Benutzer mit Fremdberechtigungen kann beispielsweise Lese- oder Schreibrechte auf die generierten Daten anderer Benutzer haben, jedoch nicht auf seine privaten Daten.

In Bezug auf die Berechtigungsmatrixen für Funktionsressourcen, Inhaltsressourcen und Funktions- sowie Inhaltsressourcen stellen Eigen- und Fremdberechtigungen die Grundlage für die Zuweisung und Verwaltung von Zugriffsrechten dar. Durch die klare Unterscheidung dieser Berechtigungen können Administratoren effektiv steuern, wer auf welche Ressourcen zugreifen kann und welche Aktionen erlaubt sind. Dies trägt maßgeblich zur Sicherheit und Integrität der Plattform bei, indem unbefugte Zugriffe verhindert werden.

### 4.1.2 Berechtigungsmatrix: Funktionsressourcen

Die vorliegende Berechtigungsmatrix thematisiert den Zugang zu Funktionsressourcen, d. h. Modulen bzw. Services. Die vorliegende Berechtigungsmatrix definiert, welche Module für die jeweilige Benutzergruppe verfügbar sind und welche nicht verfügbar sind.

| Berechtigungsmatrix DaFne \| Funktionsressourcen | |
|---|---|
| **Module (Services)** | **Benutzergruppe n** |
| Rule-Based | **verfügbar / nicht verfügbar** |
| Fusion | **verfügbar / nicht verfügbar** |
| Reproduction (inkl. Evaluation der Daten anhand von Evaluierungsmetriken) | **verfügbar / nicht verfügbar** |
| Generation Quality Report | **verfügbar / nicht verfügbar** |
| Data Report | **verfügbar / nicht verfügbar** |
| Neighbourhood Generator | **verfügbar / nicht verfügbar** |
| Mobilitätsdaten Generator | **verfügbar / nicht verfügbar** |
| Analyzer (EIN DATENSATZ) | **verfügbar / nicht verfügbar** |
| Use Case Explorer | **verfügbar / nicht verfügbar** |
| Add Generative Model | **verfügbar / nicht verfügbar** |
| Add Evaluation Metric | **verfügbar / nicht verfügbar** |
| Add Data Source | **verfügbar / nicht verfügbar** |

Abbildung 4.1: Konzeption der Berechtigungsmatrix Funktionsressourcen

### 4.1.3 Berechtigungsmatrix: Inhaltsressourcen

Die vorliegende Berechtigungsmatrix behandelt die Zugänglichkeit von Inhaltsressourcen, d.-h. Daten und deren Arten, für die jeweilige Nutzergruppe. Im Folgenden werden die Begriffe „inspizieren", „hochladen", „herunterladen", „verwenden", „löschen", „verbinden" und „entfernen" erläutert.

- Inspizieren: bezeichnet die Betrachtung oder Überprüfung von Daten.

- Hochladen: bezeichnet den Prozess des Hinzufügens von Daten auf der Plattform.

- Herunterladen: bezeichnet den Prozess des Abrufs oder der Speicherung von Daten von der Plattform auf ein lokales Gerät.

- Verwendung: erfolgt durch die Nutzer durch die Ausführung der folgenden Aktionen mit den Daten, beispielsweise das Bearbeiten, Analysieren oder Anwenden von Operationen auf diesen.

- Löschen: bezeichnet den Prozess des Eliminierens von Daten von der Plattform.

- Verbinden: bezeichnet die Verknüpfung verschiedener Datensätze oder Ressourcen und stellt eine weitere Möglichkeit der Interaktion dar.

- Entfernen: bezeichnet den Prozess des Trennens oder Aufhebens einer Verbindung zu Daten.

**Berechtigungsmatrix DaFne | Inhaltsressourcen**

| Art der Daten | Benutzergruppe n |
|---|---|
| Private Data | |
| | Out of scope/inspizieren/hochladen/verwenden/löschen |
| Public Data | |
| | Out of scope/inspizieren/hochladen/verwenden/löschen |
| Public Data Interface (Datenbankanbindungen) | |
| | Out of scope/inspizieren/verbinden/verwenden/entfernen |
| Own Generated Data | |
| | Out of scope/inspizieren/herunterladen/verwenden/löschen |

Abbildung 4.2: Konzeption der Berechtigungsmatrix Inhaltsressourcen

### 4.1.4 Berechtigungsmatrix: Funktions- und Inhaltsressourcen

Die Berechtigungsmatrix definiert die Zugriffsrechte der Nutzer auf die Funktions- und Inhaltsressourcen. Die Zugriffsrechte werden durch die in Kapitel 2.6.3 definierten Stufen reguliert.

**Berechtigungsmatrix DaFne | Funktions- und Inhaltsressourcen**

| OBJEKTE / SUBJEKTE | Rolle n |
|---|---|
| User Management | Stufe 0/1/2/3/4 |
| Role Management | Stufe 0/1/2/3/4 |
| Permission Management | Stufe 0/1/2/3/4 |
| Policies und Rules Management | Stufe 0/1/2/3/4 |
| Support Tickets Management | Stufe 0/1/2/3/4 |
| Personal Data Management | Stufe 0/1/2/3/4 |
| Contribution - Extend Platform Management (Generative Models) | Stufe 0/1/2/3/4 |
| Contribution - Extend Platform Management (Evaluation Metrics) | Stufe 0/1/2/3/4 |
| Contribution - Extend Platform Management (Data Sources) | Stufe 0/1/2/3/4 |
| Generate/Evaluate/Analyze Data Management | Stufe 0/1/2/3/4 |

Abbildung 4.3: Konzeption der Berechtigungsmatrix Funktions- und Inhaltsressourcen

Nachfolgend nochmals die einzelnen Berechtigungsstufen inklusive der Operationen.


Abbildung 4.4: Berechtigungsstufen inklusive der jeweiligen Operationen (vgl. [49]: Abbildung 1.5)

## 4.2 Entwurf der Berechtigungsmatrix: Funktionsressourcen

In diesem Kapitel werden die Entwürfe der Berechtigungsmatrix: Funktionsressourcen behandelt.

### 4.2.1 Benutzergruppe: Adminstratoren

Es ist erforderlich, dass alle Module jeglicher Art für die Administratoren sichtbar und nutzbar sind. Basierend auf die folgende User Story (siehe S.69): AD1.

| Module (Services) | Benutzergruppe Adminstratoren |
|---|---|
| Rule-Based | verfügbar |
| Fusion | verfügbar |
| Reproduction (inkl. Evaluation der Daten anhand von Evaluierungsmetriken) | verfügbar |
| Generation Quality Report | verfügbar |
| Data Report | verfügbar |
| Neighbourhood Generator | verfügbar |
| Mobilitätsdaten Generator | verfügbar |
| Analyzer (EIN DATENSATZ) | verfügbar |
| Use Case Explorer | verfügbar |
| Add Generative Model | verfügbar |
| Add Evaluation Metric | verfügbar |
| Add Data Source | verfügbar |

Abbildung 4.5: Berechtigungsmatrix - Funktionsressourcen: Adminstratoren

### 4.2.2 Benutzergruppe: Support

Es ist sicherzustellen, dass alle Module, die eine Generierungsfunktion aufweisen, für die Nutzer des Supports sichtbar und nutzbar sind. Basierend auf die folgende User Story (siehe S.70): SP1.

| Module (Services) | Benutzergruppe Support |
|---|---|
| Rule-Based | verfügbar |
| Fusion | verfügbar |
| Reproduction (inkl. Evaluation der Daten anhand von Evaluierungsmetriken) | verfügbar |
| Generation Quality Report | verfügbar |
| Data Report | verfügbar |
| Neighbourhood Generator | verfügbar |
| Mobilitätsdaten Generator | verfügbar |
| Analyzer (EIN DATENSATZ) | verfügbar |
| Use Case Explorer | verfügbar |
| Add Generative Model | nicht verfügbar |
| Add Evaluation Metric | nicht verfügbar |
| Add Data Source | nicht verfügbar |

Abbildung 4.6: Berechtigungsmatrix - Funktionsressourcen: Support

### 4.2.3 Benutzergruppe: Auditoren

Des Weiteren ist sicherzustellen, dass alle Module, die eine Generierungsfunktion aufweisen, auch für die Auditoren sichtbar und nutzbar sind. Basierend auf die folgende User Story siehe Seite (siehe S.71): AU1.

| Module (Services) | Benutzergruppe Auditoren |
|---|---|
| Rule-Based | verfügbar |
| Fusion | verfügbar |
| Reproduction (inkl. Evaluation der Daten anhand von Evaluierungsmetriken) | verfügbar |
| Generation Quality Report | verfügbar |
| Data Report | verfügbar |
| Neighbourhood Generator | verfügbar |
| Mobilitätsdaten Generator | verfügbar |
| Analyzer (EIN DATENSATZ) | verfügbar |
| Use Case Explorer | verfügbar |
| Add Generative Model | nicht verfügbar |
| Add Evaluation Metric | nicht verfügbar |
| Add Data Source | nicht verfügbar |

Abbildung 4.7: Berechtigungsmatrix - Funktionsressourcen: Auditoren

### 4.2.4 Benutzergruppe: Contributor

Für die Contributor müssen ebenfalls alle Module sichtbar und nutzbar sein, die eine Generierungsfunktion aufweisen. Darüber hinaus müssen auch die Module einsehbar und verwendbar sein, die einen Beitrag zur Plattform leisten bzw. die Plattform erweitern. Dies kann beispielsweise durch das Hinzufügen neuer Modelle/Metriken/Datenquellen erfolgen. Basierend auf die folgende User Story (siehe S.71): CO1.

| Module (Services) | Benutzergruppe Contributor |
|---|---|
| Rule-Based | verfügbar |
| Fusion | verfügbar |
| Reproduction (inkl. Evaluation der Daten anhand von Evaluierungsmetriken) | verfügbar |
| Generation Quality Report | verfügbar |
| Data Report | verfügbar |
| Neighbourhood Generator | verfügbar |
| Mobilitätsdaten Generator | verfügbar |
| Analyzer (EIN DATENSATZ) | verfügbar |
| Use Case Explorer | verfügbar |
| Add Generative Model | verfügbar |
| Add Evaluation Metric | verfügbar |
| Add Data Source | verfügbar |

Abbildung 4.8: Berechtigungsmatrix - Funktionsressourcen: Contributor

### 4.2.5 Benutzergruppe: Anwender

Für den Anwender ist lediglich relevant, dass die Module sichtbar und nutzbar sind, die eine Generierungsfunktion aufweisen. Diese Module dienen der Generierung synthetischer Daten, da die Generierung von Daten das primäre Ziel darstellt. Basierend auf die folgende User Story (siehe S.73): AN1.

| Module (Services) | Benutzergruppe Anwender |
|---|---|
| Rule-Based | verfügbar |
| Fusion | verfügbar |
| Reproduction (inkl. Evaluation der Daten anhand von Evaluierungsmetriken) | verfügbar |
| Generation Quality Report | verfügbar |
| Data Report | verfügbar |
| Neighbourhood Generator | verfügbar |
| Mobilitätsdaten Generator | verfügbar |
| Analyzer (EIN DATENSATZ) | verfügbar |
| Use Case Explorer | verfügbar |
| Add Generative Model | nicht verfügbar |
| Add Evaluation Metric | nicht verfügbar |
| Add Data Source | nicht verfügbar |

Abbildung 4.9: Berechtigungsmatrix - Funktionsressourcen: Anwender

## 4.3 Entwurf der Berechtigungsmatrix: Inhaltsressourcen

In diesem Kapitel werden die Entwürfe der Berechtigungsmatrix: Inhaltsressourcen behandelt.

### 4.3.1 Benutzergruppe: Adminstratoren (Eigenberechtigungen)

Den Administratoren steht selbst ein uneingeschränkter Zugriff auf jede Art von Daten zur Verfügung. Basierend auf die folgenden User Stories (siehe S.69): AD2, AD3, AD4 und AD5.

| Art der Daten | Benutzergruppe Adminstratoren (Eigenberechtigungen) |
|---|---|
| Private Data | |
| | **inspizieren/hochladen/verwenden/löschen** |
| Public Data | |
| | **inspizieren/hochladen/verwenden/löschen** |
| Public Data Interface (Datenbankanbindungen) | |
| | **inspizieren/verbinden/verwenden/entfernen** |
| Own Generated Data | |
| | **inspizieren/herunterladen/verwenden/löschen** |

Abbildung 4.10: Berechtigungsmatrix - Inhaltsressourcen: Adminstratoren (Eigenberechtigungen)

### 4.3.2 Benutzergruppe: Adminstratoren (Fremdberechtigungen)

Administratoren verfügen gegenüber anderen Nutzern über uneingeschränkten Zugriff auf sämtliche Daten. Bei privaten Daten, die potenziell als sensibel eingestuft werden können, ist ihnen lediglich die Löschfunktion zugänglich. Basierend auf die folgenden User Stories (siehe S.69): AD6, AD7, AD8 und AD9.

| Art der Daten | Benutzergruppe Adminstratoren (Fremdberechtigungen) |
|---|---|
| Private Data | |
| | **löschen** |
| Public Data | |
| | **inspizieren/verwenden/löschen** |
| Public Data Interface (Datenbankanbindungen) | |
| | **inspizieren/verwenden/entfernen** |
| Own Generated Data | |
| | **inspizieren/herunterladen/verwenden/löschen** |

Abbildung 4.11: Berechtigungsmatrix - Inhaltsressourcen: Adminstratoren (Fremdberechtigungen)

### 4.3.3 Benutzergruppe: Support (Eigenberechtigungen)

Die Nutzer des Supports selbst haben lediglich Zugriff auf die privaten und generierten Daten, da sie keine öffentlichen Daten für Testzwecke hochladen und keine neuen Datenbanken anbinden können. Basierend auf die folgenden User Stories (siehe S.74): SU2 und SU3.

| Art der Daten | Benutzergruppe Support (Eigenberechtigungen) |
|---|---|
| Private Data | |
| | **inspizieren/hochladen/verwenden/löschen** |
| Public Data | |
| | **Out of scope** |
| Public Data Interface (Datenbankanbindungen) | |
| | **Out of scope** |
| Own Generated Data | |
| | **inspizieren/herunterladen/verwenden/löschen** |

Abbildung 4.12: Berechtigungsmatrix - Inhaltsressourcen: Support (Eigenberechtigungen)

### 4.3.4 Benutzergruppe: Support (Fremdberechtigungen)

Die Nutzer des Supports haben lediglich die Möglichkeit, die Daten zu inspizieren oder zu verwenden, mit Ausnahme der privaten Daten, welche nicht zugänglich sind. Basierend auf die folgenden User Stories (siehe S.74): SU4, SU5 und SU6.

| Art der Daten | Benutzergruppe Support (Fremdberechtigungen) |
|---|---|
| Private Data | |
| | **Out of scope** |
| Public Data | |
| | **inspizieren/verwenden** |
| Public Data Interface (Datenbankanbindungen) | |
| | **inspizieren/verwenden** |
| Own Generated Data | |
| | **inspizieren/herunterladen/verwenden** |

Abbildung 4.13: Berechtigungsmatrix - Inhaltsressourcen: Support (Fremdberechtigungen)

### 4.3.5 Benutzergruppe: Auditoren (Eigenberechtigungen)

Die Auditoren selbst haben nur Zugriff auf die privaten und generierten Daten, da sie keine öffentlichen Daten für Testzwecke hochladen und keine neuen Datenbanken anbin-

den dürfen. Dies ist ihnen nicht gestattet. Basierend auf die folgenden User Stories (siehe S.71): AU2 und AU3.

| Art der Daten | Benutzergruppe Auditoren (Eigenberechtigungen) |
|---|---|
| Private Data | |
| | **inspizieren/hochladen/verwenden/löschen** |
| Public Data | |
| | **Out of scope** |
| Public Data Interface (Datenbankanbindungen) | |
| | **Out of scope** |
| Own Generated Data | |
| | **inspizieren/herunterladen/verwenden/löschen** |

Abbildung 4.14: Berechtigungsmatrix - Inhaltsressourcen: Auditoren (Eigenberechtigungen)

### 4.3.6 Benutzergruppe: Auditoren (Fremdberechtigungen)

Die Auditoren sind lediglich dazu befugt, die Daten zu inspizieren oder zu verwenden, wobei private Daten davon ausgenommen sind. Basierend auf die folgenden User Stories (siehe S.71): AU4, AU5 und AU6.

| Art der Daten | Benutzergruppe Auditoren(Fremdberechtigungen) |
|---|---|
| Private Data | |
| | **Out of scope** |
| Public Data | |
| | **inspizieren/verwenden** |
| Public Data Interface (Datenbankanbindungen) | |
| | **inspizieren/verwenden** |
| Own Generated Data | |
| | **inspizieren/herunterladen/verwenden** |

Abbildung 4.15: Berechtigungsmatrix - Inhaltsressourcen: Auditoren (Fremdberechtigungen)

### 4.3.7 Benutzergruppe: Contributor (Eigenberechtigungen)

Den Contributors selbst wird uneingeschränkter Zugriff auf jegliche Art von Daten gewährt. Basierend auf die folgenden User Stories (siehe S.71): CO2, CO3, CO4 und CO5.

| Art der Daten | Benutzergruppe Contributor (Eigenberechtigungen) |
|---|---|
| Private Data | |
| | inspizieren/hochladen/verwenden/löschen |
| Public Data | |
| | inspizieren/hochladen/verwenden/löschen |
| Public Data Interface (Datenbankanbindungen) | |
| | inspizieren/verbinden/verwenden/entfernen |
| Own Generated Data | |
| | inspizieren/herunterladen/verwenden/löschen |

Abbildung 4.16: Berechtigungsmatrix - Inhaltsressourcen: Contributor (Eigenberechtigungen)

### 4.3.8 Benutzergruppe: Contributor (Fremdberechtigungen)

Den Contributoren ist lediglich gestattet, die Daten zu inspizieren oder zu verwenden. Ausgenommen hiervon sind private Daten. Basierend auf die folgenden User Stories (siehe S.71): CO6, CO7 und CO8.

| Art der Daten | Benutzergruppe Contributor (Fremdberechtigungen) |
|---|---|
| Private Data | |
| | Out of scope |
| Public Data | |
| | inspizieren/verwenden |
| Public Data Interface (Datenbankanbindungen) | |
| | inspizieren/verwenden |
| Own Generated Data | |
| | inspizieren/herunterladen/verwenden |

Abbildung 4.17: Berechtigungsmatrix - Inhaltsressourcen: Contributor (Fremdberechtigungen)

### 4.3.9 Benutzergruppe: Anwender (Eigenberechtigungen)

Die Anwender selbst haben lediglich Zugriff auf die privaten und generierten Daten, da ihnen die Berechtigung fehlt, öffentliche Daten für Testzwecke hochzuladen und neue Datenbanken anzubinden. Basierend auf die folgenden User Stories (siehe S.73): AN2 und AN3.

| Art der Daten | Benutzergruppe Anwender (Eigenberechtigungen) |
|---|---|
| Private Data | |
| | **inspizieren/hochladen/verwenden/löschen** |
| Public Data | |
| | **Out of scope** |
| Public Data Interface (Datenbankanbindungen) | |
| | **Out of scope** |
| Own Generated Data | |
| | **inspizieren/herunterladen/verwenden/löschen** |

Abbildung 4.18: Berechtigungsmatrix - Inhaltsressourcen: Anwender (Eigenberechtigungen)

### 4.3.10 Benutzergruppe: Anwender (Fremdberechtigungen)

Den Anwenderinnen und Anwendern ist nur die Inspektion und Verwendung der öffentlichen Daten, welche zu Testzwecken bereitgestellt wurden, sowie der generierten Daten gestattet. Basierend auf die folgenden User Stories (siehe S.73): AN4 und AN5.

| Art der Daten | Benutzergruppe Anwender (Fremdberechtigungen) |
|---|---|
| Private Data | |
| | **Out of scope** |
| Public Data | |
| | **inspizieren/verwenden** |
| Public Data Interface (Datenbankanbindungen) | |
| | **Out of scope** |
| Own Generated Data | |
| | **inspizieren/herunterladen/verwenden** |

Abbildung 4.19: Berechtigungsmatrix - Inhaltsressourcen: Anwender (Fremdberechtigungen)

## 4.4 Entwurf der Berechtigungsmatrix: Funktions- und Inhaltsressourcen

In diesem Kapitel werden die Entwürfe der Berechtigungsmatrix: Funktions- und Inhaltsressourcen behandelt.

### 4.4.1 Rolle: Super-Admin (Eigenberechtigungen)

Der Super-Admin selbst verfügt über uneingeschränkten Zugriff auf alle Ressourcen. Basierend auf die folgende User Story (siehe S.73): SA1.

| OBJEKTE / SUBJEKTE | Super-Admin (Eigenberechtigungen) |
|---|---|
| User Management | Stufe 4 |
| Role Management | Stufe 4 |
| Permission Management | Stufe 4 |
| Policies und Rules Management | Stufe 4 |
| Support Tickets Management | Stufe 4 |
| Personal Data Management | Stufe 4 |
| Contribution - Extend Platform Management (Generative Models) | Stufe 4 |
| Contribution - Extend Platform Management (Evaluation Metrics) | Stufe 4 |
| Contribution - Extend Platform Management (Data Sources) | Stufe 4 |
| Generate/Evaluate/Analyze Data Management | Stufe 4 |

Abbildung 4.20: Berechtigungsmatrix - Funktions- und Inhaltsressourcen: Super-Admin (Eigenberechtigungen)

### 4.4.2 Rolle: Super-Admin (Fremdberechtigungen)

Der Super-Admin verfügt über uneingeschränkten Zugriff auf alle Ressourcen anderer Nutzer. Basierend auf die folgende User Story (siehe S.73): SA2.

| OBJEKTE / SUBJEKTE | Super-Admin (Fremdberechtigungen) |
|---|---|
| User Management | Stufe 4 |
| Role Management | Stufe 4 |
| Permission Management | Stufe 4 |
| Policies und Rules Management | Stufe 4 |
| Support Tickets Management | Stufe 4 |
| Personal Data Management | Stufe 4 |
| Contribution - Extend Platform Management (Generative Models) | Stufe 4 |
| Contribution - Extend Platform Management (Evaluation Metrics) | Stufe 4 |
| Contribution - Extend Platform Management (Data Sources) | Stufe 4 |
| Generate/Evaluate/Analyze Data Management | Stufe 4 |

Abbildung 4.21: Berechtigungsmatrix - Funktions- und Inhaltsressourcen: Super-Admin (Fremdberechtigungen)

### 4.4.3 Rolle: Plattform-Admin (Eigenberechtigungen)

Der Plattform-Admin selbst ist in voller Weise autorisiert, auf alle Objekte zuzugreifen. Basierend auf die folgende User Story (siehe S.74): PA1.

| OBJEKTE / SUBJEKTE | Plattform-Admin (Eigenberechtigungen) |
|---|---|
| User Management | Stufe 4 |
| Role Management | Stufe 4 |
| Permission Management | Stufe 4 |
| Policies und Rules Management | Stufe 4 |
| Support Tickets Management | Stufe 4 |
| Personal Data Management | Stufe 4 |
| Contribution - Extend Platform Management (Generative Models) | Stufe 4 |
| Contribution - Extend Platform Management (Evaluation Metrics) | Stufe 4 |
| Contribution - Extend Platform Management (Data Sources) | Stufe 4 |
| Generate/Evaluate/Analyze Data Management | Stufe 4 |

Abbildung 4.22: Berechtigungsmatrix - Funktions- und Inhaltsressourcen: Plattform-Admin (Eigenberechtigungen)

### 4.4.4 Rolle: Plattform-Admin (Fremdberechtigungen)

Der Plattform-Admin ist befugt, alle Ressourcen anderer Nutzer zu modifizieren und zu entfernen. Basierend auf die folgende User Story (siehe S.74): PA2.

| OBJEKTE / SUBJEKTE | Plattform-Admin (Fremdberechtigungen) |
|---|---|
| User Management | Stufe 3 |
| Role Management | Stufe 3 |
| Permission Management | Stufe 3 |
| Policies und Rules Management | Stufe 3 |
| Support Tickets Management | Stufe 3 |
| Personal Data Management | Stufe 3 |
| Contribution - Extend Platform Management (Generative Models) | Stufe 3 |
| Contribution - Extend Platform Management (Evaluation Metrics) | Stufe 3 |
| Contribution - Extend Platform Management (Data Sources) | Stufe 3 |
| Generate/Evaluate/Analyze Data Management | Stufe 3 |

Abbildung 4.23: Berechtigungsmatrix - Funktions- und Inhaltsressourcen: Plattform-Admin (Fremdberechtigungen)

### 4.4.5 Rolle: Support-User (Eigenberechtigungen)

Der Support User hat keinen Zugriff auf die Ressourcen, mit denen er zur Plattform beitragen könnte. Lediglich auf die übrigen Ressourcen hat er Lesezugriff, um seine Rolle, seine Berechtigungen etc. einsehen zu können. Voller Zugriff besteht hingegen auf die persönlichen Daten, Daten, die durch den Support User generiert werden, sowie auf Supportangelegenheiten. Basierend auf die folgenden User Stories (siehe S.74): SU1, SU2 und SU3.

| OBJEKTE / SUBJEKTE | Support User (Eigenberechtigungen) |
|---|---|
| User Management | Stufe 1 |
| Role Management | Stufe 1 |
| Permission Management | Stufe 1 |
| Policies und Rules Management | Stufe 1 |
| Support Tickets Management | Stufe 4 |
| Personal Data Management | Stufe 4 |
| Contribution - Extend Platform Management (Generative Models) | Stufe 0 |
| Contribution - Extend Platform Management (Evaluation Metrics) | Stufe 0 |
| Contribution - Extend Platform Management (Data Sources) | Stufe 0 |
| Generate/Evaluate/Analyze Data Management | Stufe 4 |

Abbildung 4.24: Berechtigungsmatrix - Funktions- und Inhaltsressourcen: Support-User (Eigenberechtigungen)

### 4.4.6 Rolle: Support-User (Fremdberechtigungen)

Der Support User ist befugt, uneingeschränkten Zugriff auf die Ressourcen anderer Nutzer im Zusammenhang mit Support-Angelegenheiten zu nehmen, einschließlich der Datengenerierungsfunktionen sowie der Befugnisse zum Ändern und Löschen. Des Weiteren ist er befugt, andere Ressourcen, wie beispielsweise Rollen/Berechtigungen und zur Plattform beitragende Elemente, zu lesen und hinzuzufügen. Allerdings sind seine Rechte in Bezug auf Richtlinien und Regeln auf das Lesen beschränkt. Basierend auf die folgenden User Stories (siehe S.74): SU4, SU5, SU6 und SU7.

| OBJEKTE / SUBJEKTE | Support User (Fremdberechtigungen) |
|---|---|
| User Management | Stufe 2 |
| Role Management | Stufe 2 |
| Permission Management | Stufe 2 |
| Policies und Rules Management | Stufe 1 |
| Support Tickets Management | Stufe 4 |
| Personal Data Management | Stufe 2 |
| Contribution - Extend Platform Management (Generative Models) | Stufe 2 |
| Contribution - Extend Platform Management (Evaluation Metrics) | Stufe 2 |
| Contribution - Extend Platform Management (Data Sources) | Stufe 2 |
| Generate/Evaluate/Analyze Data Management | Stufe 3 |

Abbildung 4.25: Berechtigungsmatrix - Funktions- und Inhaltsressourcen: Support-User (Fremdberechtigungen)

### 4.4.7 Rolle: Auditor (Eigenberechtigungen)

Der Auditor selbst hat keinen Zugriff auf die Ressourcen, die es ihm ermöglichen, zur Plattform beizutragen. Er hat lediglich Lesezugriff auf die übrigen Ressourcen, um seine Rolle, seine Berechtigungen usw. einsehen zu können. Allerdings hat er vollen Zugriff auf persönliche Daten, Daten, die durch den Auditor generiert werden, sowie auf Richtlinien und Regeln. Basierend auf die folgenden User Stories (siehe S.75): AUD1, AUD2, AUD3 und AUD4.

| OBJEKTE / SUBJEKTE | Auditor (Eigenberechtigungen) |
|---|---|
| User Management | Stufe 1 |
| Role Management | Stufe 1 |
| Permission Management | Stufe 1 |
| Policies und Rules Management | Stufe 4 |
| Support Tickets Management | Stufe 3 |
| Personal Data Management | Stufe 4 |
| Contribution - Extend Platform Management (Generative Models) | Stufe 0 |
| Contribution - Extend Platform Management (Evaluation Metrics) | Stufe 0 |
| Contribution - Extend Platform Management (Data Sources) | Stufe 0 |
| Generate/Evaluate/Analyze Data Management | Stufe 4 |

Abbildung 4.26: Berechtigungsmatrix - Funktions- und Inhaltsressourcen: Auditor (Eigenberechtigungen)

### 4.4.8 Rolle: Auditor (Fremdberechtigungen)

Der Auditor hat uneingeschränkten Zugriff auf die Ressourcen anderer Nutzer im Zusammenhang mit Richtlinien und Regeln. Er verfügt lediglich über Leserechte in Bezug auf das Zur-Plattform-beitragen und die Datengenerierungsfunktionen. Auf alle anderen Ressourcen hat er keinen Zugriff. Basierend auf die folgenden User Stories (siehe S.75): AUD5, AUD6 und AUD7.

| OBJEKTE / SUBJEKTE | Auditor (Fremdberechtigungen) |
|---|---|
| User Management | Stufe 0 |
| Role Management | Stufe 0 |
| Permission Management | Stufe 0 |
| Policies und Rules Management | Stufe 4 |
| Support Tickets Management | Stufe 0 |
| Personal Data Management | Stufe 0 |
| Contribution - Extend Platform Management (Generative Models) | Stufe 1 |
| Contribution - Extend Platform Management (Evaluation Metrics) | Stufe 1 |
| Contribution - Extend Platform Management (Data Sources) | Stufe 1 |
| Generate/Evaluate/Analyze Data Management | Stufe 1 |

Abbildung 4.27: Berechtigungsmatrix - Funktions- und Inhaltsressourcen: Auditor (Fremdberechtigungen)

### 4.4.9 Rolle: Developer Models (Eigenberechtigungen)

Der Developer für generative Models hat selbst vollen Zugriff auf persönliche Daten, die Integration neuer generativer Modelle und die Anpassung persönlicher Daten. Er hat Änderungs- und Löschrechte für die Erstellung von Support-Tickets. In Bezug auf alle anderen Ressourcen hat er nur Leserechte. Basierend auf die folgenden User Stories (siehe S.76): DMO1, DMO2, DMO3 und DMO4.

| OBJEKTE / SUBJEKTE | Developer Models (Eigenberechtigungen) |
|---|---|
| User Management | Stufe 1 |
| Role Management | Stufe 1 |
| Permission Management | Stufe 1 |
| Policies und Rules Management | Stufe 1 |
| Support Tickets Management | Stufe 3 |
| Personal Data Management | Stufe 4 |
| Contribution - Extend Platform Management (Generative Models) | Stufe 4 |
| Contribution - Extend Platform Management (Evaluation Metrics) | Stufe 0 |
| Contribution - Extend Platform Management (Data Sources) | Stufe 0 |
| Generate/Evaluate/Analyze Data Management | Stufe 4 |

Abbildung 4.28: Berechtigungsmatrix - Funktions- und Inhaltsressourcen: Developer Models (Eigenberechtigungen)

### 4.4.10 Rolle: Developer Models (Fremdberechtigungen)

Der Developer Models verfügt über Leserechte und Hinzufügerechte zur Ressource zur Plattform beitragen im Sinne von generativen Modellen sowie zum Lesen der generierten Daten anderer Nutzer. Auf alle anderen Ressourcen hat er keinen Zugriff. Basierend auf die folgenden User Stories (siehe S.76): DMO5, DMO6 und DMO7.

| OBJEKTE / SUBJEKTE | Developer Models (Fremdberechtigungen) |
|---|---|
| User Management | Stufe 0 |
| Role Management | Stufe 0 |
| Permission Management | Stufe 0 |
| Policies und Rules Management | Stufe 0 |
| Support Tickets Management | Stufe 0 |
| Personal Data Management | Stufe 0 |
| Contribution - Extend Platform Management (Generative Models) | Stufe 2 |
| Contribution - Extend Platform Management (Evaluation Metrics) | Stufe 0 |
| Contribution - Extend Platform Management (Data Sources) | Stufe 0 |
| Generate/Evaluate/Analyze Data Management | Stufe 1 |

Abbildung 4.29: Berechtigungsmatrix - Funktions- und Inhaltsressourcen: Developer Models (Fremdberechtigungen)

### 4.4.11 Rolle: Developer Metrics (Eigenberechtigungen)

Der Developer für Evaluierungsmetriken hat selbst vollen Zugriff auf persönliche Daten, die Integration neuer Evaluierungsmetriken und die Anpassung persönlicher Daten. Er besitzt Anpassungs- und Löschrechte für die Erstellung von Support-Tickets. In Bezug auf alle anderen Ressourcen verfügt er lediglich über Leserechte. Basierend auf die folgenden User Stories (siehe S.76): DME1, DME2, DME3 und DME4.

| OBJEKTE / SUBJEKTE | Developer Metrics (Eigenberechtigungen) |
|---|---|
| User Management | Stufe 1 |
| Role Management | Stufe 1 |
| Permission Management | Stufe 1 |
| Policies und Rules Management | Stufe 1 |
| Support Tickets Management | Stufe 3 |
| Personal Data Management | Stufe 4 |
| Contribution - Extend Platform Management (Generative Models) | Stufe 0 |
| Contribution - Extend Platform Management (Evaluation Metrics) | Stufe 4 |
| Contribution - Extend Platform Management (Data Sources) | Stufe 0 |
| Generate/Evaluate/Analyze Data Management | Stufe 4 |

Abbildung 4.30: Berechtigungsmatrix - Funktions- und Inhaltsressourcen: Developer Metrics (Eigenberechtigungen)

### 4.4.12 Rolle: Developer Metrics (Fremdberechtigungen)

Der Developer Metrics hat Leserechte und Hinzufügerechte zur Ressource zur Plattform beitragen im Kontext von Evaluierungsmetriken sowie zum Lesen der generierten Daten anderer Nutzer. Auf alle anderen Ressourcen hat er keinen Zugriff. Basierend auf die folgenden User Stories (siehe S.76): DME5, DME6 und DME7.

| OBJEKTE / SUBJEKTE | Developer Metrics (Fremdberechtigungen) |
|---|---|
| User Management | Stufe 0 |
| Role Management | Stufe 0 |
| Permission Management | Stufe 0 |
| Policies und Rules Management | Stufe 0 |
| Support Tickets Management | Stufe 0 |
| Personal Data Management | Stufe 0 |
| Contribution - Extend Platform Management (Generative Models) | Stufe 0 |
| Contribution - Extend Platform Management (Evaluation Metrics) | Stufe 2 |
| Contribution - Extend Platform Management (Data Sources) | Stufe 0 |
| Generate/Evaluate/Analyze Data Management | Stufe 1 |

Abbildung 4.31: Berechtigungsmatrix - Funktions- und Inhaltsressourcen: Developer Metrics (Fremdberechtigungen)

### 4.4.13 Rolle: Data Engineer (Eigenberechtigungen)

Der Data Engineer hat selbst vollen Zugriff auf persönliche Daten, die Integration neuer Datenquellen/Datenbanken und die Anpassung persönlicher Daten im Zusammenhang mit neuen Datenbankanbindungen. Er besitzt Änderungs- und Löschrechte für die Erstellung von Support-Tickets. In Bezug auf alle anderen Ressourcen verfügt er lediglich über Leserechte. Basierend auf die folgenden User Stories (siehe S.77): DE1, DE2, DE3 und DE4.

| OBJEKTE / SUBJEKTE | Data Engineer (Eigenberechtigungen) |
|---|---|
| User Management | Stufe 1 |
| Role Management | Stufe 1 |
| Permission Management | Stufe 1 |
| Policies und Rules Management | Stufe 1 |
| Support Tickets Management | Stufe 3 |
| Personal Data Management | Stufe 4 |
| Contribution - Extend Platform Management (Generative Models) | Stufe 0 |
| Contribution - Extend Platform Management (Evaluation Metrics) | Stufe 0 |
| Contribution - Extend Platform Management (Data Sources) | Stufe 4 |
| Generate/Evaluate/Analyze Data Management | Stufe 4 |

Abbildung 4.32: Berechtigungsmatrix - Funktions- und Inhaltsressourcen: Data Engineer (Eigenberechtigungen)

### 4.4.14 Rolle: Data Engineer (Fremdberechtigungen)

Der Data Engineer hat Leserechte und Hinzufügerechte, um der Plattform Ressourcen hinzuzufügen, im Kontext von Datenbankanbindungen sowie zum Lesen der generierten Daten anderer Nutzer. Auf alle anderen Ressourcen hat er keinen Zugriff. Basierend auf die folgenden User Stories (siehe S.77): DE5, DE6 und DE7.

| OBJEKTE / SUBJEKTE | Data Engineer (Fremdberechtigungen) |
|---|---|
| User Management | Stufe 0 |
| Role Management | Stufe 0 |
| Permission Management | Stufe 0 |
| Policies und Rules Management | Stufe 0 |
| Support Tickets Management | Stufe 0 |
| Personal Data Management | Stufe 0 |
| Contribution - Extend Platform Management (Generative Models) | Stufe 0 |
| Contribution - Extend Platform Management (Evaluation Metrics) | Stufe 0 |
| Contribution - Extend Platform Management (Data Sources) | Stufe 2 |
| Generate/Evaluate/Analyze Data Management | Stufe 1 |

Abbildung 4.33: Berechtigungsmatrix - Funktions- und Inhaltsressourcen: Data Engineer (Fremdberechtigungen)

### 4.4.15 Rolle: Data Scientist (Eigenberechtigungen)

Der Data Scientist hat selbst vollen Zugriff auf persönliche Daten und die Generierung synthetischer Daten im Kontext von Smart Cities. Er besitzt Änderungs- und Löschrechte für die Erstellung von Support-Tickets. In Bezug auf alle anderen Ressourcen hat er lediglich Leserechte, um Informationen einzusehen. Basierend auf die folgenden User Stories (siehe S.78): DS1, DS2, DS3 und DS4.

| OBJEKTE / SUBJEKTE | Data Scientist (Eigenberechtigungen) |
|---|---|
| User Management | Stufe 1 |
| Role Management | Stufe 1 |
| Permission Management | Stufe 1 |
| Policies und Rules Management | Stufe 1 |
| Support Tickets Management | Stufe 3 |
| Personal Data Management | Stufe 4 |
| Contribution - Extend Platform Management (Generative Models) | Stufe 0 |
| Contribution - Extend Platform Management (Evaluation Metrics) | Stufe 0 |
| Contribution - Extend Platform Management (Data Sources) | Stufe 0 |
| Generate/Evaluate/Analyze Data Management | Stufe 4 |

Abbildung 4.34: Berechtigungsmatrix - Funktions- und Inhaltsressourcen: Data Scientist (Eigenberechtigungen)

### 4.4.16 Rolle: Data Scientist (Fremdberechtigungen)

Der Data Scientist besitzt lediglich Leserechte und Hinzufügerechte auf der Ressource Datengenerierung anderer Nutzer. Auf alle übrigen Ressourcen hat er keinen Zugriff. Basierend auf die folgenden User Stories (siehe S.78): DS5 und DS6.

| OBJEKTE / SUBJEKTE | Data Scientist (Fremdberechtigungen) |
|---|---|
| User Management | Stufe 0 |
| Role Management | Stufe 0 |
| Permission Management | Stufe 0 |
| Policies und Rules Management | Stufe 0 |
| Support Tickets Management | Stufe 0 |
| Personal Data Management | Stufe 0 |
| Contribution - Extend Platform Management (Generative Models) | Stufe 0 |
| Contribution - Extend Platform Management (Evaluation Metrics) | Stufe 0 |
| Contribution - Extend Platform Management (Data Sources) | Stufe 0 |
| Generate/Evaluate/Analyze Data Management | Stufe 2 |

Abbildung 4.35: Berechtigungsmatrix - Funktions- und Inhaltsressourcen: Data Scientist (Fremdberechtigungen)

# 5 Fazit und Ausblick

In diesem Kapitel wird die Schlussfolgerung der Arbeit behandelt.

## 5.1 Fazit

Die im Rahmen dieser Arbeit entwickelte Lösung für das Konzept eines rollenbasierten Berechtigungskonzepts ermöglicht die Implementierung der Plattform gemäß den Anforderungen. Bei der Entscheidungsfindung wurden die Plattformanforderungen berücksichtigt. Dennoch ist es empfehlenswert, während des Implementierungsprozesses der Plattform und der damit verbundenen Entwicklungszyklen regelmäßige Überprüfungen vorzunehmen, um sicherzustellen, dass die vorgeschlagene Lösung auch für die weiterentwickelte Architektur und eventuell veränderte Anforderungen adäquat ist.

Obgleich die in dieser Arbeit präsentierte Lösung für das berechtigungsbasierte Konzept ausgewählt wurde, existieren darüber hinaus weitere geeignete Lösungsansätze im Bereich des Berechtigungskonzepts für diese Plattform. Die in dieser Arbeit präsentierte Lösung stellt lediglich eine von mehreren möglichen Alternativen dar. Vor der langfristigen Integration der vorgeschlagenen Lösung in die Plattform sollten in weiteren Arbeiten durch Erprobungen die praktische Eignung und Überzeugungskraft der präsentierten Lösung überprüft werden. Bislang wurde in dieser Arbeit lediglich die theoretische Konzeption erörtert. Eine praktische Validierung könnte die Effektivität und Anwendbarkeit weiter untermauern.

## 5.2 Rückblick

Die vorliegende Arbeit hatte zum Ziel, ein rollenbasiertes Berechtigungsmodell für die DaFne-Plattform zu entwerfen, welches die Anforderungen erfüllt, die im Rahmen des

Requirements Engineering ermittelt wurden. Das Ziel ist die Schaffung eines Berechtigungsmodells, welches die Plattform in die Lage versetzt, den Zugriff auf Ressourcen und Funktionen zu regulieren. Das Modell soll Aufschluss darüber geben, welche Benutzer in welchen Anwendungsszenarien Zugriff auf welche Daten und Funktionen haben. Des Weiteren soll es dazu beitragen, unterschiedliche Benutzer innerhalb derselben Rolle voneinander abzugrenzen, um die Datensicherheit zu gewährleisten und zu schützen. Des Weiteren zielt das Modell darauf ab, die Sicherheit des Systems oder der Plattform, insbesondere die Verfügbarkeit, zu erhöhen und zu optimieren.

Das Resultat dieser Arbeit ist das entworfene rollenbasierte Berechtigungskonzept, welches in Kapitel 4 detailliert beschrieben wurde. Im Rahmen der Anforderungsanalyse konnte festgestellt werden, dass die ermittelten Anforderungen des Requirements Engineering-Prozesses durch die gewählte Auswahl erfüllt werden. Zudem stellt die gewählte Option eine passende Option für die DaFne-Plattform dar, wobei der aktuelle Entwicklungsstand berücksichtigt wurde.

Im Verlauf dieser Arbeit wurden nach einer Einführung und Zielsetzung im ersten Kapitel im zweiten Kapitel theoretische Grundlagen vermittelt, um eine grundlegende Wissensbasis zu schaffen. Im dritten Kapitel wurde eine Analysephase mittels Requirements Engineering durchgeführt, wobei bewährte Anforderungsanalysetechniken zum Einsatz kamen. Des Weiteren wurde dort definiert, welche Services miteinander kommunizieren müssen, um die Funktionalität der Plattform zu präsentieren. In Kapitel 4 wurde schließlich ein geeignetes Berechtigungsmodell ausgewählt, das durch vorhandene Literatur gestützt wird.

Zusammenfassend lässt sich festhalten, dass die Arbeit erfolgreich die Entwicklung eines rollenbasierten Berechtigungskonzepts vorantreibt, um die Plattform funktional und sicher für die Benutzer zu gestalten.

## 5.3 Ausblick

Nach Abschluss dieser Arbeit ergeben sich verschiedene vielversprechende Möglichkeiten zur Weiterentwicklung des Berechtigungskonzepts. Ein erster Schritt könnte in der praktischen Anwendung des rollenbasierten Berechtigungskonzepts im Kontext von DaFne

bestehen. Die Implementierung in der Plattform sowie die vielfältige Erprobung verschiedener Szenarien könnten einen nützlichen Schritt darstellen, um die Leistungsfähigkeit und Anwendbarkeit des Konzepts in der Praxis zu testen bzw. validieren.

Ein weiterer wesentlicher Aspekt, der in künftigen Forschungsarbeiten Berücksichtigung finden sollte, betrifft die Sicherheit, insbesondere in Bezug auf Datenschutz, Compliance sowie die Einhaltung und Erstellung von Regeln und Richtlinien. Dieser Bereich ist von entscheidender Bedeutung und stellt eine zentrale Herausforderung dar, deren sorgfältige Bearbeitung erforderlich ist, um die Integrität und Sicherheit der Plattform zu gewährleisten.

Des Weiteren wird empfohlen, im Rahmen zukünftiger Forschung und Entwicklung das Verfahren des Role Mining zu nutzen, um potenziell neue Rollen und Berechtigungen zu identifizieren und hinzuzufügen. Dies könnte dazu beitragen, die Effizienz und Relevanz des Berechtigungskonzepts weiter zu optimieren und an die dynamischen Anforderungen anzupassen.

Abschließend wäre noch interessanter Ansatz vorhanden, die Analyse und Verbesserung des Prozesses der Rollenvalidierung mithilfe von Process Mining durchzuführen. Die intelligente Einbindung von KI-gestützten Technologien könnte eine automatisierte und effiziente Validierung der Rollen ermöglichen, was zu einer gesteigerten Effizienz und Genauigkeit des Berechtigungssystems führen könnte.

# Literaturverzeichnis


[1] *Open source identity and access management.* Keycloak. – URL https://www.keycloak.org/

[2] *User Flow.* ProductPlan. – URL https://www.productplan.com/glossary/user-flow/

[3] *Was ist das Least Privilege-Prinzip?* Tools4Ever. – URL https://www.tools4ever.de/glossar/was-ist-das-least-privilege-prinzip/

[4] *Was ist rollenbasierte Zugriffskontrolle?* Entrust. – URL https://www.entrust.com/de/resources/learn/what-is-role-based-access-control

[5] *Zugriffsregeln.* ptc. – URL https://support.ptc.com/help/windchill/r13.0.0.0/de/index.html#page/Windchill_Help_Center/policyadmin/PolicyAdminACLAbout.html

[6] *RBAC vs. ABAC: Definitions and When to Use.* okta, 15.09.2023. – URL https://www.okta.com/identity-101/role-based-access-control-vs-attribute-based-access-control/

[7] *Mandatory Access Control (MAC): Wie funktioniert die regelbasierte Zugriffskontrolle?* Digital Guide IONOS, 24.07.2020. – URL https://www.ionos.de/digitalguide/server/sicherheit/was-ist-mandatory-access-control-mac/

[8] Abts, Dietmar / Mülder W.: *Masterkurs Wirtschaftsinformatik.* Wiesbaden, Deutschland: Vieweg+Teubner Verlag, 2010. – ISBN 9783834800022

[9] Aftab, Z./ Rafiq A.: *Traditional and Hybrid Access Control Models: A Detailed Survey.* ResearchGate, 2022. – URL https://www.researchgate.net/publication/358430862_Traditional_and_Hybrid_Access_Control_Models_A_Detailed_Survey

[10] Baddam, U.: *Evaluating Tabular Data Generation Techniques on the DaFne Platform: Insights from a Predictive Maintenance Case Study on Bridges.* Hamburg, Deutschland: HAW Hamburg, 2023

[11] Beims, Martin: *IT-Service-Management in der Praxis mit ITIL 3: Zielfindung, Methoden, Realisierung.* München, Deutschland: Carl Hanser Verlag GmbH und Co. KG, 2010

[12] BSI, Bundesamt für Sicherheit in der I.: *ORP.4 Identitäts- und Berechtigungsmanagement.* Bundesamt für Sicherheit in der Informationstechnik, Februar, 2021. – URL https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Grundschutz/Kompendium_Einzel_PDFs_2021/02_ORP_Organisation_und_Personal/ORP_4_Identitaets_und_Berechtigungsmanagement_Edition_2021.pdf?__blob=publicationFile&v=2

[13] Bäumler, Helmut: *E-Privacy.* Braunschweig/Wiesbaden, Deutschland: Vieweg + Teubner Verlag, 2000. – URL https://doi.org/10.1007/978-3-322-89183-9. – ISBN 9783528039219

[14] Christian, Beis: *Zugriffskontrollen für WAM-Systeme - Diplomarbeit.* Hamburg, Deutschland: Uni Hamburg, November, 2000. – URL https://swa.informatik.uni-hamburg.de/files/abschlussarbeiten/DiplomarbeitCBeis.pdf

[15] Datenschutz, Aufsichtsstelle: *Leitfaden Berechtigungskonzept.* Basel Landschaft, 11.04.2017. – URL https://www.baselland.ch/politik-und-behorden/besondere-behoerden/datenschutz/publikationen/merkblatter-musterschreiben/downloads-2/Leitfaden%20Berechtigungskonzept%20V%201.0.pdf/@@download/file/Leitfaden%20Berechtigungskonzept%20V%201.0.pdf

[16] Domnick, André / Ebner Fabian / Fox Dirk / Gora Stefan / Hammer Volker / Jendrian Kai/ Knoblauch Hans-Joachim/ Knöppler Michael/ Knopp Michael/ Niederer Sarah/ Pinter Jannis/ Schellhas-Mende Frederike/ Schlichting Jochen/ Völker J.: *Informationssicherheit und Datenschutz.* Heidelberg, Deutschland: dpunkt.verlag GmbH, 2019. – ISBN 9783960885405

[17] Eckert, Claudia: *IT-Sicherheit.* München, Deutschland: de Gruyter Oldenbourg Wissenschaftsverlag GmbH, 2008. – ISBN 9783486582703

[18] Eichholtz, Erik: *Vorstellung Analyzer/Architekturerweiterung - Präsentation.* Hamburg, Deutschland: HAW Hamburg, 21.08.2023

[19] Fox, Dirk / Köhntopp Marit / Pfitzmann A.: *Verlässliche IT-Systeme.* Wiesbaden, Deutschland: Vieweg + Teubner Verlag, 2001. – URL https://doi.org/10.1007/978-3-663-05918-9. – ISBN 9783663059196

[20] Freds, Frieden: *REGELBASIERTE ZUGRIFFSKONTROLLE (RuBAC): Definition und Best Practices.* Businessyield, 02.01.2023. – URL https://businessyield.com/de/technology/rule-based-access-control/

[21] Glass, Ayse/ Tokuc K./ Noenning J.-R./ Steffens U./ Bek B.: *DaFne: Data Fusion Generator and Synthetic Data Generation for Cities.* Hamburg, Deutschland: HAW Hamburg, 2022

[22] Hanscke, Inge: *Informationssicherheit und Datenschutz systematisch und nachhaltig ge-stalten.* Wiesbaden, Deutschland: Springer Vieweg, 2019. – URL https://doi.org/10.1007/978-3-658-27063-6. – ISBN 9783658270629

[23] Herwig, Volker/ Schlabitz L.: *Unternehmensweites Berechtigungsmanagement.* Wirtschaftsinf 46 | S. 289-294, 2004

[24] Hoffmann, Barbara: *Access Control: Models and Methods.* Delinea Blog. – URL https://delinea.com/blog/access-control-models-methods

[25] Howard, Michael / Gantenbein H./ Curzi S.: *Microsoft Azure Security.* Heidelberg, Deutschland: dpunkt.verlag GmbH, 2023. – ISBN 9783864909856

[26] Hühnlein, Detlef: *Identitätsmanagement.* DuD - Datenschutz und Datensicherheit 3|2008 S. 161-163, 2008

[27] Klarl, Heiko: *Zugriffskontrolle in Geschäftsprozessen.* Wiesbaden, Deutschland: Vieweg + Teubner Verlag, 2001. – ISBN 9783834814654

[28] Kleiner, Fritz: *IT Service Management.* Wiesbaden, Deutschland: Springer Vieweg, 2013. – ISBN 9783658001803

[29] Koup, Yev: *Rollenbasierte Zugriffskontrolle (RBAC) vs. Attributbasierte Zugriffskontrolle (ABAC): Was ist der Unterschied?* PingIdentity, 10.02.2022. – URL https://www.pingidentity.com/de/resources/blog/post/rbac-vs-abac.html

[30] Krause, Tom: *An Evaluation of Synthetic Data Generation Tools.* Hamburg, Deutschland: HAW Hamburg, 2022

[31] Kunert, Pamela: *A Platform Providing Machine Learning Algorithms for Data Generation and Fusion - An Architectural Approach - SAML Präsentation.* Hamburg, Deutschland: HAW Hamburg, 2022

[32] Kunert, Pamela: *Tabular Data Synthesis, Master Thesis.* Hamburg, Deutschland: HAW Hamburg, 2023

[33] Kunert, Pamela / Krause Tom / Zukunft Olaf / Steffens U.: *A Platform Providing Machine Learning Algorithms for Data Generati-on and Fusion – An Architectural Approach.* Hamburg, Deutschland: HAW Hamburg, 2022

[34] Köller, Joe: *Arten von Zugriffskontrolle: Alle Modelle erklärt!* Tenfold, 16.01.2024. – URL https://www.tenfold-security.com/wiki/zugriffskontrolle/

[35] Kübra, Tokuc: *A Platform Providing Machine Learning Algorithms for Data Generation and Fusion – A User and Product Perspective.* Hamburg, Deutschland: HAW Hamburg, 2022

[36] Leicht, Michael / Möllers F.: *Zur (Un-)Sicherheit von Zwei-Faktor-Authentifizierung via SMS.* Datenschutz Datensich 45, S. 541–545, 2021

[37] Lochmahr, Andrea / Müller Patrick / Planing Patrick / Popović T.: *Digitalen Wandel gestalten.* Berlin und Heidelberg, Deutschland: Springer Gabler, 2018. – URL https://doi.org/10.1007/978-3-658-24651-8. – ISBN 9783658246501

[38] Lutkevich, Ben: *identity provider.* TechTarget Security, 2024. – URL https://www.techtarget.com/searchsecurity/definition/identity-provider

[39] Menk, Alexander/ Gedigk S.: *Grundprojekt: Die DaFne Plattform,.* Hamburg, Deutschland: HAW Hamburg, 2023

[40] Müller, Klaus-Rainer: *IT-Sicherheit mit System.* Wiesbaden, Deutschland: Springer Vieweg, 2014. – URL https://doi.org/10.1007/978-3-658-04334-6. – ISBN 9783658043346

[41] Petrlic, Ronald/ Sorge C.: *Datenschutz*. Wiesbaden, Deutschland: Springer Vieweg, 2017. – URL https://doi.org/10.1007/978-3-658-16839-1. – ISBN 9783658168384

[42] Rawal, Bharat S./ Manogaran Gunasekaran/ Peter A.: *Cybersecurity and Identity Access Management*. Singapore, Springer Nature, 2023. – URL https://doi.org/10.1007/978-981-19-2658-7. – ISBN 9789811926570

[43] Rouse, Margaret: *Discretionary Access Control*. Techopedia, 05.06.2023. – URL https://www.techopedia.com/de/definition/discretionary-access-control-dac

[44] Shea, Sharon: *Wie können Unternehmen Zugriffsrechte richtig überprüfen?* TechTarget, 2023. – URL https://www.computerweekly.com/de/antwort/Wie-koennen-Unternehmen-Zugriffsrechte-richtig-ueberpruefen

[45] Steffens, Ulrike: *Sachbericht zum Zwischennachweis, DaFne*. Hamburg, Deutschland: HAW Hamburg, 2022

[46] Steffens, Ulrike / Noennig Jörg Rainer / Niederée Claudia / Nagel Wolfgang E. / Schloemer Lars / von den Brincken C.: *Vorhabensbeschreibung DaFne - Projektvorschlag zur Ausschreibung: „Erzeugung von synthetischen Daten für Künstliche Intelligenz“ des BMBF*. Hamburg, Deutschland: HAW Hamburg, 21.06.2016. – URL https://www.bmbf.de/foerderungen/bekanntmachung-3068.html

[47] Sädtler, Stephan: *Rechtskonformes Identitätsmanagement im Cloud Computing*. Wiesbaden, Deutschland: Springer Fachmedien, 2017. – URL https://doi.org/10.1007/978-3-658-14807-2. – ISBN 9783658148065

[48] Thakur, Manav-A./ Gaikwad R.: *User Identity and Access Management Trends in IT Infrastructure- An Overview*. International Conference on Pervasive Computing (ICPC), 2015

[49] Tsolkas, Alexander / Schmidt K.: *Rollen und Berechtigungskonzepte*. Wiesbaden, Deutschland: Springer Vieweg, 2017. – URL https://doi.org/10.1007/978-3-658-17987-8. – ISBN 9783658179861

[50] Urbach, Nils / Ahlemann F.: *IT-Management im Zeitalter der Digitalisierung*. Berlin und Heidelberg, Deutschland: Springer Gabler, 2016. – URL https://doi.org/10.1007/978-3-662-52832-7. – ISBN 9783662528310

# A Anhang

## A.1 Personas

**PERSONA 02: SUPER-ADMIN**

***Name:*** Julian Krüger

***Lebensmotto/Leitsatz:*** 'Technologie gestaltet unsere Zukunft - sei ein Teil dieser Gestaltung'.

***Alter:*** 35

***Wohnort:*** Berlin, Deutschland

***Ausbildung:*** Master in Informatik mit Schwerpunkt Data Science

***Beruf:*** Systemadministrator

***Branche:*** Informationstechnologie

***Spezialgebiet:*** Datenmanagement und IT-Infrastruktur

***Technische Fähigkeiten:***

- Kenntnisse in modernen Datenbanktechnologien und -architekturen

- Fortgeschrittene Kenntnisse in Netzwerksicherheit und -management

- Umfassende Erfahrung mit Cloud Computing Plattformen (AWS, Azure)

- Programmier- und Scripting-Kenntnisse (Python, Bash)

- Erfahrung mit Werkzeugen zur Datengenerierung und -manipulation

- Fähigkeit, große IT-Systeme zu implementieren, zu warten und hochverfügbar zu halten.

***Motivation und Ziele:***

Julian ist motiviert durch die Herausforderung, leistungsfähige IT-Systeme zu entwerfen und zu verwalten, die innovative Forschung und Entwicklung unterstützen. Sein Hauptziel ist der Aufbau einer robusten IT-Infrastruktur, die die Generierung und den sicheren Umgang mit synthetischen Daten erleichtert. Es will dazu beitragen, die Lücke in der Datenverfügbarkeit zu schließen und Forschern und Entwicklern Werkzeuge an die Hand geben, um mit KI-Methoden bahnbrechende Fortschritte zu erzielen. Dabei legt es besonderen Wert auf Datenschutz und Datensicherheit.

***Schwächen:***

- Neigung zur Überforderung, daher manchmal Probleme mit der Work-Life-Balance

- Tendenz, Detailfragen Vorrang einzuräumen, was von strategischen Zielen ablenken kann

- Gelegentliche Schwierigkeiten, Nicht-Technikern die Bedeutung technischer Details zu vermitteln.

***Stärken:***

- Ausgeprägte analytische Fähigkeiten und Problemlösungskompetenz

- Ausgeprägte Fähigkeit, sich schnell in neue Technologien einzuarbeiten und diese in bestehende Systeme zu integrieren

- Hervorragende kommunikative Fähigkeiten, insbesondere in der Vermittlung technischer Konzepte an Nicht-Techniker

- Hohes Engagement für Datenschutz und IT-Sicherheit

- Führungsqualitäten und die Fähigkeit, Teams zu motivieren und zu leiten

eu machen, um Innovation zu fördern.

***Schwächen:***

- Manchmal gestresst durch den Druck einer sich schnell verändernden Technologielandschaft.

- Kann zögerlich sein, wenn es darum geht, Konflikte zu lösen, insbesondere wenn es um technische Diskussionen geht.

- Neigt dazu, Perfektion in der Plattformautomatisierung anzustreben, was zu Verzögerungen bei der Umsetzung führen kann.

***Stärken:***

- Hervorragende organisatorische Fähigkeiten und die Fähigkeit, Prioritäten effektiv zu setzen.

- Schnelle Auffassungsgabe und Fähigkeit zur Lösung komplexer technischer Probleme

- Starker Fokus auf Benutzererfahrung und Benutzerfreundlichkeit

- Begeisterung für Teamarbeit und Zusammenarbeit

- Ständige Weiterbildung im Bereich der Informationstechnologie, um Herausforderungen vorwegzunehmen

**PERSONA 03: PLATTFORM-ADMIN**

***Name:*** Giulia Schmidt

***Lebensmotto/Leitsatz:*** 'In der Einfachheit liegt der Schlüssel zu wahrer Größe'.

***Alter:*** 29

***Wohnort:*** München, Deutschland

***Ausbildung:*** Bachelor in Informationssysteme, Weiterbildung in Cloud-Technologien

***Beruf:*** Plattform-Administratorin

***Branche:*** Informationstechnologie und Dienstleistungen

***Spezialgebiet:*** Cloud Plattform Management, Datenanalyse

***Technische Fähigkeiten:***

- Umfassendes Wissen über Cloud-Servicemodelle (IaaS, PaaS, SaaS) und Anbieter (insbesondere Google Cloud und Azure)

- Erfahrung in der Administration von Servern, Datenbanken und Webservices

- Kenntnisse in automatisierten Deployment-Tools (z.B. Jenkins, Kubernetes)

- Programmierkenntnisse in Python und R

- Erfahrung mit Monitoring-Tools zur Performance- und Sicherheitsbewertung

- Starke Fähigkeiten in Datenvisualisierung und -analyse

***Motivation und Ziele:***

Giulias Ziel ist es, eine zugängliche, effiziente und sichere Plattform zur Datengenerierung zu schaffen und zu pflegen, die Entwickler und Forscher bei ihrer Arbeit unterstützt. Sie möchte die Plattform so gestalten, dass sie flexibel auf die Bedürfnisse verschiedener Nutzergruppen reagieren kann, ohne die Sicherheitsstandards und den Datenschutz zu gefährden. Ihre Motivation ist der Wunsch, Technologie zugänglich und verständlich zu machen, um Innovation zu fördern.

***Schwächen:***

- Manchmal gestresst durch den Druck einer sich schnell verändernden Technologielandschaft.

- Kann zögerlich sein, wenn es darum geht, Konflikte zu lösen, insbesondere wenn es um technische Diskussionen geht.

- Neigt dazu, Perfektion in der Plattformautomatisierung anzustreben, was zu Verzögerungen bei der Umsetzung führen kann.

***Stärken:***

- Hervorragende organisatorische Fähigkeiten und die Fähigkeit, Prioritäten effektiv zu setzen.

- Schnelle Auffassungsgabe und Fähigkeit zur Lösung komplexer technischer Probleme

- Starker Fokus auf Benutzererfahrung und Benutzerfreundlichkeit

- Begeisterung für Teamarbeit und Zusammenarbeit

- Ständige Weiterbildung im Bereich der Informationstechnologie, um Herausforderungen vorwegzunehmen

**PERSONA 04: SUPPORT USER**

***Name:*** Frederik Meyer

***Lebensmotto/Leitsatz:*** 'Jedes Problem ist eine Chance zu lernen'.

***Alter:*** 32

***Wohnort:*** Hamburg, Deutschland

***Ausbildung:*** Fachinformatiker für Systemintegration, Fortbildungen im Bereich Kundensupport und IT Service Management

***Beruf:*** Support-Mitarbeiter

***Branche:*** Informationstechnologie und Dienstleistungen

***Spezialgebiet:*** IT-Support, Kundenbetreuung

***Technische Fähigkeiten:***

- Breites Wissen über Hardware- und Software-Problemlösungen

- Erfahrung im Umgang mit unterschiedlichen Betriebssystemen und Softwareanwendungen

- Kenntnisse in Netzwerktechnik und Fehlerbehebung

- Kenntnisse in Datenbankmanagement und Grundlagen der Programmierung

- Kompetenz im Umgang mit Ticketing-Systemen und Support-Tools

- Fähigkeit, technische Anweisungen und Lösungen klar und verständlich zu kommunizieren

***Motivation und Ziele:***

Frederik ist hochmotiviert, exzellenten Kundensupport zu leisten und Anwendern dabei zu helfen, technische Herausforderungen effektiv zu meistern. Sein Ziel ist es, die Nutzerzufriedenheit kontinuierlich zu steigern und Lösungen anzubieten, die nicht nur kurzfristige Probleme lösen, sondern auch langfristig zur Stabilität und Nutzbarkeit der Plattform beitragen. Es ist bestrebt, sein technisches Wissen kontinuierlich zu erweitern, um auf ein breiteres Spektrum von Anfragen reagieren zu können.

***Schwächen:*** - Kann manchmal von der Menge der gleichzeitig zu bearbeitenden Anfragen überwältigt sein.

- Neigt dazu, zu viel Zeit in einzelne Support-Tickets zu investieren, um sicherzustellen, dass das Problem vollständig gelöst wird.

- Hat gelegentlich Schwierigkeiten, bei sehr technischen oder spezifischen Problemen die Initiative zu ergreifen.

***Stärken:***

- Außergewöhnliche Geduld und Einfühlungsvermögen im Umgang mit Benutzeranfragen

- Fähigkeit, komplexe technische Konzepte in einfache Sprache zu übersetzen

- Ausgeprägte analytische Fähigkeiten zur raschen Identifizierung von Problemursachen

- Selbstmotivation mit einer proaktiven Einstellung zur Problemlösung

- Ausgeprägte Teamfähigkeit und die Bereitschaft, Kollegen zu unterstützen

**PERSONA 05: AUDITOR**

***Name:*** Dr. Marie Schröder

***Lebensmotto/Leitsatz:*** 'Genauigkeit ist der Schlüssel zur Integrität'.

***Alter:*** 40

***Wohnort:*** Frankfurt/Main, Deutschland

***Ausbildung:*** Promotion in Wirtschaftsinformatik mit den Schwerpunkten IT-Revision und Cyber-Security

***Beruf:*** IT-Auditorin

**Branche:** Finanzdienstleistungen und Consulting

***Spezialgebiet:*** IT-Revision, IT-Governance, Risikomanagement

***Technische Fähigkeiten:***

- Fundiertes Verständnis von IT-Governance-Modellen und Audit-Methodologien

- Expertise in der Analyse und Prüfung von IT- und Informationssicherheits-Managementsystemen

- Umfassende Kenntnisse nationaler und internationaler Standards wie ISO/IEC 27001, NIST und ITIL

- Erfahrung in der Bewertung von Data Governance-, Datenintegritäts- und Compliance-Management-Systemen

- Sicherer Umgang mit Tools zur Risikoanalyse und -bewertung

- Kenntnisse in Kryptographie und Zugriffskontrollsystemen

***Motivation und Ziele:***

Dr. Marie Schröder hat sich zum Ziel gesetzt, die Sicherheit, Effizienz und Compliance von IT-Systemen zu gewährleisten. Ihre Motivation liegt in der Verbesserung interner Kontrollen und der Erhöhung der Transparenz in IT-Umgebungen. Sie möchte Organisationen dabei unterstützen, ihre Cyber-Resilienz zu stärken und sicherzustellen, dass die Umsetzung des Berechtigungskonzepts die definierten Sicherheits- und Compliance-Anforderungen erfüllt.

***Schwächen:***

- Neigt zu Perfektionismus, was bei hoher Arbeitsbelastung zu Stress führen kann.

- Kann manchmal überkritisch sein, wenn es um die Einhaltung von Richtlinien und Standards geht.

- Starker Fokus auf Details kann gelegentlich den Blick für das Ganze verstellen

***Stärken:***

- Ausgezeichnete analytische Fähigkeiten und Sinn für Details

- Starke ethische Grundsätze und hohe Professionalität

- Fähigkeit, komplexe technische Sachverhalte verständlich darzustellen

- Ausgezeichnete zwischenmenschliche und kommunikative Fähigkeiten, die eine effektive Zusammenarbeit mit allen Ebenen der Organisation ermöglichen.

- Proaktive Herangehensweise an die Identifizierung und Lösung von Sicherheitsrisiken

- Ständige Weiterbildung und berufliche Entwicklung, um auf dem neuesten Stand der Technik und der Best Practices zu bleiben.

**PERSONA 06: DEVELOPER METRICS**

***Name:*** Fabian Gruber

***Lebensmotto/Leitsatz:*** 'Messbarer Fortschritt ist die Basis für Erfolg'.

***Alter:*** 34

***Wohnort:*** Nürnberg, Deutschland

***Ausbildung:*** Master in Statistik und Data Science

***Beruf:*** Entwickler für Evaluationsmetriken

***Branche:*** Softwareentwicklung und Analytik

***Spezialgebiet:*** Datenanalyse, Evaluierungsmetriken, Maschinelles Lernen

***Technische Fähigkeiten:***

- Erfahrung in statistischer Modellierung und Analyse

- Solide Programmierkenntnisse in Python und Erfahrung mit R für statistische Berechnungen

- Kenntnisse in der Entwicklung und Implementierung von Evaluierungsmetriken für maschinelles Lernen und KI-Modelle

- Erfahrung mit maschinellen Lernframeworks wie Scikit-Learn zur Evaluierung von Modellen

- Erfahrung im Umgang mit Big Data Technologien und Werkzeugen zur Datenverarbeitung

- Fähigkeit zur kritischen Analyse der Modellperformance und Identifizierung von Verbesserungspotenzialen

***Motivation und Ziele:***

Fabian ist motiviert durch die Überzeugung, dass der Schlüssel zur Verbesserung von KI-Modellen in präzisen Evaluierungsmethoden liegt. Sein Hauptziel ist es, innovative Metriken zu entwickeln, die über traditionelle Metriken hinausgehen und Entwicklern einen tieferen Einblick in die Leistung ihrer Modelle geben. Ziel ist es, den Entwicklungsprozess durch genaue Messungen effektiver und effizienter zu gestalten.

***Schwächen:***

- Kann sich in Details verstricken, was manchmal den Fortschritt verlangsamt.

- Neigt dazu, bestehende Evaluierungsmethoden kritisch zu hinterfragen, was manchmal zu Konflikten im Team führen kann.

- Manchmal zu vorsichtig bei der Implementierung neuer Metriken aus Angst, bestehende Prozesse zu stören.

***Stärken:*** - Ausgeprägte analytische Fähigkeiten und Liebe zum Detail

- Fähigkeit, komplexe Probleme zu erkennen und durch innovative Ansätze zu lösen

- Hohes Verständnis für die Bedeutung von Qualität und Genauigkeit bei der Bewertung von KI-Modellen

- Starkes Engagement für Transparenz und Nachvollziehbarkeit in der Modellbewertung

- Teamplayer mit der Fähigkeit, Wissen und Best Practices zu teilen

**PERSONA 07: DATA ENGINEER**

***Name:*** Nick Ohrmann

***Lebensmotto/Leitsatz:*** 'Daten sind das neue Öl - aber Wissen ist die Raffinerie'.

***Alter:*** 33

***Wohnort:*** Stuttgart, Deutschland

***Ausbildung:*** Master in Data Science

***Beruf:*** Dateningenieur

***Branche:*** Technologie und Datenanalyse

***Spezialgebiet:*** Data Engineering, Big Data Technologien

***Technische Fähigkeiten:***

- Expertise im Hadoop-Ökosystem, Spark und Kafka für die Verarbeitung großer Datenmengen

- Erfahrung im Umgang mit relationalen Datenbanken (SQL) und NoSQL Datenbanken (MongoDB, Cassandra)

- Erfahrung in der Entwicklung und Optimierung von ETL-Prozessen (Extract, Transform, Load)

- Kenntnisse in Programmiersprachen wie Python und Java, insbesondere für die Datenverarbeitung und -analyse

- Erfahrung mit Cloud-Diensten (AWS, Google Cloud Platform) für das Datenmanagement

- Fähigkeit, komplexe Datenpipelines und Datenarchitekturen zu entwerfen und zu implementieren

***Motivation und Ziele:***

Nick ist leidenschaftlich daran interessiert, Unternehmen dabei zu unterstützen, den Wert ihrer Daten voll auszuschöpfen. Er ist bestrebt, effiziente Dateninfrastrukturen aufzubauen, die robuste Datenanalysen und -einblicke ermöglichen. Sein Ziel ist es, die Datenqualität kontinuierlich zu verbessern und innovative Datenlösungen zu entwickeln, die Geschäftsentscheidungen unterstützen und vorantreiben.

***Schwächen:***

- Neigt dazu, sich zu sehr in technische Details zu vertiefen, was sich gelegentlich auf den Projektzeitplan auswirken kann.

- Hat manchmal Schwierigkeiten, sich von etablierten Lösungen zu lösen und neue Ansätze auszuprobieren.

- Kann unter Druck die Prioritäten zwischen verschiedenen Projekten verlieren.

***Stärken:***

- Ausgezeichnete Fähigkeit, komplexe Datensätze zu analysieren und daraus handlungsorientierte Schlussfolgerungen zu ziehen.

- Starke Problemlösungskompetenz und Fähigkeit zur Optimierung von Datenverarbeitungsprozessen

- Fähigkeit, technisch komplexe Konzepte nicht-technischen Stakeholdern verständlich zu erklären

- Proaktiver Teamplayer, der eng mit anderen Technikern sowie mit Geschäfts- und Analyseteams zusammenarbeitet

- Hält sich über die neuesten Datentechnologien und Trends auf dem Laufenden und ist dadurch in der Lage, innovative Lösungen vorzuschlagen und umzusetzen.

**PERSONA 08: DATA SCIENTIST - BEISPIEL 1**

***Name:*** Christine Herrmann

***Lebensmotto/Leitsatz:*** 'Hinter jedem Datensatz steckt eine Geschichte, die es zu entschlüsseln gilt'.

***Alter:*** 30

***Wohnort:*** Leipzig, Deutschland

***Ausbildung:*** Promotion in Statistik mit Schwerpunkt maschinelles Lernen

***Beruf:*** Datenwissenschaftler

***Branche:*** E-Commerce und Online-Marketing

***Spezialgebiet:*** Datenanalyse, Predictive Modelling, Maschinelles Lernen

***Technische Fähigkeiten:***

- Fortgeschrittene Kenntnisse in statistischen Analysemethoden und maschinellem Lernen

- Beherrschung von Programmiersprachen wie Python und R zur Datenanalyse und Modellierung

- Erfahrung mit Machine Learning Frameworks (z.B. TensorFlow, SciKit Learn) und Deep Learning

- Vertraut mit Datenvisualisierungstools wie Tableau und Power BI

- Erfahrung mit Big-Data-Technologien wie Apache Spark

- Erfahrung in der Anwendung von A/B-Tests und anderen statistischen Testverfahren zur Hypothesenprüfung

***Motivation und Ziele:***

Christine ist von der Herausforderung motiviert, komplexe Daten zu analysieren und daraus wertvolle Erkenntnisse zu gewinnen, die Geschäftsentscheidungen unterstützen können. Ihr Ziel ist es, durch präzise Datenmodelle und innovative Analysemethoden

einen direkten Beitrag zum Unternehmenserfolg zu leisten. Sie strebt danach, datengetriebene Lösungen zu entwickeln, die das Kundenerlebnis verbessern und den Umsatz steigern.

***Schwächen:***

- Manchmal zu sehr auf die Perfektionierung von Modellen fokussiert, was zu längeren Entwicklungszeiten führen kann.

- Könnte klarer in der Kommunikation ihrer Erkenntnisse sein, insbesondere gegenüber Stakeholdern ohne technischen Hintergrund.

- Neigt dazu, unter Zeitdruck den Überblick über Projekttermine zu verlieren.

***Stärken:***

- Ausgezeichnete analytische Fähigkeiten und die Fähigkeit, komplexe Muster und Trends in Daten zu erkennen

- Kreativ bei der Entwicklung datengestützter Lösungen für Geschäftsprobleme

- Engagiert bei der ständigen Weiterbildung und beim Einsatz neuester Technologien und Methoden im Bereich des maschinellen Lernens

- Starke Teamplayerin, die eng mit Ingenieuren, Produktmanagern und Marketingteams zusammenarbeitet, um gemeinsame Ziele zu erreichen

- Hat die Gabe, datengestützte Geschichten zu erzählen, die überzeugen und zum Handeln anregen

**PERSONA 09: DATA SCIENTIST - BEISPIEL 2**

***Name:*** Dr. Ralf Otto

***Lebensmotto/Leitsatz:*** 'Smart Cities sind nicht nur intelligent, sie sind empathisch.'

***Alter:*** 45

***Wohnort:*** Aachen, Deutschland

***Ausbildung:*** Promotion in Stadtplanung und Smart City Technologien

***Beruf:*** Smart City Berater

***Branche:*** Öffentliche Verwaltung und Stadtentwicklung

***Spezialgebiet:*** Stadtplanung, Nachhaltige Entwicklung, Smart City Lösungen

***Technische Fähigkeiten:***

- Tiefes Verständnis von IoT (Internet of Things) Technologien und deren Anwendung in Smart Cities

- Kenntnisse in Datenanalyse und -management speziell für urbane Daten

- Erfahrung in der Entwicklung und Umsetzung von Smart City Projekten

- Erfahrung in der Anwendung von GIS (Geographische Informationssysteme) für Stadtplanungsaufgaben

- Kenntnisse in grüner Infrastruktur und nachhaltiger Mobilität

***Motivation und Ziele:***

Dr. Ralf ist leidenschaftlich daran interessiert, Städte durch den Einsatz intelligenter Technologien nachhaltiger, effizienter und lebenswerter zu gestalten. Sein Hauptziel ist es, zum globalen Wandel hin zu intelligenten urbanen Ökosystemen beizutragen, die die Lebensqualität verbessern und gleichzeitig die Umweltbelastung minimieren. Es ist bestrebt, innovative Smart-City-Projekte zu initiieren und zu unterstützen, die auf den Prinzipien der Nachhaltigkeit und der Bürgerbeteiligung basieren.

***Schwächen:***

- Kann in seinen Visionen für Smart Cities zu idealistisch sein, was manchmal zu Enttäuschungen führt, wenn Projekte aufgrund von Budget- oder Technologiebeschränkungen angepasst werden müssen.

- Neigt dazu, den technologischen Aspekt über die sozialen Auswirkungen zu stellen.

- manchmal ungeduldig mit dem langsamen Tempo bürokratischer Entscheidungsprozesse

***Stärken:***

- Ausgezeichnete Fähigkeit, komplexe städtische Herausforderungen zu analysieren und innovative Lösungen vorzuschlagen.

- Starke Überzeugungskraft, um Stakeholder und Politiker für Smart-City-Initiativen zu gewinnen

- Hohe Kompetenz in der Führung interdisziplinärer Teams

- Ausgeprägte Fähigkeit zur Strategieentwicklung und zum Projektmanagement

- Weitblick und die Fähigkeit, zukünftige Trends in der Stadtentwicklung zu antizipieren

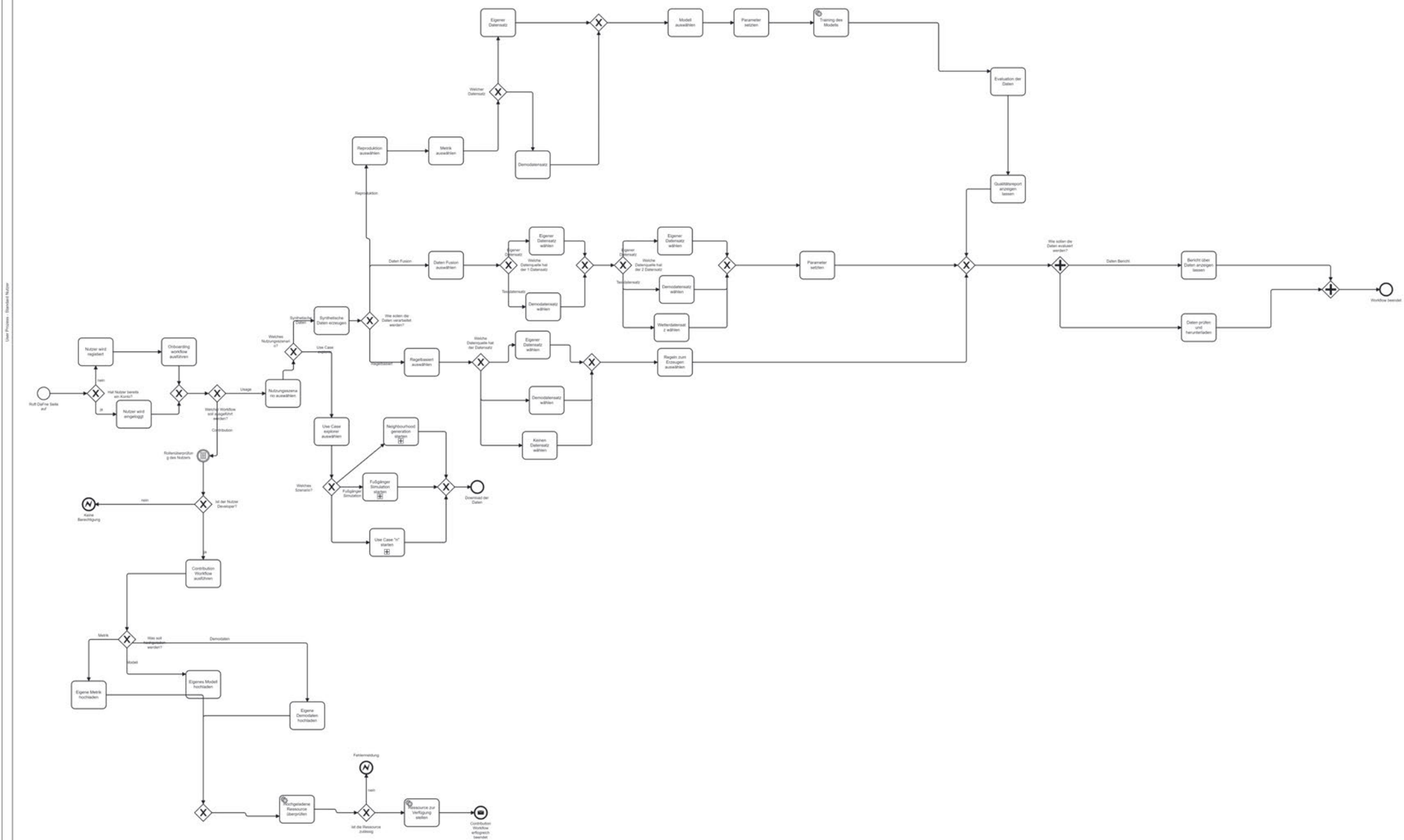
## A.2 Benutzerprozess des Standardbenutzers auf DaFne

Der hier beschriebene Benutzerprozess wurde von der Masterstudentin der Informatik an der HAW Hamburg, Julia Seufert, für einen Standardbenutzer auf der DaFne-Plattform modelliert.

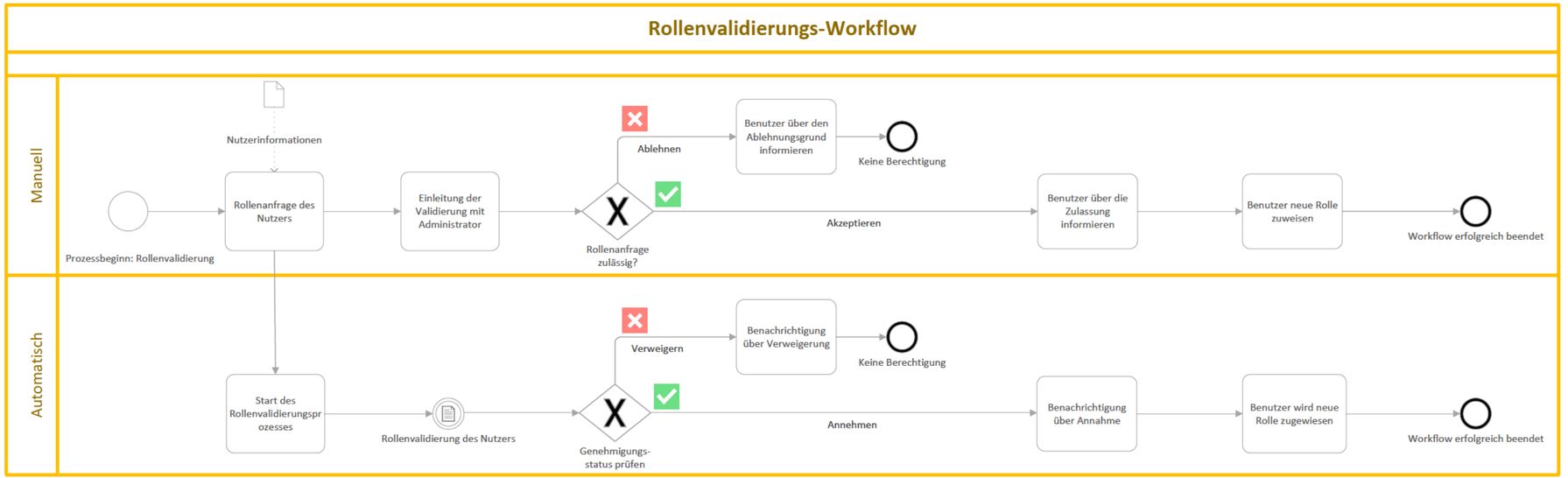
## A.3 Workflow vom Validierungsprozess von Benutzerrollen

Der folgende Workflow veranschaulicht den Validierungsprozess von Benutzerrollen.

# A.2 Benutzerprozess des Standardbenutzers auf DaFne



# A.3 Workflow vom Validierungsprozess von Benutzerrollen



## Erklärung zur selbständigen Bearbeitung

Hiermit versichere ich, dass ich die vorliegende Arbeit ohne fremde Hilfe selbständig verfasst und nur die angegebenen Hilfsmittel benutzt habe. Wörtlich oder dem Sinn nach aus anderen Werken entnommene Stellen sind unter Angabe der Quellen kenntlich gemacht.

| Stade | 27.05.2024 | [redacted] |
|---|---|---|
| Ort | Datum | Unterschrift im Original |