



CLOCKWISE

PROJECT ORIENTED TIME MANAGEMENT APPLICATION

Software-Entwicklungspraktikum (SEP)
Sommersemester 2025

Pflichtenheft

Auftraggeber
Technische Universität Braunschweig
Peter L. Reichertz Institute for Medical Informatics
Prof. Dr. Thomas M. Deserno
Mühlenpfordtstraße 23
38106 Braunschweig

Betreuer: Corinna Thoben

Auftragnehmer:

Name	E-Mail-Adresse
Joud Mawad	j.mawad@tu-braunschweig.de
Laden Zeynep Erkenci	l.erkenci@tu-braunschweig.de
Sophie Gebauer	sophie.gebauer@tu-braunschweig.de
Luis Jair Gutierrez Pacheco	l.gutierrez-pacheco@tu-braunschweig.de
Chantal Ebben	c.ebben@tu-braunschweig.de
Merle Lüer	m.lueer@tu-braunschweig.de
Moetez Belleh Hosni	m.hosni@tu-braunschweig.de

Braunschweig, 29. Mai 2025

Bearbeiterübersicht

Kapitel	Autoren	Kommentare
1	Laden Zeynep Erkenci	
1.1	Laden Zeynep Erkenci	
1.2	Laden Zeynep Erkenci	
2	Chantal Ebben	
2.1	Chantal Ebben	
2.2	Chantal Ebben	
2.3	Chantal Ebben	
3	Sophie Gebauer	
3.1	Sophie Gebauer	
3.2	Sophie Gebauer	
4	Luis Jair Gutierrez Pacheco	
5	Moetez Belleh Hosni	
6	Merle Lüer	
6.1	Merle Lüer	
6.2	Merle Lüer	
6.3	Merle Lüer	
7	Joud Mawad	
8	Chantal Ebben	
8.1	Chantal Ebben	
8.2	Chantal Ebben	
8.3	Chantal Ebben	
9	Alle Teammitglieder	

Inhaltsverzeichnis

1 Zielbestimmung	5
1.1 Musskriterien	5
1.2 Sollkriterien	6
1.3 Kannkriterien	7
1.4 Abgrenzungskriterien	7
2 Produkteinsatz	8
2.1 Anwendungsbereiche	8
2.2 Zielgruppen	9
2.3 Betriebsbedingungen	9
3 Produktübersicht	11
3.1 Use-Case-Diagramm: ClockWise Hauptfunktionen	11
3.2 Aktivitätsdiagramm: Zeiterfassung starten und stoppen	12
4 Produktfunktionen	15
5 Produktdaten	28
6 Nichtfunktionale Anforderungen	30
6.1 Funktionalität	30
6.2 Sicherheit	30
6.3 Benutzbarkeit	31
6.4 Änderbarkeit	32
6.5 Qualitätsanforderungen	32
7 Benutzeroberfläche/Schnittstellen	34
8 Technische Produktumgebung	40
8.1 Software	40
8.2 Hardware	40
8.3 Produktschnittstellen	41
9 Glossar	42

Abbildungsverzeichnis

3.1	Use-Case-Diagramm der Anwendung ClockWise	12
3.2	Zweites Use-Case-Diagramm der Anwendung ClockWise	13
3.3	Aktivitätsdiagramm: Zeiterfassung starten und stoppen	14
7.1	Login $\langle UI10 \rangle$	35
7.2	Dashboard $\langle UI20 \rangle$	36
7.3	Projects $\langle UI30 \rangle$	36
7.4	Time Tracking $\langle UI40 \rangle$	37
7.5	Analysis $\langle UI50 \rangle$	37
7.6	Team Management $\langle UI60 \rangle$	38
7.7	Kalenderansicht $\langle UI70 \rangle$	39

1 Zielbestimmung

Die zunehmende Komplexität von Studium, Beruf und Alltag stellt hohe Anforderungen an das persönliche Zeitmanagement. Nutzer:in müssen ihre Lern- und Arbeitszeiten effektiv organisieren und zielgerichtet einsetzen. Die Motivation zur Entwicklung der Anwendung ClockWise liegt im Bedarf nach einem intuitiven Werkzeug zur strukturierten Erfassung und Analyse von Zeiteinträgen.

Ziel des Projekts ist die Entwicklung einer webbasierten Anwendung, die Einzelpersonen und Teams bei der Planung, Dokumentation und Auswertung von Lern- und Arbeitszeiten unterstützt. Zeiträume lassen sich über eine Start-/Stopp-Funktion oder manuelle Eingaben flexibel erfassen und einzelnen Projekten zuordnen. Ergänzend können individuelle Lernziele definiert und der Fortschritt visuell dargestellt werden - etwa über eine Kalenderansicht oder Fortschrittsbalken.

ClockWise bietet darüber hinaus grafische Auswertungen auf Tages-, Wochen- oder Monatsebene und ermöglicht den Export erfasster Zeitdaten zur externen Nutzung.

Insgesamt unterstützt ClockWise eine effiziente Selbstorganisation und fördert modernes Zeitmanagement durch eine klar strukturierte, benutzerfreundliche Oberfläche - sowohl für Einzel- als auch für Gruppenarbeit.

1.1 Musskriterien

Muss-Kriterien sind Anforderungen, die zwingend erfüllt sein müssen, damit das System funktionsfähig ist und das Projektziel erreicht wird.

⟨RM1⟩ Das System muss eine Registrierung und Authentifizierung mit Passwortänderung und -wiederherstellung ermöglichen.

⟨RM2⟩ Das System muss eine Zeiterfassung mit Start-/Stopp-Funktion für Lern- und Arbeitszeiten ermöglichen.

⟨RM3⟩ Nutzer:in müssen Zeiteinträge manuell erfassen und nachträglich bearbeiten können.

⟨RM4⟩ Nutzer:in müssen neue Projekte anlegen, bearbeiten und löschen können. Der Typ des Projekts (Solo oder Team) muss festgelegt werden.

⟨RM5⟩ Jeder Zeiteintrag muss eindeutig einem Task zugeordnet sein. Ein Task kann mehrere

Zeiteinträge enthalten. Tasks wiederum sind immer einem übergeordneten Projekt zugeordnet. Die Zeiterfassung erfolgt ausschließlich auf Task-Ebene.

⟨RM6⟩ Das System muss es ermöglichen, pro Projekt mehrere Tasks anzulegen und zu verwalten.

⟨RM7⟩ Es muss eine Kalenderansicht integriert sein, die alle erfassten Zeiten übersichtlich darstellt.

⟨RM8⟩ Das System muss Fortschrittsdiagramme bereitstellen, die den Bearbeitungsstand pro Projekt visuell in Prozent darstellen.

⟨RM9⟩ Das System muss einen visuellen Ist-Soll-Vergleich der prozentualen Zeitverteilung über mehrere Projekte hinweg ermöglichen, z. B. durch nebeneinander stehende gestapelte Balkendiagramme mit farblicher Zuordnung und Prozentangaben.

⟨RM10⟩ Das System muss es ermöglichen, Teamprojekte mit mehreren Mitgliedern eines bestehenden Teams zu teilen. Jedes Mitglied muss ein registrierter Benutzer sein.

⟨RM11⟩ In Teamprojekten mit mehreren Mitgliedern müssen Tasks einzelnen Mitgliedern zugewiesen werden können.

⟨RM12⟩ Zeitdaten müssen exportierbar sein (z. B. im PDF- oder CSV-Format).

⟨RM13⟩ Die Benutzeroberfläche muss mindestens in englischer Sprache verfügbar sein.

⟨RM14⟩ Die Anwendung muss plattformunabhängig in Google Chrome ab Version 136.0.7103.93 lauffähig sein.

⟨RM15⟩ Die erfassten Zeitdaten müssen sicher und dauerhaft in einer Datenbank (SQLite) gespeichert werden.

1.2 Sollkriterien

Soll-Kriterien sind wünschenswerte Funktionen, die die Benutzerfreundlichkeit und Qualität erhöhen, jedoch bei begrenzten Ressourcen verzichtbar sind.

⟨RS1⟩ Tasks sollen z. B. als „Prüfung“, „Hausarbeit“ oder „Wiederholung“ kategorisiert werden.

⟨RS2⟩ Die Anwendung soll eine Such- und Filterfunktion bereitstellen, mit der Projekte und Tasks nach Eigenschaften wie Name, Tags, Status, Fälligkeitsdatum gefiltert werden können.

⟨RS3⟩ Systeminterne Benachrichtigungen sollen durchsuchbar und filterbar sein (z. B. nach Benachrichtigungstyp wie „Task-Zuweisung“, Erstellungsdatum oder Lesestatus).

⟨RS4⟩ Das System soll bei Teamprojekten systeminterne Benachrichtigungen senden können (z. B. bei Task-Zuweisungen, Teambeitritten oder neuen Kommentaren).

⟨RS5⟩ Das System soll auf Basis der eingetragenen Leistungspunkte pro Projekt automatisch prozentuale Zielwerte für den wöchentlichen Lernaufwand vorschlagen.

⟨RS6⟩ Nutzer:in sollen sich über OAuth (z.B. Google-Account) anmelden.

⟨RS7⟩ Das Frontend soll aktive und inaktive Projekte klar unterscheiden.

⟨RS8⟩ Die Anwendung soll responsiv sein und auf verschiedenen Endgeräten funktionieren (z. B. Desktop, Tablet, Smartphone).

⟨RS9⟩ Die Frequenz der Benachrichtigungen zum Ist-Soll-Vergleich (z. B. täglich, wöchentlich) soll in den Einstellungen konfigurierbar sein.

1.3 Kannkriterien

Kann-Kriterien sind optionale Erweiterungen, die bei ausreichenden Ressourcen umgesetzt werden können und zusätzlichen Nutzen bieten.

⟨RC1⟩ Das System kann eine Farbanpassung für Projekte, Kalender und Diagramme ermöglichen.

⟨RC2⟩ Das System kann ein automatisches Backup der Zeitdaten bereitstellen.

⟨RC3⟩ Das System kann Balkendiagramme zur Auswertung des Zeitaufwands pro Projekt (täglich, wöchentlich, monatlich) bereitstellen.

⟨RC4⟩ Das System kann zeitzonenfähig sein, um Zeiten ortsunabhängig korrekt darzustellen.

⟨RC5⟩ Die Anwendung kann auch in deutscher Sprache verfügbar sein.

⟨RC6⟩ Das System kann wöchentliche Benachrichtigungen zum Ist-Soll-Vergleich senden, wenn individuell festgelegte Zeitziele für Projekte deutlich unterschritten, erreicht oder überschritten werden.

⟨RC7⟩ In Teamprojekten kann sichtbar gemacht werden, welche Mitglieder an welchen Tasks arbeiten und wie viel Zeit sie investiert haben.

⟨RC8⟩ Die Benutzeroberfläche kann eine Umschaltfunktion zwischen Dark- und Light-Mode bereitstellen.

1.4 Abgrenzungskriterien

Abgrenzungskriterien beschreiben bewusst Anforderungen, die nicht Teil des Projekts sind. Sie dienen der klaren Eingrenzung des Projektumfangs.

⟨RW1⟩ Offline-Nutzung wird nicht unterstützt; eine Internetverbindung ist erforderlich.

⟨RW2⟩ Integration in Kalenderdienste wie Google Calendar oder Outlook ist nicht vorgesehen.

⟨RW3⟩ Mobile Apps für Android/iOS werden in dieser Projektphase nicht entwickelt.

⟨RW4⟩ Automatische Zeiterfassung über Desktop-Aktivität oder ähnliche Hintergrundprozesse wird nicht unterstützt.

2 Produkteinsatz

Im Folgenden wird beschrieben, in welchem Kontext der Einsatz von ClockWise vorgesehen ist. Dabei werden typische Anwendungsbereiche und Zielgruppen beschrieben sowie die Betriebsbedingungen aufgeführt, unter denen eine zuverlässige Nutzung gewährleistet werden kann.

2.1 Anwendungsbereiche

ClockWise ist als plattformunabhängige Webanwendung konzipiert, die der digitalen Zeiterfassung und -planung dient. Die Anwendung ist nicht an einen speziellen institutionellen Kontext gebunden, sondern lässt sich flexibel sowohl im professionellen als auch im privaten Umfeld sowie im Bildungssektor einsetzen.

Im beruflichen und im Bildungsbereich eignet sich ClockWise insbesondere für projektbezogene Teamarbeit – beispielsweise in Unternehmen, wissenschaftlichen Einrichtungen oder studentischen Gruppen. Dabei unterstützt die Anwendung die strukturierte Dokumentation und Auswertung investierter Arbeitszeit pro Projekt und Teammitglied. Auch im Rahmen von Remote-Arbeitsmodellen bietet die Anwendung eine sinnvolle Ergänzung zur Selbstorganisation und Leistungsdokumentation.

Darüber hinaus ist ClockWise explizit auch für den privaten Alltag konzipiert. Studierende und Schüler:innen können ihre Zeit für Hausarbeiten und Prüfungsvorbereitungen erfassen und planen. Ebenso können Privatpersonen die Anwendung zur optimalen Strukturierung von Freizeitvorhaben nutzen – etwa zur besseren Selbstorganisation oder zur Förderung einer ausgewogenen Work-Life-Balance.

ClockWise ist nicht als rechtlich anerkannte Arbeitszeiterfassung klassifiziert. Dementsprechend ist der Einsatz als abrechnungsfähiges, rechtlich verbindliches Arbeitszeiterfassungssystem im Sinne des Arbeitszeitgesetzes ausdrücklich nicht vorgesehen.

2.2 Zielgruppen

Die Anwendung richtet sich an eine breite Nutzergruppe:

- Studierende und Schüler:innen, die ihre Lernzeiten strukturieren, einteilen und überwachen möchten.
- Projektteams – z. B. in Hochschulen oder Unternehmen –, die Transparenz über Zeitinvestitionen schaffen und Aufgabenverteilungen nachvollziehen wollen.
- Freiberufler:innen und Selbstständige, die ihre Zeit flexibel planen und rückblickend auswerten möchten.
- Privatpersonen, die persönliche Projekte oder Freizeitgestaltung effizient organisieren wollen.

ClockWise ist so gestaltet, dass auch technisch weniger versierte Nutzer:innen die Anwendung ohne umfangreiche Vorkenntnisse bedienen können. Die Benutzeroberfläche ist intuitiv, browserbasiert und somit weitgehend unabhängig vom eingesetzten Betriebssystem und der verwendeten Hardware. Da die Anwendung kostenlos verfügbar ist, entstehen darüber hinaus keine finanziellen Zugangshürden. Diese Merkmale ermöglichen es, dass ClockWise von der zuvor beschriebenen, breit aufgestellten Nutzergruppe gleichermaßen genutzt werden kann – unabhängig von technischer Erfahrung, Ausstattung oder finanziellen Mitteln.

2.3 Betriebsbedingungen

Da es sich um eine Webanwendung handelt, sind für den Betrieb keine besonderen physikalischen Voraussetzungen notwendig. Die Nutzung ist mit allen gängigen internetfähigen Endgeräten wie PCs, Laptops, Tablets oder Smartphones möglich.

Vorausgesetzt wird lediglich ein moderner Webbrowser sowie eine stabile Internetverbindung. Die Anwendung ist responsiv gestaltet und passt sich automatisch an unterschiedliche Bildschirmgrößen an. Eine Installation lokaler Software ist nicht erforderlich.

Die Nutzung erfolgt in der Regel unbeaufsichtigt und asynchron. Folglich ist keine permanente Interaktion mit einem Systemadministrator oder Support erforderlich. Wartung, Updates und Erweiterungen können zentral serverseitig bereitgestellt werden, wodurch die Anwendung jederzeit einsatzbereit und aktuell gehalten werden kann. Manuelle Downloads von Neuerungen durch die Nutzer:innen sind nicht notwendig.

Die Bereitstellung der Anwendung erfolgt zunächst lokal auf dem jeweiligen Endgerät. Ein öffentlich zugängliches Deployment über einen zentral gehosteten VM-Server ist ebenfalls umsetzbar.

Nutzer:innen können dann plattformunabhängig über einen modernen Webbrowser mittels HTTPS auf die Webanwendung zugreifen, ohne eine zusätzliche Softwareinstallation vorzunehmen. Hierfür kann eine zentrale Bereitstellung über die Instituts-Infrastruktur des Peter L. Reichertz Instituts für Medizinische Informatik erfolgen, inklusive einer HTTPS-Absicherung sowie einem angebundenen SMTP-Dienst für den E-Mail-Versand.

3 Produktübersicht

Dieses Kapitel dient der Übersicht über die Funktionen der Webanwendung ClockWise. Interaktionen der Benutzer:innen mit dem System werden im Use-Case-Diagramm dargestellt. Die Abläufe einzelner komplexer Funktionen werden ergänzend im Aktivitätsdiagramm erläutert. Jedes Diagramm wird durch erklärenden Text begleitet.

3.1 Use-Case-Diagramm: ClockWise Hauptfunktionen

Die Hauptfunktion der Anwendung ClockWise besteht darin, die projektbezogene Zeiterfassung und -auswertung effizient zu unterstützen. Nutzer:innen können durch verschiedene Interaktionen und Befehle im System – wie im Use-Case-Diagramm in Abbildung 3.1 und 3.2 dargestellt – ihre Arbeitszeiten verwalten und auswerten.

Die Anwendung stellt umfangreiche Funktionen zur Verfügung, die die Organisation und Durchführung von Projekten erleichtern. Nutzer:innen können ihre Arbeitszeiten direkt erfassen sowie bei Bedarf nachträglich bearbeiten oder ergänzen. Projekte lassen sich erstellen und bearbeiten, wobei jeder Aufgabe (Task) ein übergeordnetes Projekt zugeordnet wird. Die Verwaltung von Tasks ermöglicht eine feingranulare Steuerung der Arbeitsinhalte.

Zur Auswertung bietet ClockWise grafische Statistiken, einen visuellen Ist-Soll-Vergleich sowie eine Kalenderansicht, die alle erfassten Zeiten übersichtlich darstellt. Zudem können sämtliche Zeitdaten im PDF- oder CSV-Format exportiert werden.

Die integrierte Teamfunktion erlaubt es, Projekte gemeinsam mit anderen registrierten Nutzer:innen zu bearbeiten. Für Nutzer:innen mit der Rolle "Teamleitung" steht darüber hinaus eine Teamverwaltungsfunktion zur Verfügung. Diese ermöglicht es, Teammitglieder hinzuzufügen oder zu entfernen, Rollen zuzuweisen und die Arbeitszeiten des gesamten Teams einzusehen. Die Teamleitung kann darüber hinaus Aufgaben Teammitgliedern zuweisen. Dadurch wird eine zentrale Koordination und Kontrolle innerhalb des Projektteams unterstützt.

Eine Benutzerregistrierung mit Authentifizierungsfunktion sorgt dafür, dass alle personenbezogenen Daten sicher gespeichert und verwaltet werden.

Alle Use Cases sind darauf ausgerichtet, die Zeiterfassung und Analyse innerhalb projektbezogener Arbeitsabläufe strukturiert, transparent und benutzerfreundlich zu gestalten.

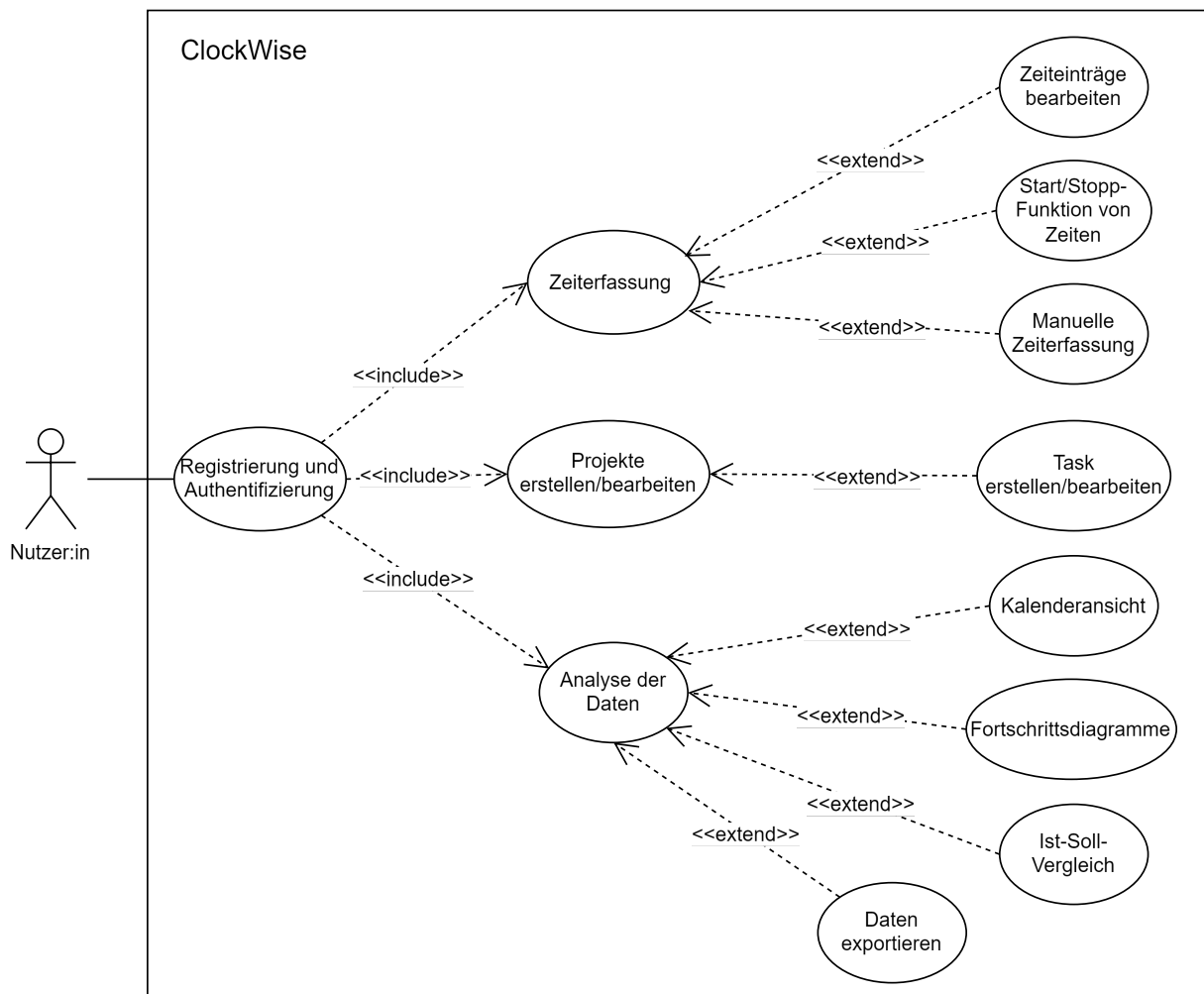


Abbildung 3.1: Use-Case-Diagramm der Anwendung ClockWise

3.2 Aktivitätsdiagramm: Zeiterfassung starten und stoppen

Ein zentraler Use-Case ist die Verwaltung der Zeiterfassung. Das Aktivitätsdiagramm auf der folgenden Seite veranschaulicht den Ablauf der Zeiterfassung innerhalb der Anwendung.

Die Nutzer:in meldet sich über das Webinterface an und gelangt anschließend zur Übersichtsseite, auf der ein Projekt ausgewählt wird. Dort kann die Zeiterfassung über einen Start-Button begonnen werden. Während der Erfassung besteht die Möglichkeit, den Vorgang zu pausieren oder zu stoppen. Beim Stoppen wird der entsprechende Zeiteintrag automatisch gespeichert.

Die Abbildung 3.3 zeigt die möglichen Aktionen während der Zeiterfassung und markiert den Abschluss des Use Cases, woraufhin die Nutzer:in weitere Zeiten erfassen oder zu anderen Funktionen der Anwendung wechseln kann.

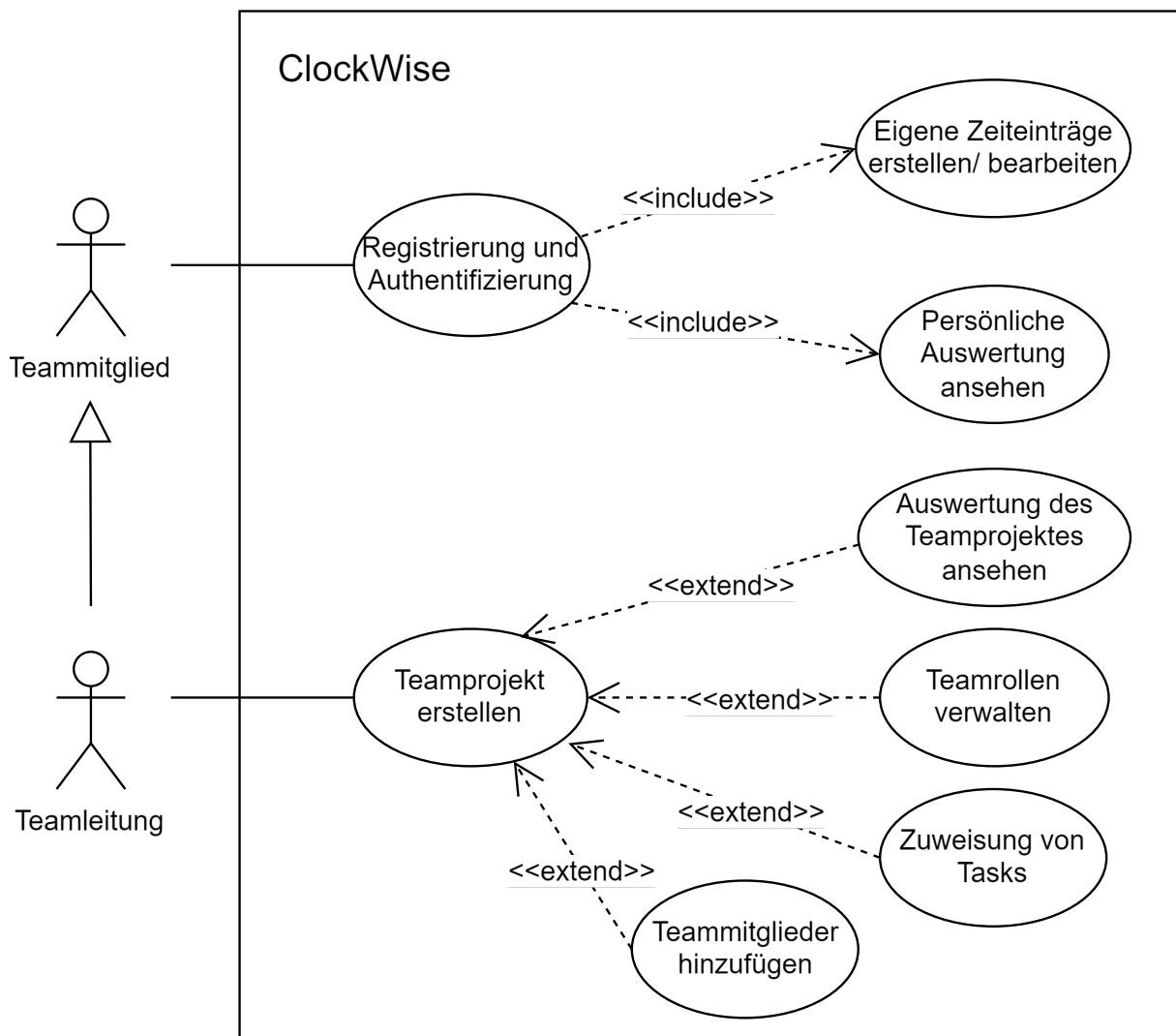


Abbildung 3.2: Zweites Use-Case-Diagramm der Anwendung ClockWise

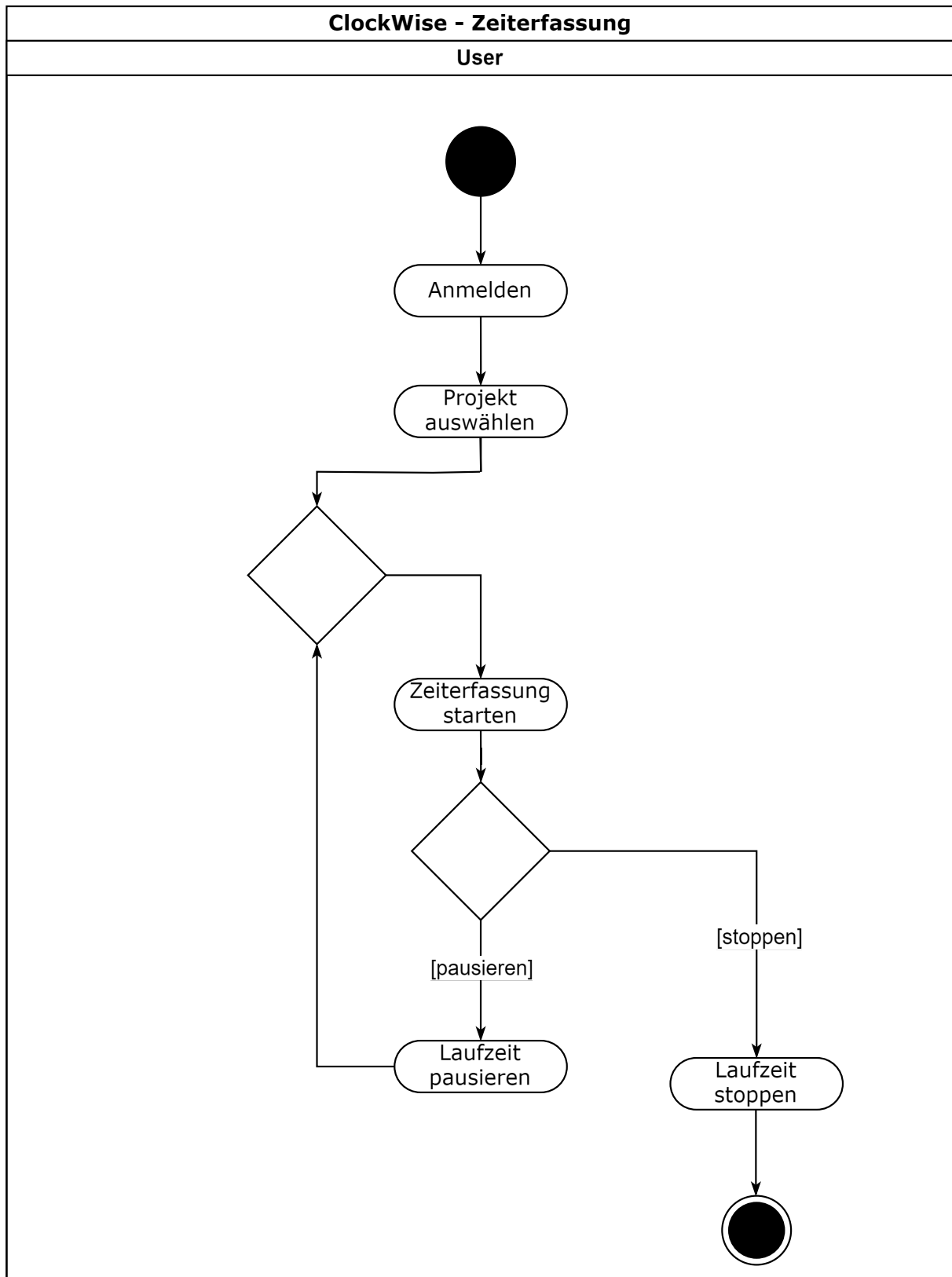


Abbildung 3.3: Aktivitätsdiagramm: Zeiterfassung starten und stoppen

4 Produktfunktionen

In diesem Kapitel werden die zentralen Funktionen der Webanwendung detailliert beschrieben.

Registrierung $\langle F10 \rangle$

Anwendungsfall: Erstellung eines Nutzerkontos zur erstmaligen Nutzung der Anwendung ClockWise.

Anforderung: $\langle RM1 \rangle$

Ziel: Nutzer:innen können ein persönliches Konto erstellen, um ClockWise im vollen Funktionsumfang nutzen zu können.

Vorbedingung: Die Nutzer:innen haben noch kein Konto und rufen die Registrierungsseite über die Webanwendung auf.

Nachbedingung Erfolg: Das neue Konto wird erfolgreich erstellt, verschlüsselt in der Datenbank gespeichert und die Nutzer:in wird auf das Dashboard weitergeleitet.

Nachbedingung Fehlschlag: Bei ungültigen oder unvollständigen Eingaben (z. B. bereits verwendete E-Mail, schwaches Passwort) wird eine Fehlermeldung angezeigt, z. B. "This email address is already in use."

Akteure: Neue Nutzer:innen

Auslösendes Ereignis: Die Nutzer:in wählt auf der Startseite der Anwendung die Option „Registrieren“.

Beschreibung:

1. Öffnen der Startseite von ClockWise.
2. Auswahl der Option „Registrieren“.
3. Eingabe der persönlichen Daten (z. B. Name, E-Mail-Adresse, Passwort).
4. Validierung der Eingaben.
5. Speicherung der Nutzerdaten in der Datenbank.
6. Weiterleitung zum persönlichen Dashboard.

Erweiterung: Registrierung über OAuth (z. B. Google-Account) als alternative Methode zur Eingabe von Zugangsdaten. (Erweiterung zu Schritt 3)

Login ⟨F20⟩

Anwendungsfall: Anmeldung mit bestehenden Zugangsdaten zur Nutzung der ClockWise Anwendung.

Anforderung: <RM1>, <RS6>

Ziel: Nutzer:innen können sich sicher in der Anwendung anmelden, um auf persönliche Projekt- und Zeitdaten zuzugreifen.

Vorbedingung: Die Nutzer:innen verfügen über ein registriertes Konto und rufen die Login Seite auf.

Nachbedingung Erfolg: Der Zugriff auf das persönliche Dashboard wird gewährt; Nutzer:innen sind authentifiziert.

Nachbedingung Fehlschlag: Bei falschen Zugangsdaten oder fehlenden Pflichtangaben wird eine Fehlermeldung angezeigt und der Zugriff verweigert, z. B. "Password is incorrect."

Akteure: Registrierte Nutzer:innen

Auslösendes Ereignis: Nutzer:in öffnet die Login-Seite und gibt die Zugangsdaten ein.

Beschreibung:

1. Öffnen der ClockWise Anwendung im Webbrowser.
2. Auswahl der Option „Login“.
3. Eingabe von E-Mail-Adresse und Passwort.
4. Validierung der Zugangsdaten.
5. Bei erfolgreicher Authentifizierung: Weiterleitung zum Dashboard.

Erweiterung: Möglichkeit zur Passwort-Zurücksetzung über E-Mail bei vergessenem Passwort. (Erweiterung zu Schritt 3)

Alternativen: Anmeldung über OAuth (z. B. Google-Login). (Alternative zu Schritt 3: Nutzung externer Login-Daten)

Zeiterfassung (Start/Stop) $\langle F30 \rangle$

Anwendungsfall: Erfassen von Arbeitszeiten durch Starten, Pausieren und Stoppen eines Timers.

Anforderung: $\langle RM2 \rangle$, $\langle RM5 \rangle$

Ziel: Nutzer:innen können ihre Arbeitszeit in Echtzeit erfassen und verwalten.

Vorbedingung: Die Nutzer:in ist eingeloggt.

Nachbedingung Erfolg: Die erfasste Zeit wird korrekt gespeichert und dem Projekt bzw. Task zugeordnet.

Nachbedingung Fehlschlag: Bei Verbindungsproblemen oder falscher Bedienung erfolgt eine Fehlermeldung.

Akteure: Nutzer:in

Auslösendes Ereignis: Nutzer:in startet die Zeiterfassung über die Benutzeroberfläche.

Beschreibung:

1. Anmeldung in der Anwendung.
2. Auswahl einer Task (optional).
3. Starten der Zeiterfassung.
4. Möglichkeit zum Pausieren oder Stoppen der Zeiterfassung.
5. Speicherung der erfassten Zeit nach dem Stoppen.
6. Falls kein bestehender Task gewählt wurde, wird automatisch ein neuer Task erzeugt („Unassigned Task“), der später einem Projekt zugeordnet werden kann.

Alternativen: Alternativ können Nutzer:innen die Zeit manuell erfassen, z. B. wenn die Live Zeiterfassung nicht geeignet ist. In diesem Fall geben sie Start und Endzeit manuell ein, und der Eintrag wird direkt gespeichert. (Alternative zu Schritt 3)

Zeiteinträge bearbeiten $\langle F40 \rangle$

Anwendungsfall: Anpassung oder Korrektur von bereits erfassten Zeitdaten durch die Nutzer:innen.

Anforderung: $\langle RM2 \rangle \langle RM3 \rangle$

Ziel: Nutzer:innen können ihre erfassten Zeiten flexibel ändern, um eine korrekte und vollständige Dokumentation der Arbeitszeit sicherzustellen.

Vorbedingung: Nutzer:in ist eingeloggt und hat bereits bestehende Zeiteinträge im System.

Nachbedingung Erfolg: Die bearbeiteten Zeitdaten werden korrekt gespeichert und ersetzen oder ergänzen die vorherigen Einträge.

Nachbedingung Fehlschlag: Bei ungültigen Eingaben (wie negative Zeiten) oder technischen Fehlern wird eine entsprechende Fehlermeldung angezeigt, z. B. "Finish time must be after start time."

Akteure: Nutzer:in

Auslösendes Ereignis: Nutzer:in wählt in der Zeiteinträge-Übersicht einer bestehenden Task einen vorhandenen Zeiteintrag zur Bearbeitung aus oder erstellt einen neuen.

Beschreibung:

1. Öffnen der Übersicht der erfassten Zeiten.
2. Auswahl des zu bearbeitenden Eintrags oder Hinzufügen eines neuen.
3. Anpassung von Startzeit, Endzeit.
4. Validierung der neuen Eingaben.
5. Speicherung der Änderungen.

Erweiterung: Möglichkeit, mehrere Einträge gleichzeitig (Batch) zu bearbeiten oder Einträge zu löschen.

Alternativen: Wiederherstellung einer früheren Version des Eintrags (Undo-Funktion) könnte zukünftig integriert werden.

Projekte erstellen und bearbeiten $\langle F50 \rangle$

Anwendungsfall: Erstellung, Bearbeitung und Löschung von Projekten zur Strukturierung der Zeiterfassung.

Anforderung: $\langle RM4 \rangle$

Ziel: Nutzer:innen können ihre Arbeits und Lernzeiten strukturiert Projekten zuordnen, indem sie eigene Projekte verwalten und dabei den Projekttyp festlegen.

Vorbedingung: Die Nutzer:in ist angemeldet und befindet sich in der Projektübersicht.

Nachbedingung Erfolg: Das Projekt wird erfolgreich erstellt, geändert oder gelöscht und in der Datenbank gespeichert.

Nachbedingung Fehlschlag: Bei fehlenden Pflichtangaben oder ungültigen Eingaben erscheint eine Fehlermeldung, z. B. "Project name is required."

Akteure: Nutzer:in

Auslösendes Ereignis: Nutzer:in wählt in der Projektübersicht die Option „Create Project“ oder „Edit Project“.

Beschreibung:

1. Öffnen der Projektübersicht im Dashboard.
2. Auswahl der Option „New Project“ oder „Edit Project“.
3. Eingabe bzw. Anpassung der Projektdaten (Name, Beschreibung, Typ: Solo oder Team).
4. Validierung der Eingaben.
5. Speicherung des Projekts in der Datenbank oder Löschung bei Bedarf.

Erweiterung: Möglichkeit zur Farbzweisung für Projekte zur besseren visuellen Unterscheidung. (Erweiterung zu Schritt 3)

Tasks erstellen und bearbeiten ⟨F60⟩

Anwendungsfall: Verwaltung einzelner Arbeitseinheiten innerhalb eines Projekts in Form von Tasks.

Anforderung: <RM6>, <RM5>, <RM12>

Ziel: Nutzer:innen können für jedes Projekt beliebig viele Tasks erstellen, bearbeiten, löschen und ggf. Teammitgliedern zuweisen.

Vorbedingung: Die Nutzer:in ist angemeldet. Tasks können entweder direkt innerhalb eines Projekts oder nachträglich für zuvor nicht zugeordnete Tasks (Unassigned Tasks) bearbeitet werden..

Nachbedingung Erfolg: Tasks werden erfolgreich erstellt, aktualisiert oder gelöscht und korrekt dem Projekt zugeordnet (sofern zugewiesen).

Nachbedingung Fehlschlag: Wird kein Titel eingegeben, speichert das System den Task automatisch mit dem Titel „Untitled Task“. Bei anderen ungültigen Eingaben erscheint eine Fehlermeldung.

Akteure: Nutzer:in (Projektverantwortliche:r)

Auslösendes Ereignis: Die Nutzer:in öffnet die Projektübersicht oder filtert nach „Unassigned Tasks“ und wählt dort „Add Task“, „Edit Task“ oder „Delete Task“.

Beschreibung:

1. Öffnen der Projektdetailansicht oder Auswahl eines unzugeordneten Tasks über die Filterfunktion in Projektseite.
2. Auswahl der Option „Add Task“, „Edit Task“ oder „Delete Task“.
3. Eingabe bzw. Bearbeitung von Task-Details (z.B. Titel, Beschreibung, Kategorie, Fälligkeitsdatum).
4. Optionale Projektzuweisung für zuvor unzugeordnete Tasks.
5. Optionale Zuweisung des Tasks an ein Teammitglied (bei Teamprojekten).
6. Speichern oder Löschen des Tasks.

Erweiterung: Möglichkeit zur Priorisierung von Tasks zur besseren Planung. (Erweiterung zu Schritt 3)

Export von Zeitdaten $\langle F70 \rangle$

Anwendungsfall: Exportieren erfasster Zeitdaten zur externen Nutzung oder Archivierung.

Anforderung: $\langle RM12 \rangle$

Ziel: Nutzer:innen können ihre Zeitdaten als Datei exportieren, um diese außerhalb der Anwendung weiterzuverwenden.

Vorbedingung: Die Nutzer:in ist angemeldet und hat mindestens ein Projekt mit erfassten Zeiteinträgen.

Nachbedingung Erfolg: Die exportierten Daten werden erfolgreich im gewählten Format (CSV oder PDF) generiert und zum Herunterladen bereitgestellt.

Nachbedingung Fehlschlag: Bei fehlender Auswahl eines Projekts oder bei Serverproblemen erscheint eine Fehlermeldung, z. B. "No project selected for export."

Akteure: Nutzer:in

Auslösendes Ereignis: Nutzer:in wählt in der Projektansicht die Option „Exportieren“.

Beschreibung:

1. Öffnen der Projektansicht.
2. Auswahl des gewünschten Projekts und Zeitraums.
3. Auswahl des Exportformats (CSV oder PDF).
4. Auslösen des Exports.
5. Download der generierten Datei.

Erweiterung: Möglichkeit zur Filterung der zu exportierenden Daten nach Zeitraum, Projekt oder Task. (Erweiterung zu Schritt 2)

Kalenderansicht $\langle F80 \rangle$

Anwendungsfall: Darstellung aller erfassten Zeitdaten in einer übersichtlichen Kalenderansicht.

Anforderung: $\langle RM7 \rangle$

Ziel: Nutzer:innen können ihre dokumentierten Arbeits- und Lernzeiten innerhalb eines Kalenders visuell einsehen.

Vorbedingung: Die Nutzer:in ist angemeldet und hat erfasste Daten (z. B. Tasks oder Zeiteinträge) in einem oder mehreren Projekten.

Nachbedingung Erfolg: Die relevanten Daten werden korrekt in der Monatsansicht angezeigt und lassen sich optional nach Projekt und Zeitraum filtern.

Nachbedingung Fehlschlag: Bei fehlenden Einträgen erscheint eine Leermeldung („Keine Daten verfügbar“), bei Serverproblemen eine Fehlermeldung.

Akteure: Nutzer:in

Auslösendes Ereignis: Nutzer:in wählt im Dashboard oder im Projektbereich die Kalenderansicht aus.

Beschreibung:

1. Öffnen des Dashboards oder Analysebereichs.
2. Auswahl der Option „Kalenderansicht“.
3. Kalender wird in der Monatsansicht angezeigt.
4. Daten werden visuell nach Datum und Projekt dargestellt.

Erweiterung: Möglichkeit zur Filterung der Ansicht nach Projekt oder Zeitraum. (Erweiterung zu Schritt 4)

Alternativen: Wechsel in die Jahresansicht oder Export der Kalenderdaten als Bild oder Integration in einen Bericht.

Fortschritts- und Vergleichsdiagramme $\langle F90 \rangle$

Anwendungsfall: Visualisierung des Bearbeitungsstands und des Ist-Soll-Vergleichs zur besseren Selbstkontrolle.

Anforderung: $\langle RM8 \rangle$, $\langle RM9 \rangle$

Ziel: Nutzer:innen können ihre zeitliche Zielerreichung und den Fortschritt innerhalb einzelner Projekte sowie im Vergleich zwischen mehreren Projekten visuell nachvollziehen.

Vorbedingung: Die Nutzer:in ist angemeldet und hat für mindestens ein Projekt Zeiteinträge und Soll-Zeitziele festgelegt.

Nachbedingung Erfolg: Die entsprechenden Diagramme werden korrekt berechnet und angezeigt, inkl. prozentualer Werte und farblicher Markierungen.

Nachbedingung Fehlschlag: Bei fehlenden Daten wird ein entsprechender Hinweis angezeigt („No target values available“); bei Serverfehlern erscheint eine Fehlermeldung.

Akteure: Nutzer:in

Auslösendes Ereignis: Nutzer:in öffnet die Analyse- oder Statistikansicht der Anwendung.

Beschreibung:

1. Öffnen des Dashboards oder Analysebereichs.
2. Auswahl eines Projekts oder Vergleich mehrerer Projekte.
3. Darstellung des Fortschrittsdiagramms pro Projekt (z. B. Balken mit).
4. Darstellung des Ist-Soll-Vergleichs über Projekte hinweg (z. B. gestapelte Balkendiagramme mit Farbcode).

Erweiterung: Das System kann Benachrichtigungen erzeugen, wenn der Ist-Soll-Vergleich signifikante Abweichungen zeigt. Diese Funktion kann wöchentlich automatisch ausgeführt werden. $\langle RS9, RC6 \rangle$

Teamverwaltung $\langle F100 \rangle$

Anwendungsfall: Verwaltung von Teamprojekten und deren Mitgliedern, einschließlich Zuweisung von Aufgaben an bestimmte Personen.

Anforderung: $\langle RM10 \rangle$, $\langle RM11 \rangle$

Ziel: Nutzer:innen können Teamprojekte erstellen und mit anderen registrierten Mitgliedern teilen. Innerhalb dieser Projekte können Aufgaben einzelnen Teammitgliedern zugewiesen werden.

Vorbedingung: Die Nutzer:in ist angemeldet und hat ein Projekt mit Typ „Team“ erstellt oder betreten.

Nachbedingung Erfolg: Das Teamprojekt wurde erfolgreich erstellt oder geteilt, Mitglieder wurden hinzugefügt oder entfernt, und Aufgaben wurden bestimmten Personen zugewiesen.

Nachbedingung Fehlschlag: Bei fehlenden Berechtigungen oder ungültigen Benutzern erscheint eine Fehlermeldung, z. B. "You do not have permission to manage this team."

Akteure: Projektverantwortliche:r (Admin), Teammitglieder

Auslösendes Ereignis: Projektverantwortliche:r öffnet ein Teamprojekt und verwaltet Mitglieder oder weist Aufgaben zu.

Beschreibung:

1. Erstellung eines neuen Projekts mit dem Typ „Team“.
2. Einladung registrierter Nutzer:innen zur Teilnahme am Projekt.
3. Annahme oder Ablehnung der Einladung durch eingeladene Mitglieder.
4. Anzeige aller Teammitglieder und deren Rollen (z. B. Admin, Mitglied).
5. Zuweisung einzelner Tasks an bestimmte Teammitglieder.

Erweiterung: Möglichkeit zur Rollenzuweisung für Mitglieder innerhalb eines Teamprojekts. (Erweiterung zu Schritt 4)

Benutzeroberfläche – Sprache $\langle F110 \rangle$

Anwendungsfall: Anpassung der Benutzeroberfläche an persönliche Spracheinstellungen.

Anforderung: $\langle RM13 \rangle$

Ziel: Nutzer:innen können die Sprache der Benutzeroberfläche auf Englisch umstellen.

Vorbedingung: Die Nutzer:in ist angemeldet oder befindet sich auf der Startseite der Webanwendung.

Nachbedingung Erfolg: Die gewählte Sprache wird erfolgreich übernommen und im Nutzerprofil gespeichert.

Nachbedingung Fehlschlag: Bei technischen Problemen erfolgt eine Fehlermeldung oder die Standardsprache bleibt erhalten.

Akteure: Nutzer:in

Auslösendes Ereignis: Nutzer:in öffnet die Einstellungen oder klickt auf eine sichtbare Sprachumschaltoption.

Beschreibung:

1. Öffnen der Einstellungen oder Nutzung eines sichtbaren Sprachumschalters.
2. Auswahl der Sprache: Englisch (Standard).
3. Änderungen werden übernommen und im Profil gespeichert.

Erweiterung: Weitere Sprachen wie Deutsch oder Spanisch könnten zukünftig integriert werden.

Plattformkompatibilität (Chrome) $\langle F120 \rangle$

Anwendungsfall: Nutzung der Anwendung in einem Webbrowser.

Anforderung: $\langle RM14 \rangle$

Ziel: Die Anwendung ist plattformunabhängig nutzbar, indem sie vollständig mit der aktuellen Version von Google Chrome kompatibel ist.

Vorbedingung: Nutzer:in verwendet Google Chrome in der jeweils aktuellen Version.

Nachbedingung Erfolg: Die Anwendung läuft ohne Funktions- oder Darstellungsfehler im Chrome-Browser unter verschiedenen Betriebssystemen (z.B. Windows, macOS, Linux).

Nachbedingung Fehlschlag: Bei Nutzung veralteter Versionen oder inkompatibler Browser können Einschränkungen auftreten.

Akteure: Nutzer:in

Auslösendes Ereignis: Öffnen der Anwendung im Chrome-Browser.

Beschreibung:

1. Nutzer:in ruft die Webanwendung im Chrome-Browser auf.
2. Die Anwendung wird plattformunabhängig gerendert.
3. Alle Funktionen sind uneingeschränkt bedienbar.

Erweiterung: Unterstützung weiterer Browser wie Firefox, Safari oder Edge. (Erweiterung zu Schritt 1)

Datenpersistenz in SQLite $\langle F130 \rangle$

Anwendungsfall: Dauerhafte Speicherung aller erfassten Zeitdaten in einer zentralen Datenbank.

Anforderung: $\langle RM15 \rangle$

Ziel: Alle relevanten Zeitdaten (z. B. Zeiteinträge, Projekte, Tasks) werden zuverlässig gespeichert und sind dauerhaft verfügbar.

Vorbedingung: Die Anwendung ist funktionsfähig und korrekt mit der Datenbank verbunden.

Nachbedingung Erfolg: Alle neu erfassten oder geänderten Daten werden korrekt in der SQLite-Datenbank gespeichert und sind bei erneutem Zugriff abrufbar.

Nachbedingung Fehlschlag: Bei fehlerhafter Verbindung oder beschädigter Datenbank erfolgt eine Fehlermeldung und die Daten werden nicht gespeichert, z. B. "Failed to save data. Please check your connection."

Akteure: System (Backend)

Auslösendes Ereignis: Jede Aktion, die Daten erzeugt oder verändert (z. B. Zeiterfassung, Projektanlage).

Beschreibung:

1. Nutzer:in führt eine Aktion aus, die Daten erzeugt oder verändert.
2. Die Daten werden serverseitig validiert.
3. Persistente Speicherung in einer SQLite-Datenbank erfolgt automatisch.
4. Die gespeicherten Daten sind bei zukünftigen Zugriffen verfügbar.

Erweiterung: Möglichkeit zur regelmäßigen Erstellung automatischer Backups. (Erweiterung zu Schritt 3)

5 Produktdaten

Die dauerhaft zu speichernden Daten sollen aus Sicht der Nutzer:innen klar und strukturiert dargestellt werden. Eine formale Beschreibung dient dabei der besseren Nachvollziehbarkeit und Genauigkeit.

Hinweis: In ClockWise erfolgt die Zeiterfassung ausschließlich auf *Task-Ebene*. Tasks gehören immer zu einem *Projekt* (oder einer Lehrveranstaltung). Projekte wiederum sind einem *Team* oder einer Einzelperson zugeordnet. Diese Datenstruktur erlaubt eine klare Zuordnung und Analyse der Arbeitszeit.

Nutzerdaten $\langle D10 \rangle$

Daten der Benutzer:innen (max. 1000):

- Benutzer-ID (eindeutiger Schlüssel),
- Name,
- E-Mail-Adresse,
- Passwort (verschlüsselt gespeichert),
- Rolle (z.B. Admin, Mitglied),
- Zugehörige Team-IDs (Referenzen),
- Zeitzone.

Teamdaten $\langle D20 \rangle$

Daten der Teams (max. 20):

- Team-ID (eindeutiger Schlüssel),
- Teamname,
- Beschreibung (optional),
- Liste von Benutzer-IDs (Mitglieder),
- Erstellungsdatum.

Projektdaten $\langle D30 \rangle$

Daten zu Projekten (max. 50):

- Projekt-ID (eindeutiger Schlüssel),
- Projektname,
- Beschreibung,
- Kategorie (z.B. Hausarbeit, Teamarbeit),
- Zugehöriges Team oder Benutzer-ID,

- Zielzeit (geplante Zeit in Minuten),
- Startdatum,
- Enddatum,
- Liste von Task-IDs (Referenzen auf $\langle D35 \rangle$).

Taskdaten $\langle D35 \rangle$

Daten zu Aufgaben innerhalb von Projekten (max. 1.000):

- Task-ID (eindeutiger Schlüssel),
- Titel der Aufgabe,
- Beschreibung (optional),
- Zugehörige Projekt-ID (Referenz),
- Verantwortliche Benutzer-ID (optional),
- Status (offen, in Bearbeitung, erledigt),
- Fälligkeitsdatum (optional),
- Erstellungsdatum,
- Letztes Änderungsdatum.

Zeiteinträge $\langle D40 \rangle$

Zeiterfassungsdaten (max. 10.000):

- Zeiteintrags-ID (eindeutiger Schlüssel),
- Task-ID (Referenz auf $\langle D35 \rangle$),
- Startzeitpunkt,
- Endzeitpunkt,
- Dauer (in Minuten),
- Beschreibung der Tätigkeit (optional),
- Art der Erfassung (manuell/automatisch),
- Letztes Änderungsdatum.

Benachrichtigungen $\langle D50 \rangle$

Systembenachrichtigungen (max. 200):

- Benachrichtigungs-ID (eindeutiger Schlüssel),
- Empfänger (Benutzer-ID),
- Typ (z.B. Warnung bei Zeitlimit, Einladung, Erinnerung),
- Nachrichtentext,
- Status (gelesen/ungelesen),
- Erstellungszeitpunkt.

6 Nichtfunktionale Anforderungen

In diesem Kapitel werden die nichtfunktionalen Anforderungen, sowie die zugehörigen Qualitätsmerkmale des Projekts systematisch beschrieben. Zur Bewertung der einzelnen Qualitätsmerkmale werden zwei Kriterien herangezogen: ihre Relevanz für das Projekt, sowie der angestrebte Grad ihrer Umsetzung. Diese Einschätzung erfolgt übersichtlich in Tabellenform. Im Anschluss an jede Tabelle werden die getroffenen Bewertungen und Entscheidungen näher erläutert.

6.1 Funktionalität

Produktqualität	sehr gut	gut	normal	nicht relevant
Angemessenheit		x		
Richtigkeit		x		
Interoperabilität			x	
Ordnungsmäßigkeit			x	

Das Produkt ist dahingehend gut angemessen, da die Funktionalitäten praxis- und benutzerorientiert sind und unnötige Komplexität dadurch vermieden wird. Die Richtigkeit der Daten und Zeiten ist besonders wichtig für die Analyse und Kontrolle des Zeitaufwands. Bezüglich der Interoperabilität und Ordnungsmäßigkeit werden die wichtigsten Anforderungen grundsätzlich erfüllt. Eine tiefgreifende Integration in externe Kalenderdienste wie Google Kalender sowie eine vollständige Zertifizierung sind jedoch nicht vorgesehen.

6.2 Sicherheit

Produktqualität	sehr gut	gut	normal	nicht relevant
Zuverlässigkeit		x		
Reife		x		
Fehlertoleranz		x		
Wiederherstellbarkeit	x			

Die Zeiterfassung und Analyse funktionieren im Alltag zuverlässig und weisen nur selten Störungen auf, die in der Regel auch manuell durch den Nutzer:in behoben werden können. Aufgrund

fehlender Langzeiterfahrungen kann das Produkt derzeit noch keine herausragende Reife vorweisen, jedoch wird durch umfassende Tests mindestens ein guter Reifegrad sichergestellt. Eine hohe Fehlertoleranz wird erwartet, da ein fehlerfreier Ablauf der Funktionen die Grundlage des Produkts bildet. Da die zu erfassenden Projekte eine längere Laufzeit haben, sind Datenverluste nicht vertretbar. Deshalb werden die Daten zur schnellen Wiederherstellung auf einem Server gespeichert.

6.3 Benutzbarkeit

Produktqualität	sehr gut	gut	normal	nicht relevant
Verständlichkeit	x			
Erlernbarkeit		x		
Bedienbarkeit	x			
Effizienz	x			
Zeitverhalten		x		
Verbrauchsverhalten		x		

Durch eine klar strukturierte, logisch aufgebaute und gut beschriftete Benutzeroberfläche ist eine verständliche Nutzung der Anwendung gewährleistet. Dies führt zu einer leicht erlernbaren Bedienung, unterstützt durch leicht zugängliche Funktionen und klar definierte Anforderungen. Die Anwendung dient der effizienten Bearbeitung von Projekten und unterstützt die Nutzer:innen mit minimalem Aufwand in ihren Arbeitsprozessen. Die Antwortzeiten sind ausreichend schnell für eine flüssige Nutzung, ohne die Zeiterfassung zu beeinträchtigen. Da es sich um eine webbasierte Anwendung handelt, wird bei den Nutzer:in nahezu kein Speicher beansprucht. Allerdings sind die Serverressourcen begrenzt, weshalb das Verbrauchsverhalten insgesamt als gut einzustufen ist.

6.4 Änderbarkeit

Produktqualität	sehr gut	gut	normal	nicht relevant
Analysierbarkeit		x		
Modifizierbarkeit				x
Stabilität	x			
Prüfbarkeit		x		
Übertragbarkeit	x			
Anpassbarkeit				x
Installierbarkeit				x
Konformität		x		
Austauschbarkeit		x		

Aufgrund einer ausführlichen und korrekten Dokumentation ist eine gute Analysierbarkeit gegeben. Eine Modifizierung oder Anpassung des Produkts ist derzeit nicht vorgesehen. Ungewollte Seiteneffekte durch Nutzeränderungen sind eingeschränkt, weshalb man auf eine sehr gute Stabilität schließen kann. Die verschiedenen Funktionalitäten lassen sich gut durch unterschiedliche Tests überprüfen. Da es sich um eine webbasierte Anwendung handelt, ist eine sehr gute Übertragbarkeit und Konformität gewährleistet, da sie auf den meisten Browsern einwandfrei funktioniert. Eine klassische Installation ist daher nicht erforderlich. Der Austausch einzelner Komponenten des Produkts ist innerhalb gewisser Grenzen möglich, kann jedoch eine Anpassung des Quellcodes erforderlich machen.

6.5 Qualitätsanforderungen

Die zuvor als besonders relevant eingestuftten Qualitätsmerkmale der Software werden an dieser Stelle präzisiert. Dabei werden die Qualitätsmerkmale in konkrete Anforderungen an das Produkt überführt.

- $\langle Q10 \rangle$ Die Funktionalitäten und Darstellungen der Webanwendung sind kompatibel in der aktuellsten Version von Google Chrome.
- $\langle Q20 \rangle$ Nach einem Fehlerereignis soll die Wiederherstellung innerhalb von 5 Minuten erfolgen.
- $\langle Q30 \rangle$ Das Produkt ist einfach zu verstehen und leicht zu bedienen.

- $\langle Q40 \rangle$ Das System soll Benutzeranfragen innerhalb von 5 Sekunden beantworten.
- $\langle Q50 \rangle$ Das Produkt soll fehlertolerant bezüglich Bedien- und Eingabefehler sein. Im Nach-
herein soll es durch den Nutzer:in anpassbar sein.

7 Benutzeroberfläche/Schnittstellen

In diesem Kapitel wird die Benutzeroberfläche sowie die vorgesehenen Schnittstellen der Anwendung ClockWise beschrieben. Ziel ist es, die spezifischen Anforderungen an das User Interface zu definieren und die wichtigsten Designentscheidungen, angepasst an die Projektziele, darzustellen.

Benutzeroberfläche:

Die Benutzeroberfläche von ClockWise ist darauf ausgerichtet, Nutzer:in eine einfache, strukturierte und effiziente Arbeitsumgebung zu bieten. Im Vordergrund stehen eine intuitive Navigation, übersichtliche Fensterlayouts und eine klare Dialogführung. Die Gestaltung ist responsiv, um sowohl auf Desktop-PCs als auch auf mobilen Geräten eine optimale Darstellung zu ermöglichen.

Zentrale Bestandteile der Benutzeroberfläche sind:

- Login-Maske für den Zugang zum System
- Dashboard mit Projekt- und Zeitübersicht
- Time-Tracking-Seite zur Visualisierung eingetragener Zeiten
- Analysebereich mit grafischer Auswertung und Statistiken
- Projektseite zur Erstellung und Ergänzung von Projekten und Aufgabenverteilung
- Team-Management-Seite zur Verwaltung von Teams und Mitgliedern
- Kalenderansicht zur übersichtlichen Darstellung und Auswahl von Zeiteinträgen

Die Hauptnavigation erfolgt über ein Menü am linken Bildschirmrand. Zur Bedienung kommen neben Maus auch Touch-Eingaben in Frage.

Spezielle Anforderungen an die Gestaltung orientieren sich an etablierten Usability-Standards wie der ISO 9241-10. Die Details der Umsetzung werden durch Prototypen und Testsysteme weiter konkretisiert.

Gibt es verschiedene Rollen, die das Produkt nutzen, so werden die Oberflächen und Zugriffsrechte für jede Rolle individuell angepasst. So erhält beispielsweise die Teamleitung erweiterte

Verwaltungsoptionen, während normale Teammitglieder ausschließlich ihre eigenen Zeiteinträge und persönliche Auswertungen einsehen können.

Wichtig: Im Folgenden werden beispielhaft zwei Mockups der Benutzeroberfläche dargestellt.

Die einzelnen UI-Anforderungen sind mit dem Präfix UI analog zu den Funktionsanforderungen nummeriert.

Rolle	Rechte	Benutzeroberfläche
Teammitglied	Zeiterfassung, Statistiken	Zeiterfassungsmaske, Kalender, Analyse
Teamleitung	Verwaltung Team, Projektübersicht	Teamverwaltung, Projektübersicht, Statistik

Login $\langle UI10 \rangle$

Nach dem Aufruf erscheint eine kompakte Login-Maske mit Logo, Eingabefeldern für E-Mail und Passwort sowie einem Login-Button und der Möglichkeit zur Passwortwiederherstellung. Siehe Abbildung 7.1.

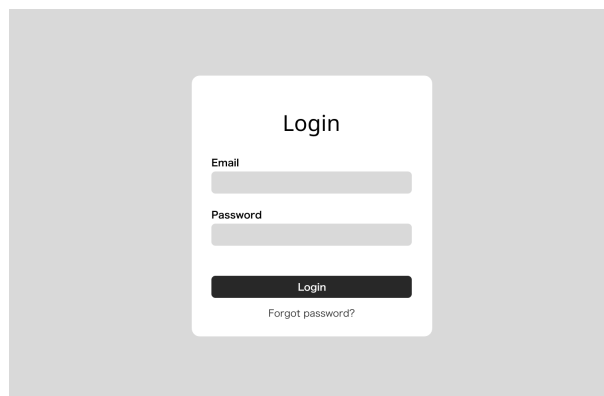


Abbildung 7.1: Login $\langle UI10 \rangle$

Dashboard $\langle UI20 \rangle$

Das Dashboard bietet eine Übersicht über laufende Projekte und zeigt die insgesamt auf jedes Projekt aufgewendete Zeit. Es bietet eine Kalenderansicht zur Auswahl von Daten und eine grafische Auswertung der Teamaktivitäten und Zeitverteilung. Siehe Abbildung 7.2.

Projects $\langle UI30 \rangle$

Die Projektseite zeigt eine Übersicht aller erstellten Projekte, aufgeteilt in Solo- und Teamprojekte. Für jedes Projekt werden Typ, Zeitlimit und bereits geleistete Stunden angezeigt. Die Projekte können gefiltert und über den „View“-Button aufgerufen werden. Neue Projekte können über die „New Project“-Schaltfläche erstellt werden.

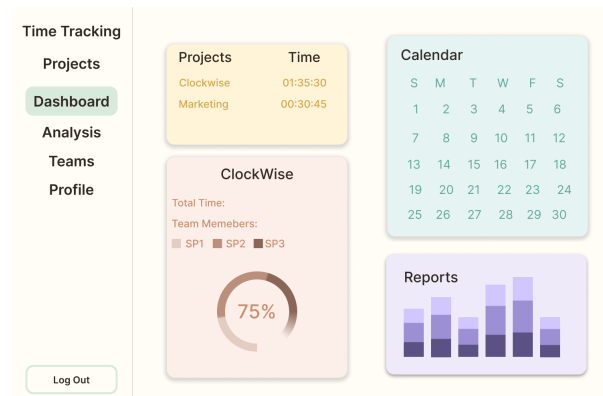


Abbildung 7.2: Dashboard $\langle UI20 \rangle$

Zusätzlich können auf der Projektseite direkt neue Tasks erstellt, bearbeitet und den Teammitgliedern zugewiesen werden. Die Verwaltung der Tasks erfolgt vollständig innerhalb der jeweiligen Projektansicht.

Siehe Abbildung 7.3.

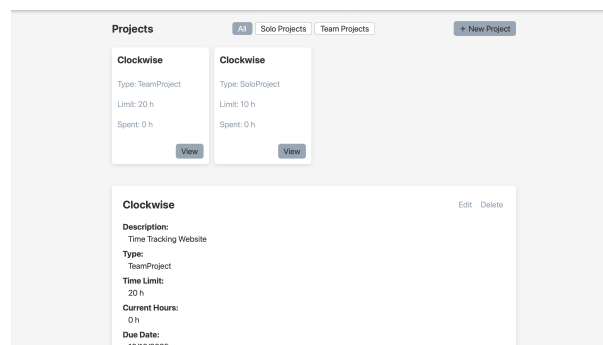


Abbildung 7.3: Projects $\langle UI30 \rangle$

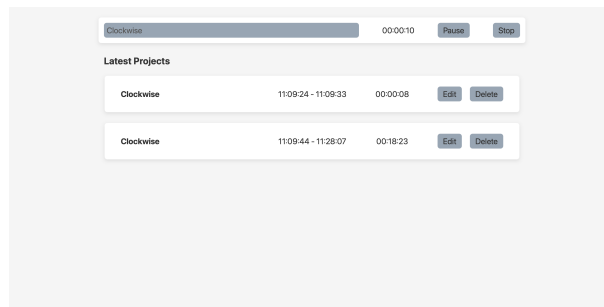
Time Tracking $\langle UI40 \rangle$

Die Time-Tracking-Seite ermöglicht es Nutzer:in, ihre Arbeitszeit projektbezogen zu erfassen. Über ein zentrales Eingabefeld kann ein Projekt ausgewählt und ein Timer gestartet werden. Zusätzlich werden alle kürzlich erfassten Zeitspannen gelistet, mit Optionen zum Bearbeiten und Löschen. Die Buttons „Pause“ und „Stop“ ermöglichen eine präzise Steuerung der Zeiterfassung. Siehe Abbildung 7.4.

Analysis $\langle UI50 \rangle$

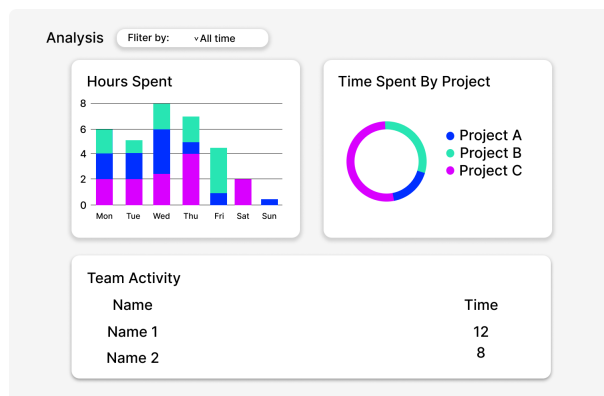
Die Analyseseite bietet einen umfassenden Überblick über die erfassten Arbeitszeiten und deren Verteilung auf verschiedene Projekte und Teammitglieder. Nutzer:in können den Analysezeitraum über ein Dropdown-Menü (z. B. All time, This week, This month) filtern.

Die Seite ist in folgende Hauptbereiche unterteilt:

Abbildung 7.4: Time Tracking $\langle UI40 \rangle$

- **Stundenverteilung (Bar-Chart):** Visualisiert die Anzahl der gearbeiteten Stunden pro Wochentag, um Stoßzeiten und produktive Tage zu identifizieren.
- **Projektverteilung (Pie-Chart):** Zeigt die prozentuale Aufteilung der Arbeitszeit auf verschiedene Projekte.
- **Teamaktivität (Tabelle):** Stellt die geleisteten Stunden pro Teammitglied in einer tabellarischen Form dar, um die Teambeiträge transparent zu machen.

Siehe Abbildung 7.5.

Abbildung 7.5: Analysis $\langle UI50 \rangle$

Team Management $\langle UI60 \rangle$

Die Team-Management-Seite ermöglicht die Verwaltung von Teams und Teammitgliedern. Nutzer:innen können neue Mitglieder hinzufügen oder bestehende Mitglieder entfernen. Außerdem wird angezeigt, welchen Teammitgliedern bereits Aufgaben zugewiesen sind. Die Projektauswahl erfolgt über ein Dropdown-Menü, um das Team-Management kontextabhängig für ein bestimmtes Projekt anzuzeigen.

Die Seite ist in folgende Hauptbereiche unterteilt:

- **Teamübersicht (Tabelle):** Zeigt eine Liste aller Teammitglieder mit Name, E-Mail und einem Profilbild zur schnellen visuellen Identifikation.
- **Aktionen:** „Add Member“- und „Delete Member“-Buttons zur einfachen Verwaltung der Teammitglieder.
- **Task-Übersicht:** Zeigt eine Liste der im aktuellen Projekt vorhandenen Tasks sowie die zugewiesenen Teammitglieder mit ihren Profilbildern.

Hinweis: Die Zuweisung von Tasks erfolgt ausschließlich über die Projektseite.

Siehe Abbildung 7.6.

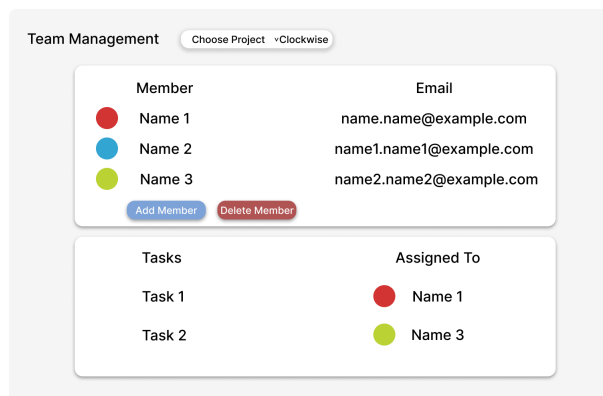


Abbildung 7.6: Team Management $\langle UI60 \rangle$

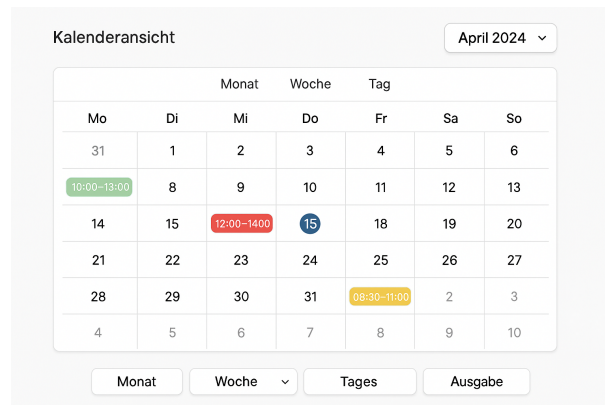
Kalenderansicht $\langle UI70 \rangle$

Die Kalenderansicht ermöglicht eine übersichtliche Darstellung aller erfassten Zeiten im Kalenderformat, wie in den Muss-Kriterien ($\langle RM7 \rangle$) gefordert. Nutzer:innen können über die Kalenderansicht gezielt einzelne Tage auswählen, um eine detaillierte Übersicht der an diesem Tag erfassten Zeiteinträge zu erhalten.

Die Seite unterstützt eine Monats-, Wochen- und Tagesansicht, um je nach Bedarf eine kompakte oder detaillierte Darstellung zu ermöglichen. Farbliche Markierungen oder Symbole visualisieren unterschiedliche Projekte und Tasks.

Die Hauptfunktionen der Kalenderansicht sind:

- **Direkte Datumauswahl:** Schneller Zugriff auf die Zeiteinträge eines bestimmten Tages.
- **Visuelle Darstellung:** Übersichtliche Darstellung der erfassten Zeiten in verschiedenen Ansichten (Monat, Woche, Tag).
- **Navigation:** Möglichkeit, zwischen vorherigen und kommenden Zeiträumen zu wechseln.

Abbildung 7.7: Kalenderansicht $\langle UI70 \rangle$

Siehe Abbildung 7.7.

Schnittstellen:

Im Rahmen von ClockWise werden Schnittstellen in Form von REST-APIs angeboten, über die externe Systeme auf Zeitdaten oder Projektinformationen zugreifen können. Diese Schnittstellen erlauben beispielsweise den Export von Daten sowie die Integration mit anderen Management-Tools. Detaillierte Spezifikationen erfolgen im entsprechenden Anhang.

8 Technische Produktumgebung

In diesem Kapitel erfolgt die Beschreibung der technischen Umgebung, in der ClockWise entwickelt, betrieben und genutzt werden kann. Da es sich um eine Client-Server-basierte Webanwendung handelt, wird zwischen Anforderungen an Client und Server unterschieden.

8.1 Software

Server: Die serverseitige Logik von ClockWise basiert auf Python in Kombination mit dem Webframework Flask. Als Datenbanksystem dient SQLite. Der Datenzugriff erfolgt über den ORM SQLAlchemy. Diese Kombination ermöglicht eine strukturierte und effiziente Speicherung sowie Abfrage der erfassten Zeit- und Nutzerdaten. Während der Entwicklung erfolgt der Betrieb lokal auf Windows-Systemen. Auch die Ausführung der Anwendung erfolgt lokal auf einem Windows-System mithilfe von Flask. Ein Deployment auf einem externen Server ist im Rahmen dieses Projekts zunächst nicht vorgesehen, kann jedoch bei Bereitstellung eines geeigneten VM-Servers durch das Peter L. Reichertz Institut für Medizinische Informatik realisiert werden.

Client: Die Benutzeroberfläche basiert auf HTML5, CSS3 und JavaScript. Sie ist mit allen gängigen modernen Webbrowsern (z. B. Chrome, Firefox, Edge, Safari) kompatibel und erfordert keine zusätzliche lokale Installation von Software.

8.2 Hardware

Server: Während der Entwicklung wird ClockWise lokal auf Standardrechnern mit Windows-Betriebssystem betrieben. Bei einem Deployment auf einem externen VM-Server wird ein System mit mindestens 2 CPU-Kernen, 4 GB RAM und mindestens 10 GB freiem Festplattenspeicher benötigt.

Client: ClockWise kann auf allen internetfähigen Endgeräten mit einem aktuellen Webbrowser genutzt werden. Dazu zählen insbesondere PCs, Laptops, Tablets und Smartphones. Eine hohe Bildschirmauflösung sowie eine hohe Reaktionsgeschwindigkeit des Endgeräts – etwa beim Laden von Analysen oder bei Nutzerinteraktionen – verbessern die Nutzungserfahrung, sind jedoch nicht zwingend erforderlich.

Als Referenzgerät für die Client-Nutzung und Tests wird ein handelsüblicher Laptop mit folgenden Spezifikationen verwendet: Samsung Galaxy Book4 360 mit Windows 11 Pro (64-Bit), Core(TM) i7-150U, 16 GB RAM, 512 GB SSD, einem 15,6 Zoll Bildschirm (1,920 × 1,080 Pixel) sowie Wi-Fi 6 (802.11ax) für die Internetverbindung. Als Webbrowser wird Google Chrome in der Version 136.0.7103.93 eingesetzt.

8.3 Produktschnittstellen

ClockWise ist als eigenständige Webanwendung konzipiert, sodass keine direkte Integration in eine bereits bestehende Systemlandschaft vorgenommen wird. Es werden jedoch Schnittstellen zur Datenweitergabe durch Exporte im CSV- und PDF-Format implementiert, sodass die erzeugten Daten in anderen Anwendungen weiterverwendet werden können. Der Versand von Benachrichtigungen per E-Mail erfordert eine Anbindung an einen SMTP-Dienst. Sofern ein Deployment auf einem externen VM-Server erfolgt, kann der E-Mail-Versand dort über einen konfigurierbaren SMTP-Server realisiert und der Zugriff über HTTPS bereitgestellt werden.

9 Glossar

Webanwendung Eine über den Webbrowser nutzbare Software, die keine Installation auf dem eigenen Gerät erfordert.

Benutzeroberfläche Der Teil einer Software, den Nutzer:innen sehen und bedienen können.

Webbrowser Ein Programm zur Anzeige und Nutzung von Webseiten (z.B. Chrome, Firefox, Edge).

Export Funktion zur Ausgabe und Speicherung von Daten aus der Anwendung in verschiedenen Dateiformaten, um sie weiterzuverwenden oder zu archivieren.

CSV „Comma-Separated Values“ – ein Dateiformat zur Speicherung tabellarischer Daten in Textform.

PDF „Portable Document Format“ – ein weit verbreitetes Dateiformat zur originalgetreuen Darstellung und Weitergabe von Dokumenten, unabhängig vom verwendeten Betriebssystem oder Gerät.

Flask Leichtgewichtiges Web-Framework für Python zur Entwicklung von Webanwendungen (siehe <https://flask.palletsprojects.com/>).

CSS3 Cascading Style Sheets – eine Stylesheet-Sprache zur Gestaltung von HTML-Inhalten.

HTML5 Die fünfte Version der Hypertext Markup Language zur Strukturierung von Webseiteninhalten.

JavaScript Eine Programmiersprache zur Umsetzung interaktiver Funktionen im Webbrowser.

SQLite Eine leichtgewichtige, serverlose Datenbank, die direkt in Anwendungen eingebettet wird (siehe <https://www.sqlite.org/>).

SQLAlchemy Ein in Python geschriebenes Toolkit zur Datenbankprogrammierung, das die Arbeit mit relationalen Datenbanken vereinfacht (siehe <https://www.sqlalchemy.org/>).

ORM „Object-Relational Mapping“ – eine Technik, die Objekte in Programmiersprachen mit Datenbanktabellen verknüpft, um Datenbankoperationen objektorientiert durchzuführen.

Datenbanksystem Software zur Speicherung, Verwaltung und Abfrage strukturierter Daten.

SMTP-Dienst „Simple Mail Transfer Protocol“ – ein Protokoll zum Versenden von E-Mails über das Internet.

RAM „Random Access Memory“ – der Arbeitsspeicher eines Computers, in dem laufende Programme und Daten temporär gespeichert werden.

CPU-Kern(e) Ein Verarbeitungseinheit innerhalb eines Prozessors (CPU), die eigenständig Rechenoperationen ausführen kann. Mehrere Kerne ermöglichen parallele Verarbeitung.

Deployment Der Prozess der Veröffentlichung und Bereitstellung einer Softwareanwendung auf einem Server oder in der Cloud zur Nutzung durch Endnutzer:innen.

VM-Server Ein Server, der mehrere virtuelle Maschinen (VMs) auf einer physischen Hardware betreibt. Jede VM läuft wie ein eigener Computer mit eigenem Betriebssystem.

HTTPS „Hypertext Transfer Protocol Secure“ – eine verschlüsselte Version des HTTP-Protokolls, die für eine sichere Datenübertragung im Web sorgt. Erkennbar an der URL mit „https://“.

OAuth Ein offener Standard zur sicheren Autorisierung, der es Nutzern ermöglicht, sich über Drittanbieter-Dienste wie Google oder GitHub bei Anwendungen anzumelden, ohne eigene Zugangsdaten zu hinterlegen.

Systeminterne Benachrichtigung Eine Nachricht, die direkt innerhalb der Anwendung angezeigt wird – z. B. als Hinweis, Pop-up oder Symbol in der Benutzeroberfläche. Sie informiert den Nutzer über relevante Ereignisse, ohne eine externe E-Mail oder Push-Nachricht zu verwenden.