



Entwicklung und Implementierung eines Modells zur Erfassung und Bewertung der User Experience am Beispiel der Anwendung eDok des LWV Hessen

Ali Alkhiami

Bachelorarbeit am Fachbereich AI der HAW Fulda

Matrikelnummer: 844620

Erstbegutachtung: Prof. Dr. Alexander Gepperth

Zweitbegutachtung: Prof. Dr. Dr. YYY

Eingereicht am 05.mm.yyyy

0.1 Abstract

Diese Arbeit befasst sich mit der Entwicklung und Implementierung eines Modells zur umfassenden Erfassung und Bewertung der User Experience (UX), exemplarisch angewendet auf die Anwendung eDok des Landeswohlfahrtsverbands Hessen (LWV Hessen). Ziel des Modells ist es, einen detaillierten Überblick über die UX innerhalb der Anwendung zu schaffen, indem während der produktiven Nutzung anonymisierte Daten gesammelt werden.

Die gesammelten Daten sind bewusst nicht benutzerorientiert, um Datenschutzrichtlinien einzuhalten und die Privatsphäre der Nutzer zu schützen. Das Modell präsentiert diese aggregierten Daten in einem Dashboard, das ausschließlich für Nutzer mit Administratorrechten zugänglich ist. Dieses Dashboard bietet anpassbare Grafiken und Visualisierungen, die durch Filter und weitere Kriterien modifiziert werden können. Dadurch erhalten Administratoren nicht nur Einblicke in die aktuelle UX, sondern können auch Trends und Entwicklungen über einen bestimmten Zeitraum analysieren.

Ein wesentlicher Aspekt des Modells ist die Möglichkeit für Projektverantwortliche, die Datenerfassung auf bestimmten Seiten der Anwendung gezielt an- oder auszuschalten. Dies stellt sicher, dass nur relevante Bereiche beobachtet werden, ohne die Performance auf anderen Seiten zu beeinträchtigen. Das Modell ermöglicht es, die Verweildauer der Nutzer auf spezifischen Seiten zu überwachen, häufig durchlaufene Navigationsmuster zu identifizieren und die Häufigkeit bestimmter Fehler zu erfassen, die von den Nutzern gemacht werden.

Die gewonnenen Erkenntnisse unterstützen die Projektverantwortlichen dabei, problematische Stellen innerhalb der Anwendung zu identifizieren und gezielte Maßnahmen zur Verbesserung der Usability zu ergreifen. Darüber hinaus ermöglicht das Modell eine nachträgliche Überprüfung der implementierten Änderungen, um deren Effektivität zu bewerten und weitere Optimierungen vorzunehmen. Durch diesen iterativen Prozess trägt das Modell wesentlich zur Steigerung der Benutzerzufriedenheit und Effizienz der Anwendung bei.

Inhaltsverzeichnis

0.1 Abstract	1
Inhaltsverzeichnis	1
1 Kapitel 1	3
Abkürzungsverzeichnis	3
1.1 Einleitung	3
1.2 Über den Landeswohlfahrtsverband Hessen	4
1.3 Die ANLEI-Service GmbH	4
1.4 Über eDok	4
1.5 Motivation	5

1.6	Ziel der Arbeit	6
1.7	Fragestellung	6
1.8	Eigenleistung	6
1.9	Grundlagen und Methodik	6
1.9.1	Architektur und Softwarekomponenten	7
1.9.2	Datenbankmodell	8
1.9.3	Methodisches Vorgehen bei der Datenerfassung	8
1.9.4	SQL-Datenbankmodell	9
1.9.5	Zusammenfassung	10
1.10	Relevanz der Arbeit	10
1.11	Datenschutz und DSGVO	11
1.12	Performance-Überlegungen	11
1.13	Fazit	11
2	Kapitel 2	12
2.1	Einführung in das Thema UX	12
2.2	Herausforderungen und Offenheit der UX-Definition	12
2.3	Forschungsperspektiven: UX als vielschichtiges Phänomen	12
2.4	Zusammenfassung der UX-Definitionen	13
2.5	Usability und ihre Bedeutung	13
2.6	Usability in eDok	14
2.7	Erweiterte Ziele der UX- und Usability-Bewertung in eDok	14
2.7.1	Abgrenzung von UX- und Usability-Evaluationsmethoden	15
2.7.2	Fokussierung auf Usability in dieser Arbeit	15
2.7.3	Vergleich von Nutzerinterviews, Fragebögen und der Sammlung quantitativer Daten	17
2.7.4	Nutzerinterviews und Fragebögen	18
2.8	Sammlung und Analyse quantitativer Daten	18
2.9	Erkenntnisse und Schlussfolgerung	19
2.10	Methodik	19
2.10.1	Systemarchitektur und Datenerfassung	19
2.10.2	Datenpersistenz und Datenbankmodell	20
2.10.3	Backend-Logik und REST-Schnittstellen	20
2.10.4	Anonymisierung und Datenschutz	21
2.10.5	Visualisierung und Heatmap-Darstellung	21
2.10.6	Iterative Verbesserung und Anpassung	21
3	Kapitel 3	21
3.1	Anforderungen	21
3.2	Spezifikation der Erfassten UX-Daten in eDok	22
3.3	Implementierung der Heatmap	22
3.4	Zweck und Ziele der Heatmap	22
3.5	Datenerfassung und Speicherung	22

3.6	Implementierung Strategie	23
3.7	Global Heatmap Service	23
3.8	Overlay Directive	23
3.9	Admin Toggle Control	23
3.10	Beispiele für Auswertungen	23
3.11	Bezug zu den Zielen (Kapitel 1)	26

1 Kapitel 1

UX	User Experience
LWV	Landeswohlfahrtsverband
SUS	System Usability Scale
UEQ	User Experience Questionnaire
UI	User Interface
eDok	elektronische Dokumentenerstellung
HCI	Humin Computer Interaction

1.1 Einleitung

Die **Mensch-Computer-Interaktion** (Human-Computer Interaction, HCI) ist ein interdisziplinäres Forschungsfeld, das sich mit der Gestaltung, Evaluation und Implementierung interaktiver Computersysteme für die menschliche Nutzung sowie mit der Untersuchung der damit verbundenen Phänomene beschäftigt. Innerhalb dieses Feldes haben sich zwei zentrale Ansätze zur **User Experience** (UX) herausgebildet: das **UX-Design** und das **nutzerzentrierte Design** (User-Centered Design) [4].

Das **UX-Design** strebt danach, ein ganzheitliches Nutzererlebnis zu schaffen, das über die reine Funktionalität eines Produkts hinausgeht. Es berücksichtigt emotionale, psychologische und ästhetische Aspekte, um Produkte zu entwickeln, die nicht nur effizient, sondern auch angenehm und zufriedenstellend in der Nutzung sind. Ziel ist es, ein positives Gesamterlebnis zu gestalten, das die Bedürfnisse und Erwartungen der Nutzer übertrifft.

Im Gegensatz dazu fokussiert das **nutzerzentrierte Design** auf die Einbindung der Endnutzer während des gesamten Entwicklungsprozesses. Durch aktive Einbindung von Nutzerfeedback wird sichergestellt, dass das Endprodukt intuitiv bedienbar ist und den Anforderungen der Zielgruppe entspricht. Dieser Ansatz betont die Bedeutung, die

tatsächlichen Bedürfnisse, Ziele und Fähigkeiten der Nutzer zu verstehen und in die Gestaltung einzubeziehen.

Obwohl beide Ansätze entscheidend für die Entwicklung qualitativ hochwertiger Produkte sind, liegt der Schwerpunkt dieser Arbeit nicht auf der Gestaltung selbst, sondern darauf, **wie UX erfasst und gemessen werden kann**. In einer zunehmend digitalisierten Welt, in der Anwendungen immer komplexer werden und die Erwartungen der Nutzer stetig steigen, ist die systematische Messung der UX von entscheidender Bedeutung. Nur durch ein tiefgehendes Verständnis des Nutzerverhaltens und der Nutzerzufriedenheit können Produkte entwickelt werden, die den hohen Ansprüchen gerecht werden.

1.2 Über den Landeswohlfahrtsverband Hessen

Der **Landeswohlfahrtsverband Hessen** (LWV Hessen) ist eine zentrale Organisation, die sich für die soziale Integration und Unterstützung von Menschen mit Behinderungen einsetzt. Als überörtlicher Träger der Eingliederungshilfe fördert der Verband Bereiche wie Wohnen, Bildung, Mobilität und Arbeit. Mit der Umsetzung des Bundesteilhabegesetzes und Programmen wie dem *Persönlichen Budget* und dem *Budget für Arbeit* unterstützt der LWV Hessen die selbstbestimmte Lebensführung und gesellschaftliche Teilhabe der Betroffenen. Zudem bietet der Verband Förderangebote für Kinder und Jugendliche mit geistigen, emotionalen oder körperlichen Einschränkungen, einschließlich Frühberatungsstellen und Förderschulen. Darüber hinaus ist der LWV Hessen Ansprechpartner für Menschen, die Leistungen des Sozialen Entschädigungsrechts benötigen.

1.3 Die ANLEI-Service GmbH

Die **ANLEI-Service GmbH** ist eine Tochtergesellschaft des LWV Hessen und bietet IT-Dienstleistungen für die Sozialverwaltung an. Sie entwickelt Systeme zur Unterstützung der Antragsbearbeitung, Leistungsgewährung und Abrechnung von Sozialleistungen, insbesondere in den Bereichen Eingliederungshilfe, Sozialhilfe und Kriegsopferfürsorge. Zu den zentralen Anwendungen zählen das *Integrierte Berichtssystem* (IBS) für betriebswirtschaftliche Analysen und *MASS* zur maschinellen Abrechnung mit Einrichtungen und Trägern.

1.4 Über eDok

eDok dient als Service für alle aufrufenden Anwendungen zur Erstellung von barrierefreien Dokumenten und löst das bisherige *Schriftstück-Erstellungssystem* (SE) ab, das auf Microsoft Word und Makros basiert. Es integriert *Doxee/Infonica* als Subsystem für die Dokumentbearbeitung und Output-Generierung, sodass eine Verwendung sowohl mit

interaktiven Elementen (Bearbeitung des entstehenden Dokuments) als auch im Hintergrund (Dunkelverarbeitung) erfolgen kann. Ziel und Vision von *eDok* ist es, eine universelle und generische Schnittstelle für die Anwendungen des LWV Hessen zu bieten, die eine einfache, effiziente und performante Erstellung von barrierefreien Dokumenten ermöglicht, unter Einbeziehung der Daten aus den Fachverfahren.

1.5 Motivation

Die rasante technologische Entwicklung und die immer stärkere Integration des Internets in den Alltag haben die Art und Weise, wie Menschen mit digitalen Systemen interagieren, grundlegend verändert. Laut der ARD/ZDF-Onlinestudie [11] nutzen mittlerweile rund 90 Prozent der deutschsprachigen Bevölkerung ab 14 Jahren regelmäßig das Internet, wobei insbesondere die mediale Internetnutzung in den letzten Jahren signifikant zugenommen hat. Diese Entwicklung stellt neue Anforderungen an die Gestaltung von Nutzererlebnissen und unterstreicht die Bedeutung einer herausragenden UX.

In diesem Kontext wird es immer wichtiger, digitale Inhalte und Dienste so zu gestalten, dass sie den sich wandelnden Ansprüchen der Nutzer gerecht werden. Anwendungen müssen nicht nur funktional sein, sondern auch intuitiv bedienbar und ästhetisch ansprechend gestaltet sein, um im Wettbewerb bestehen zu können. Dies gilt insbesondere für komplexe Webanwendungen wie *eDok*, die von Institutionen wie dem Landeswohlfahrtsverband Hessen (LWV Hessen) genutzt werden.

Eine besondere Herausforderung besteht darin, innerhalb solcher Anwendungen Bereiche zu identifizieren, die ein verbessertes Design oder zusätzliche Informationen erfordern. Nutzer können unsicher sein, wie sie ihre Ziele innerhalb der Anwendung erreichen können, oder sich von der Informationsmenge überfordert fühlen. Hier setzt das entwickelte Modell an, das die UX automatisiert erfasst und eine fundierte Analyse ermöglicht. So können gezielt Schlüsselaspekte identifiziert werden, die optimiert werden sollten, um die Nutzerzufriedenheit zu steigern.

Zudem führen die regelmäßige Einführung neuer Funktionen und Veränderungen in der Benutzeroberfläche zu einer dynamischen Komplexitätsentwicklung in verschiedenen Bereichen der Anwendung. Ein Überblick über die Entwicklung der UX im Laufe der Zeit ermöglicht eine proaktive Anpassung und Optimierung, sodass die Anwendung trotz zunehmender Komplexität benutzerfreundlich bleibt.

Die wissenschaftliche Forschung betont die Wichtigkeit von Methoden zur UX-Messung und -Bewertung, um sicherzustellen, dass die Produktentwicklung in die richtige Richtung geht [13]. Die Fähigkeit, UX quantitativ und qualitativ zu erfassen, bietet Unternehmen und Entwicklern einen unschätzbaren Vorteil. Durch das Verständnis des Nutzerverhaltens und der Nutzerzufriedenheit können gezielte Verbesserungen vorgenommen werden, die nicht nur die Effizienz steigern, sondern auch die Bindung der Nutzer an das Produkt fördern.

1.6 Ziel der Arbeit

Das Hauptziel dieser Arbeit ist die Entwicklung und Implementierung eines Modells, das eine umfassende Erfassung und Bewertung der UX ermöglicht. Dieses Modell soll spezifische, anonymisierte Daten sammeln, um die UX präzise und systematisch zu erfassen. Durch die Analyse dieser Daten über Zeiträume hinweg sollen dynamische Veränderungen erkannt und gezielte Optimierungsmöglichkeiten der Anwendung erschlossen werden.

Exemplarisch wird dieses Modell auf die Anwendung *eDok* des LWV Hessen angewendet. Dabei soll ein System geschaffen werden, das während der produktiven Nutzung Daten sammelt, ohne die Privatsphäre der Nutzer zu beeinträchtigen. Diese Daten werden in einem Dashboard visualisiert, das ausschließlich für Administratoren zugänglich ist, um fundierte Entscheidungen zur Optimierung der Anwendung treffen zu können.

Ein weiterer Aspekt ist die Möglichkeit für Entwickler und Produktverantwortliche, die Datenerfassung auf spezifische Seiten der Anwendung zu beschränken oder zu erweitern. Dadurch kann sichergestellt werden, dass nur relevante Bereiche beobachtet werden, ohne die Performance auf anderen Seiten zu beeinträchtigen. Zudem ermöglicht das Modell, die Effektivität von Anpassungen zu überprüfen, indem Veränderungen im Nutzerverhalten nach Implementierung von Updates sichtbar gemacht werden.

1.7 Fragestellung

Welche spezifischen, anonymisierten Daten sind erforderlich, um ein Modell zu entwickeln und zu implementieren, das die User Experience präzise und systematisch erfasst, dynamische Veränderungen über Zeiträume hinweg analysiert, Verantwortliche bei fundierten Entscheidungen zur Optimierung der Anwendung unterstützt und dabei auftretende Herausforderungen bewältigt?

1.8 Eigenleistung

Im Rahmen der vorliegenden Arbeit wurde auf keinerlei nicht-öffentliche Vorarbeiten zurückgegriffen. Sämtliche entwickelte Softwarekomponenten (Angular-Frontend, Java-Spring-Backend sowie die SQL-Datenbankstrukturen) wurden von mir selbst erstellt oder basieren ausschließlich auf frei verfügbaren bzw. quelloffenen Bibliotheken, die unter entsprechend erlaubten Lizenzen stehen (Angular CLI, Spring Boot, SQLite).

1.9 Grundlagen und Methodik

In diesem Kapitel werden die methodischen und technischen Grundlagen vorgestellt, welche die Basis für die spätere Umsetzung im Rahmen dieser Arbeit bilden. Dabei

wird auf bereits existierende Konzepte und Bibliotheken verwiesen, die im Projekt verwendet wurden. Zudem wird erläutert, wie die Datenbankstruktur und die zugrunde liegende Architektur gewählt wurden, um die Zielsetzungen der Arbeit (Erfassung, Analyse und Visualisierung von Nutzerinteraktionen) zu erfüllen. Auf detaillierte Implementierungsschritte wird hier verzichtet; diese finden sich in den folgenden Kapiteln und im Anhang.

1.9.1 Architektur und Softwarekomponenten

Die gewählte Systemarchitektur besteht aus einem Angular-Frontend und einem Spring-Boot-Backend, das auf einer relationalen Datenbank (z. B. SQLite) aufsetzt:

- **Frontend (Angular)**
 - Zuständig für die Erfassung der Nutzerinteraktionen (z. B. Klicks, Fehlermeldungen) und die Darstellung der Ergebnisse im Admin-Dashboard.
 - Setzt u. a. auf asynchrone Service-Aufrufe und reaktive Konzepte (RxJS), um Daten in Echtzeit oder in festgelegten Intervallen zu erfassen und an das Backend zu schicken.
- **Backend (Spring Boot)**
 - Stellt REST-Schnittstellen zur Verfügung, die vom Frontend konsumiert werden.
 - Übernimmt die Geschäftslogik, indem es eingehende Events (Klicks, Fehlermeldungen usw.) verarbeitet, validiert und in der Datenbank persistiert.
 - Aggregiert die Daten (z. B. zur Erstellung von Heatmaps oder zur Analyse von Navigationspfaden) und sendet die Ergebnisse zurück ans Frontend.
- **Datenhaltung (relationales Modell)**
 - Datenbanktabellen, in denen Sitzungen (Sessions), Klick-Events, Fehler-Events und Netzwerkabfragen erfasst werden.
 - Trennung zwischen dynamischen Nutzungsdaten (z. B. Klickereignissen) und statischen Entitäten (z. B. Komponenten-Beschreibungen), um eine klare Struktur und Erweiterbarkeit zu gewährleisten.

Diese Aufteilung ermöglicht eine lose Kopplung zwischen Frontend und Backend sowie eine modulare Erweiterbarkeit der Datenerfassung und Analyse.

1.9.2 Datenbankmodell

Zur Speicherung und Auswertung der erhobenen Interaktionsdaten kommt ein relationales Datenmodell zum Einsatz. Wesentliche Entitäten sind:

- **Usability Session:** Repräsentiert eine anonyme Sitzung und verknüpft diese mit einer konkreten Komponente (z. B. einem Modul der Anwendung). Enthält Start- und Endzeit sowie ein Session-Token als eindeutigen Bezeichner.
- **Click Event:** Zeichnet jede Klick-Interaktion innerhalb einer Session auf. Wichtige Felder sind das angeklickte UI-Element (`element_id`), eine Sequenznummer zur Rekonstruktion der Klickfolge und ein Zeitstempel.
- **Error Event:** Erfasst Fehler, die während einer Session auftreten (z. B. Ausnahmen im Frontend oder gemeldete Fehler über ein Formular). Ermöglicht die Kategorisierung und Zählung von Fehlern, um Problembereiche zu identifizieren.
- **Network Request:** Speichert Informationen zu einzelnen Netzwerkaufrufen wie Startzeit, Endzeit, HTTP-Methode und ggf. Antwortstatus. Wichtig zur Performance-Analyse und zum Auffinden von Engpässen.
- **Component:** Enthält Metadaten zu einer (Sub-)Anwendung bzw. UI-Komponente, sodass Sitzungen und Events eindeutig zugeordnet werden können.

Dieses Datenmodell ermöglicht eine gezielte Analyse von Nutzerwegen, Klickhäufigkeiten, Fehlerursachen und Performance-Daten, ohne dabei personenbezogene Daten zu erheben. Durch die Entkopplung zwischen „Sitzungen“ und konkreten Nutzern wird die Privatsphäre gewahrt.

1.9.3 Methodisches Vorgehen bei der Datenerfassung

- **Event Listener im Frontend:** In Angular-Komponenten werden Klick- und Fehlerevents über globale oder lokale Listener abgefangen. Die Daten (Element-ID, Zeitstempel, Session-Token usw.) werden minimal verarbeitet (z. B. Ergänzung eines Sequenzzählers) und anschließend an das Backend gesendet.
- **Persistierung und Anreicherung im Backend:** Das Spring-Boot-Backend nimmt die Daten entgegen, validiert sie und legt sie in den entsprechenden Tabellen ab (siehe Abschnitt ??). Zudem können Zeitinformationen (z. B. `time_since_previous`) ergänzt oder bestehende Sessions automatisch geschlossen werden, wenn eine Inaktivität vorliegt.
- **Auswertung:** Um Heatmap-Daten, häufigste Pfade oder Fehlerstatistiken zu erzeugen, werden Aggregationsfunktionen und gruppierte Abfragen verwendet. Die Ergebnisse werden entweder direkt an das Frontend geliefert (z. B. für Live-Analysen) oder in periodischen Abständen abgerufen, um Berichte oder Dashboards zu aktualisieren.

-
- **Datenschutz:** Nur technische und kontextbezogene Daten werden gespeichert (z. B. Session-Token statt Nutzer-ID). Keine Speicherung von Klartext-Personendaten. Konzeption und Implementierung sind darauf ausgelegt, geltende Datenschutzvorgaben einzuhalten.

1.9.4 SQL-Datenbankmodell

```
1 CREATE TABLE click_event (  
2     id                BIGINT NOT NULL PRIMARY KEY,  
3     element_id        VARCHAR(255),  
4     sequence_number   INTEGER NOT NULL,  
5     time_since_previous INTEGER,  
6     timestamp         TIMESTAMP,  
7     usability_session_id BIGINT NOT NULL  
8 );  
9  
10 CREATE TABLE component (  
11     id                BIGINT NOT NULL PRIMARY KEY,  
12     description       VARCHAR(255),  
13     name              VARCHAR(255)  
14 );  
15  
16 CREATE TABLE network_request (  
17     id                BIGINT NOT NULL PRIMARY KEY,  
18     duration_ms       INTEGER,  
19     request_end_time   TIMESTAMP,  
20     request_method     VARCHAR(255),  
21     request_start_time TIMESTAMP,  
22     request_url        VARCHAR(255),  
23     response_payload   CLOB,  
24     status_code        INTEGER,  
25     usability_session_id BIGINT NOT NULL  
26 );  
27  
28 CREATE TABLE usability_session (  
29     session_id        INTEGER PRIMARY KEY,  
30     end_time          TIMESTAMP,  
31     session_token     VARCHAR(255),  
32     start_time        TIMESTAMP,  
33     component_id      BIGINT NOT NULL  
34 );
```

1.9.5 Zusammenfassung

Im Rahmen dieses Kapitels wurden die methodischen und technischen Grundlagen erläutert, welche für das Verständnis der späteren Umsetzung essenziell sind. Zentrale Punkte sind die lose gekoppelte Architektur (Angular-Frontend, Spring-Boot-Backend, relationale Datenbank), das klar strukturierte Datenmodell für Nutzungs- und Fehlerevents sowie das damit verbundene methodische Vorgehen bei der Datenerfassung und -auswertung.

Aufbauend auf diesen Grundlagen werden im folgenden Kapitel die konkreten Implementierungsdetails, Ablauflogiken und ausgewählte Code-Beispiele vorgestellt, um den Praxiseinsatz des Modells zu veranschaulichen. Längere Code-Auszüge erscheinen im Anhang und werden dort ausführlicher dokumentiert.

- **Kapitel 2** bietet einen ausführlichen Überblick über den theoretischen Hintergrund der UX-Messung, einschließlich gängiger Methoden und Techniken zur Datenerfassung und -analyse.
- In **Kapitel 3** wird das entwickelte Modell detailliert vorgestellt. Es werden die technischen Komponenten, die Datenbankstruktur sowie die Implementierungsdetails beschrieben.
- **Kapitel 4** widmet sich der Anwendung des Modells auf die *eDok*-Anwendung. Es werden die Ergebnisse der Datenerfassung präsentiert und analysiert.
- In **Kapitel 5** erfolgt eine kritische Diskussion der Ergebnisse, einschließlich der Vorteile des Modells, identifizierter Schwachstellen und möglichen Verbesserungen.
- **Kapitel 6** fasst die Arbeit zusammen und gibt einen Ausblick auf zukünftige Forschungs- und Entwicklungsmöglichkeiten.

1.10 Relevanz der Arbeit

Angeichts der wachsenden Bedeutung von UX in der Softwareentwicklung leistet diese Arbeit einen wichtigen Beitrag zur Praxis und Forschung. Das vorgestellte Modell ermöglicht es, UX-Daten in Echtzeit und unter realen Nutzungsbedingungen zu erfassen, ohne die Nutzererfahrung zu beeinträchtigen. Dies ist besonders relevant für Organisationen, die ihre Anwendungen kontinuierlich verbessern möchten, um den steigenden Erwartungen der Nutzer gerecht zu werden.

Durch die Anwendung auf die *eDok*-Anwendung des LWV Hessen wird demonstriert, wie das Modell in einer realen Umgebung implementiert werden kann. Die gewonnenen Erkenntnisse können als Grundlage für weitere Anwendungen dienen und dazu beitragen, die UX-Forschung in der Praxis voranzutreiben.

1.11 Datenschutz und DSGVO

Im Rahmen der Datenerfassung werden ausschließlich *technische* und *kontextbezogene* Informationen aufgezeichnet, die keine Rückschlüsse auf einzelne Personen zulassen. Die erhobenen Nutzungsdaten sind *anonymisiert* bzw. *pseudonymisiert*. Dazu wird jedem Nutzungsvorgang lediglich eine `session_token` zugeordnet, statt personenbezogene Identifikationsmerkmale zu verwenden.

Um die **DSGVO**-Konformität zu gewährleisten, wird zudem kein Bezug zwischen den erhobenen Daten und Nutzerprofilen hergestellt. Informationen wie IP-Adressen, Klarnamen oder E-Mail-Adressen werden nicht erfasst. Die protokollierten Daten (z. B. Klick-Ereignisse, Fehlermeldungen, Sitzungszeiten) enthalten ausschließlich Metadaten zum Nutzerverhalten.

Hinsichtlich der **Speicherdauer** existiert eine interne Aufbewahrungsrichtlinie, die eine regelmäßige Löschung oder Archivierung älterer Daten vorsieht. Auf diese Weise wird der Datenumfang kontrolliert und eine unbegrenzte Speicherung vermieden. Bei Bedarf kann das Löschkonzept an neue gesetzliche Vorgaben angepasst oder durch *Privacy-by-Design*-Konzepte ergänzt werden.

1.12 Performance-Überlegungen

Um eine hohe *Performance* zu gewährleisten, werden die erhobenen Daten nicht bei jedem Klick *sofort* synchron an das Backend gesendet, sondern *asynchron* verarbeitet. Konkret bedeutet dies, dass das Frontend eingehende Events zunächst lokal zwischenspeichert und in festgelegten Zeitintervallen (*Batching*) an das Backend übermittelt. Dadurch wird die Anzahl der HTTP-Anfragen reduziert, was eine deutliche Entlastung des Systems zur Folge hat.

Innerhalb des Backends erfolgt die *Persistierung* ebenfalls in asynchronen Prozessen, sodass umfangreiche INSERT- und UPDATE-Operationen nicht blockierend wirken. Diese Architektur stellt sicher, dass die Anwendung auch bei hohem Datenaufkommen performant bleibt und keine spürbaren Verzögerungen im Nutzerfluss auftreten. Bei Bedarf können die Zeitintervalle angepasst werden, um ein Gleichgewicht zwischen Datenaktualität und Systemlast herzustellen.

1.13 Fazit

Die systematische Erfassung und Analyse der UX ist unerlässlich, um moderne Softwareanwendungen an die Bedürfnisse und Erwartungen der Nutzer anzupassen. Durch die Entwicklung und Implementierung eines Modells zur UX-Messung in *eDok* wird gezeigt, wie Daten genutzt werden können, um die Nutzererfahrung zu verbessern und die Anwendung effizienter und benutzerfreundlicher zu gestalten. Diese Arbeit leistet somit einen wertvollen Beitrag zur Verbesserung von UX-Methoden und deren Anwendung in der Praxis.

2 Kapitel 2

2.1 Einführung in das Thema UX

Die **User Experience** (UX) ist ein Teilbereich der HCI, der in den frühen 1990er Jahren an Bedeutung gewann [4]. Mit dem Aufkommen neuer Technologien und der zunehmenden Verbreitung digitaler Produkte rückte der Fokus verstärkt auf die Benutzererfahrung und deren Qualität. UX konzentriert sich auf die Interaktion zwischen Mensch und System und zielt darauf ab, Produkte und Anwendungen so zu gestalten, dass sie nicht nur funktional sind, sondern auch emotional ansprechend und benutzerfreundlich.

2.2 Herausforderungen und Offenheit der UX-Definition

Die Offenheit der UX-Definition erlaubt es, das Thema aus verschiedenen Perspektiven zu betrachten und unterschiedliche Meinungen einzubeziehen, was zu einem umfassenderen und tieferen Verständnis des UX-Konzepts führt und die Entwicklung der Disziplin fördert. Gleichzeitig erschweren jedoch verschiedene fachliche Hintergründe und Vokabularien den Fortschritt [4].

Hassenzahl und Tractinsky thematisierten in ihrer Arbeit *User Experience – A Research Agenda* die Komplexität der UX und erklärten: „User Experience ist ein interessantes Phänomen: Es wurde von der HCI-Community schnell angenommen, aber oft kritisiert, da es vage und schwer fassbar ist. Der Begriff umfasst verschiedene Bedeutungen, von klassischer Usability bis hin zu Schönheit, hedonischen, affektiven und erfahrungsbasierten Aspekten der Technologienutzung“ [9].

2.3 Forschungsperspektiven: UX als vielschichtiges Phänomen

Sie erläutern weiter, dass sich die frühe Forschung im Bereich der HCI hauptsächlich auf Verhaltensziele in Arbeitsumgebungen konzentrierte. Dieser Fokus wurde jedoch später durch alternative Ansätze infrage gestellt, die die Bedeutung von Ästhetik, Emotionen und subjektiven Erfahrungen betonten [9].

In einem frühen Versuch, UX zu definieren, betonte Alben (1996) die Bedeutung von Ästhetik als wesentlichem Qualitätsaspekt von Technologie [9]. Hassenzahl argumentierte, dass interaktive Produkte aus zwei Perspektiven betrachtet werden können: den instrumentellen Aspekten (z. B. Usability) und den nicht-instrumentellen (hedonischen) Aspekten, die sich auf das emotionale und ästhetische Erleben der Nutzer beziehen [5].

Die Nielsen Norman Group definiert UX als die Gesamtheit aller Aspekte der Interaktion eines Endnutzers mit einem Unternehmen, seinen Dienstleistungen und Produkten. Hervorgehoben wird, dass herausragende UX nicht nur die spezifischen Bedürfnisse des Nutzers erfüllt, sondern auch durch Einfachheit und Eleganz überzeugt, sodass die Nutzung und der Besitz des Produkts als angenehm empfunden werden. Zudem

wird betont, dass eine qualitativ hochwertige UX eine nahtlose Integration verschiedener Disziplinen erfordert, darunter Ingenieurwesen, Marketing, Grafik- und Industriedesign sowie Interface-Design [3].

Die ISO 9241-210:2019 beschreibt UX als „sämtliche Wahrnehmungen und Reaktionen von Nutzern, die durch die tatsächliche oder erwartete Nutzung eines Systems, Produkts oder einer Dienstleistung hervorgerufen werden“ [1].

2.4 Zusammenfassung der UX-Definitionen

Ungeachtet der unterschiedlichen Definitionen ist klar, dass UX sowohl funktionale als auch emotionale Dimensionen der Nutzererfahrung umfasst. Faktoren wie Usability, ästhetische Wahrnehmung, inhaltliche Relevanz und das Vertrauen der Nutzer sind entscheidend dafür, wie effektiv und zufriedenstellend eine Anwendung wahrgenommen wird und tragen wesentlich zur Gestaltung positiver Nutzererlebnisse bei [10].

In dem Papier *Towards Practical User Experience Evaluation Methods* [8] wird darauf hingewiesen, dass die UX-Forschung eine Vielzahl von Modellen und Frameworks entwickelt hat. Diese Modelle adressieren die zentralen Herausforderungen der UX, wie ihre subjektive, kontextabhängige und dynamische Natur sowie die Balance zwischen pragmatischen und hedonischen Aspekten des Nutzererlebnisses. Gleichzeitig wird betont, dass UX zunehmend in der Industrie übernommen wird, die Produktentwicklung jedoch noch immer stark auf traditionellen Usability-Methoden basiert. Die Arbeit unterstreicht die Notwendigkeit praxisnaher UX-Bewertungsmethoden für die Produktentwicklung in der Industrie.

2.5 Usability und ihre Bedeutung

Usability bezieht sich auf die funktionalen Aspekte eines Produkts und beschreibt, wie effektiv und effizient sich das Produkt nutzen lässt. Viele Experten sehen Usability als Teil der UX [2]. Während sich Usability auf die Fähigkeit des Nutzers konzentriert, ein System erfolgreich zu nutzen, umfasst UX auch die gesamte Nutzererfahrung und die dabei entstehenden Emotionen.

Die Nielsen Norman Group definiert Usability als ein Qualitätsmerkmal, das bewertet, wie einfach und angenehm eine Benutzeroberfläche zu nutzen ist. Sie umfasst fünf Hauptkomponenten:

- **Erlernbarkeit:** Wie leicht können Benutzer grundlegende Aufgaben beim ersten Mal ausführen?
- **Effizienz:** Wie schnell können Benutzer Aufgaben ausführen, nachdem sie die Schnittstelle gelernt haben?
- **Merkfähigkeit:** Wie einfach können Benutzer ihre Fähigkeiten wiederherstellen, wenn sie das Design nach einer Pause erneut verwenden?

-
- **Fehlerrate:** Wie viele Fehler machen Benutzer, wie schwerwiegend sind diese und wie leicht können sie sich von ihnen erholen?
 - **Zufriedenheit:** Wie angenehm ist die Nutzung der Benutzeroberfläche?

Die Nielsen Norman Group erläutert den Unterschied zwischen Usability Testing und Usability Evaluation klar und prägnant. Beim Usability Testing beobachten Usability-Experten die Nutzer, um potenzielle Usability-Probleme zu identifizieren. Im Gegensatz dazu geht die Usability Evaluation einen Schritt weiter: Sie kombiniert das Testing mit direktem Nutzerfeedback und einer Überprüfung des Systemdesigns.

2.6 Usability in eDok

Im Kontext von *eDok* liegt der Schwerpunkt auf der effizienten und fehlerarmen Erstellung und Bearbeitung von Dokumenten. Für optimale Usability ist es entscheidend, dass Nutzer die Funktionen der Anwendung intuitiv verstehen und nutzen können. Klare Benutzerführung, schnelle Ladezeiten sowie eine übersichtliche und reduzierte Benutzeroberfläche sind dabei zentral.

Ein wesentlicher Aspekt ist die Unterstützung von Fehlertoleranz, sodass auch bei Eingabefehlern ein produktiver Arbeitsfluss erhalten bleibt. Funktionen wie automatisches Speichern, leicht zugängliche Rückgängig-Optionen und verständliche Fehlermeldungen helfen, die Nutzererfahrung positiv zu gestalten.

Durch die kontinuierliche Evaluation der Usability über systematisch gesammelte Daten können Schwachstellen identifiziert und gezielt Optimierungen vorgenommen werden. Ergänzende Nutzerbefragungen können zusätzliche Einblicke bieten, sind jedoch nicht direkt in das System integriert und dienen daher eher unterstützenden Zwecken.

Insgesamt zielt *eDok* darauf ab, eine nutzerfreundliche Anwendung zu bieten, die den Arbeitsablauf bei der Dokumentenerstellung optimiert. Diese Arbeit strebt an, bestimmte Merkmale der Usability zu messen, zu speichern und zu bewerten, um ein umfassendes Bild der Nutzererfahrung zu erhalten.

2.7 Erweiterte Ziele der UX- und Usability-Bewertung in eDok

Das Ziel der UX- und Usability-Bewertung in *eDok* ist es, gezielte Einblicke in die Nutzungsmuster der Anwendung zu gewinnen und Optimierungspotenziale klar zu identifizieren. Das entwickelte Modell liefert durch systematische Erfassung und Analyse der Nutzerinteraktionen Daten, die Entwickler und Produktverantwortliche dabei unterstützen, Nutzerbedürfnisse und häufige Herausforderungen sichtbar zu machen.

Die implementierte Heatmap bietet eine visuelle Übersicht der am stärksten genutzten Bereiche, sodass gezielt Verbesserungspotenziale an zentralen Interaktionspunkten erkannt werden können. Durch die Analyse der Verweildauer auf spezifischen Seiten,

der häufig durchlaufenen Navigationspfade und der Häufigkeit bestimmter Fehler können gezielte Maßnahmen zur Optimierung der Anwendung abgeleitet werden.

Diese Daten helfen, die Effizienz und Benutzerfreundlichkeit der Anwendung kontinuierlich zu steigern und eine datengetriebene Produktoptimierung zu ermöglichen. Darüber hinaus ermöglicht das Modell eine nachträgliche Überprüfung der implementierten Änderungen, um deren Effektivität zu bewerten und weitere Optimierungen vorzunehmen. Durch diesen iterativen Prozess trägt das Modell wesentlich zur Steigerung der Nutzerzufriedenheit und Effizienz der Anwendung bei.

2.7.1 Abgrenzung von UX- und Usability-Evaluationsmethoden

Während Usability-Tests sich primär auf die Leistung bei der Aufgabenbearbeitung konzentrieren, fokussieren UX-Evaluationsmethoden auf das subjektive Erleben der Nutzenden. Objektive Metriken wie Ausführungszeit oder Klickanzahl reichen nicht aus, um die UX vollständig zu erfassen; vielmehr müssen auch Motivation, Erwartungen und Emotionen der Nutzenden berücksichtigt werden [16].

Nach eingehender Recherche und Analyse der verschiedenen UX-Evaluationsmethoden hat sich herausgestellt, dass die Anforderungen dieser Arbeit eher einem Usability-Test entsprechen als traditionellen UX-Evaluationsmethoden wie Fragebögen oder der Beobachtung von Nutzenden. Unsere Arbeit konzentriert sich auf die systematische Erfassung von Nutzungsdaten während des Produktiveinsatzes der Anwendung. Diese Herangehensweise erlaubt es, objektive Daten über die Interaktion der Nutzenden mit der Anwendung zu sammeln und spezifische Aspekte der Usability zu bewerten.

2.7.2 Fokussierung auf Usability in dieser Arbeit

Die Entscheidung, den Schwerpunkt auf Usability-Tests zu legen, basiert auf den spezifischen Anforderungen des Projekts. Da die gesammelten Daten nicht benutzerorientiert sein sollen und keine direkten Nutzerbefragungen oder Beobachtungen stattfinden, sind traditionelle UX-Evaluationsmethoden weniger geeignet. Stattdessen ermöglicht die Messung von Interaktionsdaten, wie zum Beispiel Klickpfade, Verweildauer auf bestimmten Seiten und Fehlerhäufigkeiten, eine objektive Analyse der Benutzerfreundlichkeit der Anwendung.

Diese Daten liefern wertvolle Erkenntnisse darüber, wie effizient und effektiv die Nutzenden die Anwendung bedienen können. Sie helfen dabei, potenzielle Usability-Probleme zu identifizieren und gezielte Verbesserungen vorzunehmen. Obwohl dieser Ansatz nicht das gesamte Spektrum der UX abdeckt, trägt er wesentlich zur Optimierung der Anwendung bei und unterstützt die Nutzenden in ihrer täglichen Arbeit.

Jordan (2008)[7] betont die Bedeutung einer sorgfältigen Auswahl der Usability-Evaluationsmethode unter Berücksichtigung spezifischer Projektanforderungen und Kontextfaktoren. In Anlehnung an seine Kriterien hat sich herausgestellt, dass eine Methode, die auf der automatisierten Erfassung von Nutzungsdaten basiert, geeignet ist. Diese Entscheidung ermöglicht es, objektive Daten zu sammeln, ohne die Privatsphäre der

Nutzenden zu beeinträchtigen, und entspricht somit den praktischen und ethischen Anforderungen des Projekts.

Im Jahr 1980 führte das Xerox Palo Alto Research Center (PARC)[14] eine bahnbrechende Studie durch. Sie verglichen die vorhergesagten Ausführungszeiten mit den tatsächlich gemessenen Zeiten und stellten fest, dass das Keystroke-Level Model KLM die Benutzerleistung mit einer Genauigkeit von etwa 21% vorhersagen konnte. Damit zeigten sie, dass ihr Modell ein nützliches Werkzeug für Designer von interaktiven Systemen ist, um die Effizienz von Benutzerschnittstellen quantitativ zu bewerten und zu verbessern.

Die von Paz und Pow-Sang durchgeführte Studie [12] identifizierte die am häufigsten verwendeten Methoden zur Evaluierung der Benutzerfreundlichkeit (Usability Evaluation Methods, UEMs) in Softwareentwicklungsprozessen:

- **Umfragen/Fragebögen (26,26%)**
 - Die am häufigsten verwendete Methode.
 - Geschätzt für ihre Einfachheit und Effektivität bei der Erfassung von Benutzerzufriedenheitsdaten.
- **Benutzertests (14,14%)**
 - Beinhaltet die Beobachtung realer Benutzer, die mit der Software interagieren.
 - Ziel: Direkte Identifikation von Usability-Problemen.
- **Heuristische Evaluation (12,63%)**
 - Experten bewerten die Software anhand etablierter Usability-Prinzipien.
 - Ziel: Potenzielle Probleme aufdecken.
- **Interviews (10,35%)**
 - Direkte Gespräche mit Benutzern.
 - Ziel: Tiefgehende Einblicke in Usability-Bedenken und -Erfahrungen.
- **Lautes Denken (9,60%)**
 - Benutzer äußern ihre Gedanken während der Nutzung der Software.
 - Ziel: Echtzeit-Feedback zu ihren Interaktionen liefern.

Im Kontext der Ergebnisse der Studie, obwohl diese Methoden nicht zu den fünf am häufigsten verwendeten zählen (diese sind Umfrage/Fragebogen, Benutzertests, heuristische Evaluation, Interview und "Thinking AloudProtokoll), werden sie dennoch als anerkannte und wertvolle Techniken in der Usability-Evaluierung betrachtet.

Durch den Einsatz dieser Methoden verwenden Sie quantitative Datenanalyse, um die Benutzerfreundlichkeit zu evaluieren. Dieser Ansatz ermöglicht es, spezifische Probleme in der Benutzeroberfläche basierend auf dem tatsächlichen Nutzerverhalten zu identifizieren, was zu gezielteren und effektiveren Verbesserungen führt.

Die Studie von Srivastava et al. (2000)[15] untersucht das Konzept des *Web Usage Mining*, bei dem Data-Mining-Techniken angewendet werden, um Nutzungsmuster aus Web-Daten zu entdecken. Ziel ist es, das Verhalten von Web-Nutzern zu verstehen und daraus Erkenntnisse für die Verbesserung von Web-Anwendungen zu gewinnen.

Die Autoren identifizieren verschiedene Arten von Web-Daten, darunter Inhaltsdaten, Strukturdaten, Nutzungsdaten und Benutzerprofilen. Sie beschreiben einen dreistufigen Prozess des Web Usage Mining:

1. **Datenvorverarbeitung:** Reinigung und Transformation der Rohdaten, Identifikation von Benutzern und Sessions.
2. **Musterdetektion:** Anwendung von Data-Mining-Techniken wie Assoziationsanalyse und Clustering zur Identifikation von Nutzungsmustern.
3. **Musteranalyse:** Interpretation der entdeckten Muster, um nützliche Erkenntnisse zu gewinnen.

Die Studie zeigt, dass die Analyse von Nutzungsdaten wie Heatmaps, Fehlerzählungen, Klickanalysen und Verweildauer entscheidend ist, um das Nutzerverhalten zu verstehen und die Usability von Websites zu verbessern. Durch Web Usage Mining können spezifische Probleme in der Benutzeroberfläche identifiziert und gezielte Verbesserungen vorgenommen werden.

Die Autoren fanden heraus, dass Web Usage Mining in verschiedenen Anwendungsbereichen wie Personalisierung, Systemoptimierung, Website-Modifikation und Geschäftsanalysen wertvolle Einblicke liefert. Sie betonen jedoch auch die Herausforderungen, insbesondere in Bezug auf Datenqualität und Datenschutz.

Insgesamt unterstreicht die Studie die Bedeutung von Web Usage Mining als effektives Werkzeug zur Verbesserung von Web-Anwendungen durch ein besseres Verständnis des Nutzerverhaltens.

Diese Arbeit befasste sich mit der Sammlung und Darstellung von Daten, es wurden jedoch keine Data-Mining-Techniken angewendet. Es ist jedoch üblich, solche Methoden auf die gesammelten Daten anzuwenden, was zu interessanteren Erkenntnissen führen kann, wie in der Studie erwähnt.

In dieser Arbeit wurden stattdessen statistische Operationen durchgeführt, um einige Kennzahlen wie die Fehlerquote zu veranschaulichen.

2.7.3 Vergleich von Nutzerinterviews, Fragebögen und der Sammlung quantitativer Daten

In der Usability-Forschung gibt es verschiedene Methoden, um Nutzererfahrungen und -probleme zu evaluieren. Zu den häufig eingesetzten Ansätzen gehören qualitative Me-

thoden wie Interviews und Fragebögen, die direkte Rückmeldungen der Nutzer einholen, und quantitative Ansätze wie das Logging und die Analyse von Nutzungsdaten. Nach eingehender Analyse und Berücksichtigung der Ergebnisse aus dem Papier *Extracting Usability Information from User Interface Events* von Hilbert und Redmiles [6] wurde entschieden, dass eine Kombination dieser Methoden die umfassendsten Erkenntnisse liefert.

2.7.4 Nutzerinterviews und Fragebögen

Nutzerinterviews und Fragebögen bieten eine direkte Möglichkeit, subjektive Rückmeldungen zu sammeln. Die Vorteile dieser Ansätze sind:

- **Qualitative Einblicke:** Diese Methoden liefern tiefgehende Informationen zu Meinungen, Zufriedenheit und Emotionen der Nutzer.
- **Flexibilität:** Während der Interaktion können gezielte Fragen gestellt werden, um spezifische Probleme zu beleuchten.
- **Direktes Feedback:** Nutzer äußern direkt wahrgenommene Schwächen oder Verbesserungsvorschläge.

Dennoch zeigen sich auch wesentliche Nachteile:

- **Subjektivität:** Ergebnisse hängen stark von individuellen Meinungen und Wahrnehmungen der Nutzer ab.
- **Hoher Aufwand:** Diese Methoden erfordern erheblichen Zeit- und Ressourcenaufwand.
- **Begrenzte Skalierbarkeit:** Sie sind für kleine Nutzergruppen geeignet, aber schwer auf größere Stichproben anzuwenden.
- **Fehlende Objektivität:** Die Erfassung tatsächlichen Nutzerverhaltens ist begrenzt.

2.8 Sammlung und Analyse quantitativer Daten

Der Ansatz von Hilbert und Redmiles [6] beschreibt die Vorteile der Erfassung von Nutzungsdaten, wie Mausbewegungen, Klicks und Navigationspfade, über automatisiertes Logging. Die wesentlichen Vorteile dieses Ansatzes sind:

- **Objektivität:** Die Erfassung basiert auf realem Nutzerverhalten und eliminiert subjektive Verzerrungen.
- **Automatisierung:** Daten können kontinuierlich und effizient gesammelt werden.
- **Skalierbarkeit:** Die Methode eignet sich für große Nutzerzahlen und breite Analysen.

-
- **Erkennung von Mustern:** Systematische Probleme, wie ineffiziente Workflows oder häufige Fehler, können identifiziert werden.

Allerdings gibt es auch Einschränkungen:

- **Fehlende qualitative Einblicke:** Emotionen oder Zufriedenheit der Nutzer können nicht direkt erfasst werden.
- **Komplexe Analyse:** Die Interpretation der Daten erfordert spezialisierte Expertise.
- **Abhängigkeit von Logging-Systemen:** Eine unzureichende Erfassung kann die Ergebnisse verzerren.

2.9 Erkenntnisse und Schlussfolgerung

Die Ergebnisse aus dem Papier von Hilbert und Redmiles [6] zeigen, dass die Kombination von qualitativen und quantitativen Ansätzen essenziell ist, um umfassende Usability-Einsichten zu gewinnen. Während Interviews und Fragebögen subjektive Erlebnisse und Zufriedenheit der Nutzer erfassen, bietet die Analyse quantitativer Daten objektive und skalierbare Einblicke in tatsächliches Nutzerverhalten. Das Event-Logging ermöglichte es, systematische Probleme zu identifizieren, die bei traditionellen Methoden übersehen wurden, wie ineffiziente Navigationspfade oder wiederkehrende Fehler.

Nach Prüfung der Ergebnisse wurde die Entscheidung getroffen, dass qualitative und quantitative Methoden nicht gegeneinander stehen, sondern sich ergänzen. Die Kombination beider Ansätze stellt sicher, dass sowohl die subjektiven als auch die objektiven Aspekte der Usability umfassend berücksichtigt werden können.

2.10 Methodik

Die in dieser Arbeit entwickelte Methodik zur Erfassung, Speicherung und Analyse von UX- und Usability-Daten in der Anwendung *eDok* umfasst mehrere aufeinander abgestimmte Schritte. Sie gewährleistet eine datenschutzkonforme, modulare und erweiterbare Infrastruktur, um Nutzungsinteraktionen systematisch zu sammeln, zu aggregieren und visuell darzustellen.

2.10.1 Systemarchitektur und Datenerfassung

Die Datenerfassung erfolgt clientseitig in der Frontend-Anwendung (Angular), wobei bei Nutzungsinteraktionen, wie etwa Mausklicks, Navigationsaktionen oder Fehlermeldungen, entsprechende Ereignisse (Events) abgefangen und in strukturierter Form an den Backend-Server (Spring Boot) gesendet werden. Diese Vorgehensweise ermöglicht eine kontinuierliche Erfassung von Interaktionsdaten während des Produktiveinsatzes, ohne dass Nutzer zusätzliche Handlungen vornehmen müssen.

2.10.2 Datenpersistenz und Datenbankmodell

Für die Speicherung der erfassten Daten wird eine relationale Datenbank eingesetzt, in der die zentralen Entitäten und ihre Beziehungen abgebildet sind. Hierfür kommen die Tabellen `click_event`, `component`, `error_event`, `network_request`, `ui_element` sowie `usability_session` zum Einsatz. Dabei bilden diese Tabellen folgende Kernaaspekte der Datenerfassung ab:

- `usability_session`: Enthält Sitzungsinformationen, wie etwa Start- und Endzeitpunkte sowie einen eindeutigen Session-Token. Diese Tabelle dient als Ankerpunkt für die Zuordnung sämtlicher Interaktionsdaten.
- `click_event`: Erfasst sämtliche Klickinteraktionen, inklusive Zeitstempel, betroffenen `ui_element`-Referenzen und Sequenznummern, um Reihenfolgen nachvollziehbar zu machen.
- `error_event`: Dokumentiert aufgetretene Fehlerzustände, deren Art, Häufigkeit und Position. Diese Daten ermöglichen es, gezielt fehleranfällige Bereiche der Anwendung zu identifizieren.
- `network_request`: Speichert Informationen zu Netzwerkaufrufen (Ladezeiten und Fehlermeldungen), um mögliche Performance-Engpässe und technische Probleme sichtbar zu machen.
- `component` und `ui_element`: Beschreiben statische Strukturen der Anwendung, indem sie Komponenten und deren zugehörige UI-Elemente referenzieren. Diese Aufteilung ermöglicht eine flexible Identifikation der Interaktionspunkte.

Die in den Tabellen erfassten Daten können durch Filterung, Aggregation und Zeitreihenanalysen zu aussagekräftigen Metriken verdichtet werden. Diese Datenmodellierung gewährleistet eine klare Trennung von dynamischen Nutzungsdaten und statischen UI-Strukturen, wodurch sowohl technische als auch inhaltliche Veränderungen im Frontend flexibel abbildbar bleiben.

2.10.3 Backend-Logik und REST-Schnittstellen

Das Backend stellt REST-Schnittstellen bereit, um Nutzungsdaten entgegenzunehmen, abzufragen und für die Visualisierung aufzubereiten. Bei Eintreffen neuer Datensätze führt die Backend-Logik Validierungs- und Anreicherungsprozesse durch, bevor die Daten in die entsprechende Tabelle geschrieben werden. Zudem bietet das Backend administrativen Nutzern Endpunkte zur Konfiguration des Erfassungsmodus, um etwa die Aufzeichnung auf spezifischen Seiten ein- oder auszuschalten.

2.10.4 Anonymisierung und Datenschutz

Um den Schutz der Privatsphäre der Nutzenden sicherzustellen, werden alle erhobenen Daten strikt anonymisiert. Es werden keine personenbezogenen Merkmale gespeichert. Die Datensätze lassen keine Rückschlüsse auf einzelne Individuen zu, sondern konzentrieren sich auf aggregierte Nutzungsmuster, Fehlerhäufigkeiten und Interaktionspfade.

2.10.5 Visualisierung und Heatmap-Darstellung

Die aufbereiteten Daten werden im Admin-Dashboard visualisiert. Hierfür wurden unter anderem Heatmaps implementiert, welche die am stärksten genutzten Bereiche der Benutzeroberfläche hervorheben. Durch die Einbettung von Sequenzinformationen (Reihenfolge der Klickereignisse) können Navigationspfade und Nutzungskontexte ermittelt werden. Fehlerereignisse werden gesondert markiert und helfen dabei, potenziell problematische UI-Elemente oder Prozessschritte zu identifizieren.

2.10.6 Iterative Verbesserung und Anpassung

Die vorgestellte Methodik ist iterativ angelegt: Anhand der gewonnenen Erkenntnisse lassen sich gezielte Verbesserungsmaßnahmen im UI-Design, in den Navigationsstrukturen oder in der Performance der Anwendung ableiten. Nach erfolgten Anpassungen können erneut Daten erhoben und mit früheren Ergebnissen verglichen werden, um die Effektivität der Verbesserungen zu bewerten und den Optimierungsprozess fortlaufend zu steuern.

3 Kapitel 3

3.1 Anforderungen

- Ein Dashboard zur Darstellung von spezifischen UX- und Usability-Berichten, die durch das Modell erfasst wurden.
- Nur Benutzer mit Administratorrechten können auf das Dashboard zugreifen.
- Der Administrator hat die Möglichkeit, das Modell auf beliebigen Seiten ein- oder auszuschalten.
- Erfassung der Zeit, die Nutzer auf einer Seite verbringen, zur Analyse des Nutzeroaufwands.
- Eine Heatmap mit Reihenfolgenverfolgung, um bestimmte Verhaltensmuster der Nutzer zu erkennen und visuell darzustellen.
- Zusätzlich zur Reihenfolgen-Funktion sollen Bearbeitungs-Eingabefelder erkannt und angezeigt werden.

-
- Die Heatmap wird dem Administrator im Dashboard zur Verfügung gestellt.
 - Fehler werden erfasst, gespeichert und im Dashboard angezeigt, wobei die Anzahl und die Position der Fehler besonders hervorgehoben werden.

3.2 Spezifikation der Erfassten UX-Daten in eDok

Zur systematischen Bewertung der Usability von eDok wird die Erfassung und Analyse spezifischer Datenarten priorisiert. Im Mittelpunkt steht die Sammlung von Interaktionsdaten, die durch den Einsatz von Heatmaps visualisiert werden. Diese ermöglichen detaillierte Einblicke in Navigationsmuster, identifizieren potenzielle Stolpersteine und decken ineffiziente Seitengestaltungen sowie suboptimale Darstellungen innerhalb der Anwendung auf. Ziel ist es, Schwachstellen in der Nutzerführung zu erkennen und datenbasierte Optimierungen der Benutzeroberfläche zu ermöglichen.

3.3 Implementierung der Heatmap

Diese Sektion beschreibt die Gestaltung und Implementierung einer Heatmap zur Nachverfolgung und Visualisierung von Nutzerinteraktionen in der eDok-Anwendung. Die Heatmap dient dazu, häufig genutzte Bereiche, Nutzerverhaltensmuster und fehleranfällige Interaktionen zu identifizieren, indem sie Daten zu Klicks, Fehlern und zeitlichen Metriken erfasst und darstellt.

3.4 Zweck und Ziele der Heatmap

Die Heatmap bietet eine visuelle Darstellung des Nutzerverhaltens und unterstützt Entwickler*innen dabei, die am häufigsten verwendeten Bereiche zu erkennen und Usability-Probleme zu identifizieren. Die Anzeige der Klickreihenfolge über Linienverbindungen hilft dabei, den Navigationsfluss innerhalb der Anwendung besser zu verstehen. Fehlerpunkte werden zudem in der Heatmap hervorgehoben und in einem separaten Admin-Dashboard für umfassende Berichte angezeigt. So können schwerwiegende Fehlerquellen identifiziert und priorisiert behoben werden.

3.5 Datenerfassung und Speicherung

Um die Heatmap-Funktionalität zu realisieren, werden verschiedene Arten von Interaktionsdaten erfasst und in einer strukturierten Datenbank gespeichert. Das Datenbankschema wurde so gestaltet, dass es zeitbasierte Nachverfolgung, Fehleraufzeichnung und Vergleich zwischen verschiedenen Versionen ermöglicht.

3.6 Implementierung Strategie

3.7 Global Heatmap Service

Ein zentraler Angular-Service, der folgende Funktionen erfüllt:

- Verfolgt, ob der Heatmap-Modus aktiviert oder deaktiviert ist.
- Ruft die aktuellen Klickdaten (oder andere Usability-Metriken) vom Backend ab.
- Stellt eine Datenstruktur bereit, die Element-IDs auf Klickanzahlen abbildet.
- Sendet Statusänderungen (an/aus) und aktualisierte Daten an abonnierten Komponenten und Direktiven weiter.

3.8 Overlay Directive

Eine Direktive, die global oder dynamisch im DOM eingefügt werden kann und folgende Aufgaben übernimmt:

- Scannt den DOM nach IDs der Elemente.
- Nutzt Daten aus dem Global Heatmap Service, um Overlays zu erstellen oder Stile anzuwenden.
- Entfernt Overlays bei Deaktivierung des Heatmap-Modus.

3.9 Admin Toggle Control

Ein einfacher Schalter im Admin-Panel, der:

- Eine Methode im Global Heatmap Service aufruft, um den Heatmap-Modus zu aktivieren oder zu deaktivieren.
- Statusänderungen an die verbundenen Komponenten und Direktiven weiterleitet.

3.10 Beispiele für Auswertungen

Ein zentraler Aspekt des entwickelten Modells ist die Gewinnung *quantitativer Metriken*, die Aussagen über die aktuelle Nutzungsweise von *eDok* erlauben und damit Potenziale zur Optimierung aufzeigen. Die folgenden Beispiele illustrieren typische Auswertungen, die sich aus den erfassten Daten (vgl. `click_event`, `error_event` usw.) ableiten lassen:

1. Klickhäufigkeit pro UI-Element

Hierbei wird gezählt, wie oft ein bestimmtes `element_id` (z. B. ein Button oder Formularfeld) angeklickt wurde. Eine einfache SQL-Abfrage sähe so aus:

Listing 1: Beispiel: Klickhäufigkeit pro Element

```
1 SELECT element_id, COUNT(*) AS total_clicks
2 FROM click_event
3 GROUP BY element_id
4 ORDER BY total_clicks DESC;
```

Diese Auswertung hilft zu erkennen, welche UI-Bereiche am häufigsten genutzt werden. Elemente mit *besonders hohem* oder *unerwartet niedrigem* Klickaufkommen können im Anschluss gezielt überprüft und ggf. neu gestaltet werden.

2. Fehlerquote (Fehler pro Klick / Zeit)

Um zu untersuchen, in welchem Verhältnis Klicks und Fehleraufkommen stehen, kann man beispielsweise die *Anzahl der Fehlerereignisse* (`error_event`) ins Verhältnis zu den Klick-Ereignissen setzen. Eine vereinfachte Kennzahl (Key Performance Indicator, KPI) wäre:

$$\text{Fehlerquote} = \frac{\sum(\text{Fehler})}{\sum(\text{Klicks})} \times 100 \%$$

oder zeitbasiert:

$$\text{Fehlerquote pro Zeiteinheit} = \frac{\sum(\text{Fehler in } t)}{\sum(\text{Klicks in } t)}.$$

Eine zugehörige Abfrage könnte beispielsweise so aussehen, wenn man pro Zeiteinheit (z. B. pro Tag) die Fehler- und Klickanzahl bestimmen möchte:

Listing 2: Beispiel: Fehler- und Klickanzahl pro Tag

```
1 SELECT DATE(timestamp) AS day,
2         (SELECT COUNT(*)
3          FROM error_event e
4          WHERE DATE(e.timestamp) = DATE(c.timestamp)) AS
5         total_errors,
6         COUNT(*) AS total_clicks
7 FROM click_event c
8 GROUP BY DATE(timestamp)
9 ORDER BY day;
```

Hierdurch lassen sich Zeiträume oder UI-Bereiche identifizieren, in denen *außergewöhnlich viele* Fehler auftreten, was ein klares Indiz für Optimierungsbedarf sein kann.

3. Navigationspfade (Häufigste Klick-Sequenzen)

Anhand der `sequence_number` und der Zuordnung zu `usability_session` kann

das System bestimmen, welchen Pfad Nutzende *typischerweise* innerhalb einer Sitzung durchlaufen (z. B. Startseite -> Dokument anlegen -> Einstellungen -> Speichern). Eine mögliche Herangehensweise ist das Gruppieren nach `session_id` und Sortieren nach der Klick-Sequenz:

Listing 3: Beispiel: Pfadrekonstruktion je Session

```
1 SELECT session_id, element_id, sequence_number
2 FROM click_event
3 ORDER BY session_id, sequence_number ASC;
```

Anschließend können sich wiederholende *Pfad-Signaturen* identifiziert werden (z. B. „Startseite->DokumentErstellen->...“), um sogenannte *most common paths* zu bestimmen. Dies liefert Aufschluss darüber, ob bestimmte Navigationsrouten gewünscht sind oder ob Nutzende häufig „Umwege“ gehen.

4. Verweildauer pro Seite

Die Analyse der *Verweildauer* ist hilfreich, um *Aufwände* oder *eventuelle Probleme* zu erkennen (z. B. sehr lange Bearbeitungszeiten für ein Formular). Dazu werden z. B. Page-Load- und Page-Unload-Events (oder ein entsprechender `time_since_previous`-Wert) protokolliert. Die Differenz zwischen beiden Zeitpunkten ergibt:

$$\text{Verweildauer}(\text{Seite}_i) = \text{Endzeitpunkt} - \text{Startzeitpunkt}.$$

Um beispielsweise die durchschnittliche Verweildauer pro `usability_session` zu ermitteln, kann man die `time_since_previous` Werte summieren oder einen separaten `page_load` / `page_unload` Event-Ansatz verfolgen. Eine vereinfachte SQL-Abfrage (je nach Umsetzung) könnte so aussehen:

Listing 4: Beispiel: Durchschnittliche Verweildauer je Session

```
1 SELECT usability_session_id,
2        AVG(time_since_previous) AS avg_time_spent
3 FROM click_event
4 GROUP BY usability_session_id
5 ORDER BY usability_session_id;
```

Seiten mit *auffallend langen* Verweilzeiten könnten zu komplex oder nicht intuitiv gestaltet sein, wohingegen *extrem kurze* Zeiten auf Desinteresse oder eine Fehlbedienung hinweisen können.

Ableitung von KPIs und weiterführenden Auswertungen.

In der Praxis werden diese Basisanalysen häufig zu komplexeren *Key Performance Indicators* (KPIs) kombiniert, etwa *Fehler pro 100 Klicks* oder *durchschnittliche Navigationszeit* zwischen zwei häufig aufgerufenen Elementen. Die in den Tabellen `click_event`,

`error_event` und `usability_session` abgelegten Daten bieten somit eine *hohe Flexibilität*, um je nach Fragestellung passende Metriken zu berechnen und daraus konkrete Handlungsempfehlungen abzuleiten.

3.11 Bezug zu den Zielen (Kapitel 1)

In **Kapitel 1** wurde als Hauptziel dieser Arbeit definiert, die *User Experience (UX)* in der Anwendung *eDok* systematisch zu erfassen, zu analysieren und durch konkrete Maßnahmen zu verbessern. Dabei sollten folgende Aspekte besonders berücksichtigt werden:

- **Kontinuierliche Datenerfassung ohne Mehraufwand für Nutzende:** Das automatisierte *Event Logging* (z. B. in `click_event`) ermöglicht eine lückenlose Erfassung der Interaktionen, ohne dass Benutzer zusätzliche Aktionen ausführen müssen.
- **Anonymisierung und Datenschutz:** Durch die Abstraktion über die Tabelle `usability_session` bleibt die Identität der Nutzenden geschützt, da statt personenbezogener Daten lediglich pseudonymisierte Tokens gespeichert werden.
- **Übersichtliche Visualisierung und Trendanalysen:** Die in `click_event`, `network_request` und `error_event` gespeicherten Daten lassen sich zu aussagekräftigen *Heatmaps* und *Fehlerstatistiken* zusammenführen, sodass Administratoren im Dashboard einen klaren Überblick gewinnen.
- **Gezielte Optimierung von UI- und Nutzungsprozessen:** Anhand der gesammelten `sequence_number`-Informationen kann z. B. die Reihenfolge von Klicks analysiert werden, um häufige Pfade (oder Stolpersteine) zu identifizieren und das Benutzererlebnis weiterzuentwickeln.

Unterstützung durch die Tabellenstruktur. Die Datenbanktabellen (`usability_session`, `click_event`, `error_event` etc.) implementieren die oben genannten Ziele wie folgt:

1. `click_event` erfasst alle relevanten Interaktionen (Element-ID, Zeitstempel, Reihenfolge), die für *Heatmaps* und Klickstatistiken benötigt werden.
2. `usability_session` gewährleistet eine saubere *Session-Verfolgung*, ohne Rückschlüsse auf konkrete Personen zuzulassen. So kann z. B. die Verweildauer oder die Anzahl der Fehler pro Session analysiert werden.
3. `error_event` und `network_request` ermöglichen eine *tiefergehende Performance- und Fehleranalyse*, um systematische Probleme aufzudecken.

Erreichte Ziele. Die Arbeit konnte die gesteckten Ziele im Wesentlichen erfüllen:

- Die Datenerfassung und -speicherung funktioniert *kontinuierlich und anonymisiert*, sodass *eDok* unter realen Nutzungsbedingungen beobachtet werden kann, ohne datenschutzrechtliche Bedenken.
- Das *Dashboard* stellt die gesammelten Daten visuell dar und bietet Administrator*innen eine solide Basis, um entscheidungsrelevante Informationen (z. B. häufige Klickpfade, Fehlerhotspots) auf einen Blick zu erkennen.
- *Iterative Verbesserungen* lassen sich auf Grundlage der gewonnenen Erkenntnisse realisieren, indem problematische UI-Bereiche oder ineffiziente Interaktionsmuster konkret adressiert werden.

Nicht vollständig abgedeckte Ziele. Trotz der erfolgreichen Implementierung bleiben bestimmte Bereiche *unberücksichtigt*:

- **Keine subjektiven UX-Daten:** Da keinerlei Umfragen, Interviews oder Beobachtungen einbezogen wurden, fehlen tiefergehende Einblicke in Emotionen, Motivation und Zufriedenheit der Nutzenden.

Ungeachtet dieser Einschränkungen bildet das entwickelte Modell eine solide Grundlage, um die *Usability* von *eDok* kontinuierlich zu überwachen und gezielt zu verbessern. Zukünftige Arbeiten könnten um qualitative Methoden oder weiterführende Analysetechniken ergänzt werden, um ein noch umfassenderes Bild der *User Experience* zu gewinnen.

Literatur

- [1] ISO 9241-210:2019(en). *Ergonomics of human-system interaction — Part 210: Human-centred design for interactive systems*. <https://www.iso.org/obp/ui/en/#iso:std:iso:9241:-210:ed-2:v1:en>, abgerufen am 05.10.2024.
- [2] Halina Mohamed Dahlan A. H. Allam Ab Razak Che Hussin. *User Experience: Challenges and Opportunities*. https://seminar.utmspace.edu.my/jisri/download/F1_FinalPublished/Pub4_UserExperienceChallenges.pdf, abgerufen am 02.10.2024.
- [3] Jakob Nielsen Don Norman. *The Definition of User Experience (UX)*. <https://www.nngroup.com/articles/definition-user-experience/>, abgerufen am 02.10.2024.
- [4] Michael Glanznig. *Approaching User Experience: Moving between Modelling and In-Depth*. <https://phaidra.univie.ac.at/detail/o:1301839.pdf>, abgerufen am 11.10.2024. 2013.

-
- [5] Marc Hassenzahl. *The Thing and I: Understanding the Relationship Between User and Product*. https://www.researchgate.net/publication/226420570_The_Thing_and_I_Understanding_the_Relationship_Between_User_and_Product, abgerufen am 21.10.2024. 2006.
- [6] David M. Hilbert und David F. Redmiles. „Extracting usability information from user interface events“. In: *ACM Computing Surveys (CSUR)* 32.4 (2000), S. 384–421.
- [7] Philipp Jordan. *Auswahl einer geeigneten Methode zur Usability Evaluation*. Studienarbeit, Institut für Konstruktionstechnik und Technisches Design, Universität Stuttgart. Betreuer: Prof. Dr.-Ing. Thomas Maier und Dr.-Ing. M. Schmid. Juni 2008. URL: https://www.researchgate.net/publication/275833180_Selection_of_an_appropriate_Usability_Evaluation_Method_GERMAN_LANGUAGE.
- [8] Marc Hassenzahl Kaisa Väänänen-Vainio-Mattila Virpi Roto. *Towards Practical User Experience Evaluation Methods*. https://www.researchgate.net/publication/239749277_Towards_Practical_User_Experience_Evaluation_Methods, abgerufen am 03.10.2024.
- [9] Noam Tractinsky Marc Hassenzahl. *User experience – a research agenda*. https://www.researchgate.net/publication/233864602_User_experience_-_A_research_agenda, abgerufen am 14.10.2024. 2006.
- [10] Martin Salaschek Meinold T. Thielsch. *Toolbox zur Website-Evaluation: Erfassung der User Experience von Onlinegesundheitsinformationen*. <https://d-nb.info/121533060X/34>, abgerufen am 03.10.2024.
- [11] Carmen Schäfer Natalie Beisch Wolfgang Koch. *ARD/ZDF-Onlinestudie 2019: Mediale Internetnutzung und Video-onDemand gewinnen weiter an Bedeutung*. https://www.ard-zdf-onlinestudie.de/files/2019/0919_Beisch_Koch_Schaefer.pdf, abgerufen am 03.10.2024.
- [12] Freddy Paz und José Antonio Pow-Sang. „A Systematic Mapping Review of Usability Evaluation Methods for Software Development Process“. In: *International Journal of Software Engineering and Its Applications* 10.1 (2016), S. 165–178.
- [13] Virpi Roto, M. Obrist und Kaisa Väänänen-Vainio-Mattila. *User experience evaluation methods in academic and industrial contexts*. <https://static.googleusercontent.com/media/research.google.com/en//pubs/archive/37660.pdf>, abgerufen am 06.10.2024. 2009.
- [14] Thomas P. Moran Smart K. Card. *The Keystroke-Level Model for User Performance Time with Interactive Systems*. 1980. URL: https://www.researchgate.net/publication/220426122_The_Keystroke-Level_Model_for_User_Performance_Time_with_Interactive_Systems/citations.
- [15] Jaideep Srivastava u. a. „Web Usage Mining: Discovery and Applications of Usage Patterns from Web Data“. In: *ACM SIGKDD Explorations Newsletter* 1.2 (2000), S. 12–23.

-
- [16] Arnold P.O.S. Vermeeren u. a. *User Experience Evaluation Methods: Current State and Development Needs*. https://www.researchgate.net/publication/221517263_User_experience_evaluation_do_you_know_which_method_to_use, accessed 21.10.2024. 2010.