



# Recherchebericht

## Inhaltsverzeichnis

<b>1. Allgemeines.....</b>	<b>2</b>
1.1 Einsatzumfeld und Problemstellung.....	2
1.2 Grundsätzliche Logik einer prototypischen Applikation.....	2
1.3 Terminologie (Grundlage für das Glossar).....	3
1.4 Mögliche Leistungsmerkmale einer prototypischen Applikation.....	3
<b>2. Übersicht über Themenrelevante Applikationen.....</b>	<b>4</b>
2.1. TimeFlow.....	4
2.2. Motion.....	4
2.3. TimeHero.....	5
2.4. LiquidPlanner.....	6
<b>3. Übersicht über IT-Spezifika solcher Applikationen.....</b>	<b>7</b>
3.1. Technologien in bestehenden Anwendungen.....	7
3.2. Technologien und Kompetenzen der Projektgruppe.....	8
<b>4. Genauere Beschreibung einer konkreten Applikation.....</b>	<b>10</b>
4.1. Verbale Beschreibung der Leistungsmerkmale.....	10
4.2. Beschreibung von Anwendungsfällen.....	11
4.3. Sichtbare Aspekte der inneren Logik.....	12
<b>5. Nachforschungen für den Scheduling- Algorithmus.....</b>	<b>14</b>
<b>Quellen.....</b>	<b>16</b>



# 1. Allgemeines

## 1.1. Einsatzumfeld und Problemstellung

In modernen Arbeitsumgebungen spielt die effiziente Organisation von Aufgaben und Zeit eine zentrale Rolle. Die Tätigkeiten entscheiden sich oft in der kognitiven Leistungsfähigkeit und stellen Nutzende häufig vor die Herausforderung, eine Vielzahl von Aufgaben, Terminen und Deadlines gleichzeitig zu bewältigen.

Bestehende Planungssysteme stoßen hierbei oft an ihre Grenzen, da sie zwar Aufgaben verwalten, jedoch individuelle Faktoren wie Arbeitsbelastung oder zeitabhängige Leistungsfähigkeit nur unzureichend berücksichtigen. Dies kann zu Überlastung, ineffizienter Arbeitsweise und einer suboptimalen Nutzung der verfügbaren Zeit führen.

Vor diesem Hintergrund besteht ein Bedarf an Anwendungen, die Aufgaben nicht nur verwalten, sondern intelligent planen und dabei sowohl zeitliche als auch individuelle Faktoren berücksichtigen.

## 1.2. Grundsätzliche Logik einer prototypischen Applikation

Eine prototypische Anwendung im Bereich der Aufgaben- und Zeitplanung verfolgt das Ziel, Arbeitsprozesse strukturiert und effizient zu organisieren. Dazu werden Aufgaben zunächst vom Nutzer erfasst und mit relevanten Informationen wie Dauer, Priorität und Deadline versehen.

Das System analysiert anschließend die verfügbaren Zeiträume sowie bestehende Termine und erstellt daraus einen strukturierten Plan. Moderne Anwendungen nutzen hierfür algorithmische Verfahren, um Aufgaben automatisch geeigneten Zeitfenstern zuzuordnen („Auto-Scheduling“).

Ein zentrales Merkmal ist dabei die dynamische Anpassungsfähigkeit: Wird der Plan durch neue Aufgaben oder Termine verändert, passt das System die Planung automatisch an.

Darüber hinaus kann eine solche Anwendung Faktoren wie Arbeitsbelastung oder zeitabhängige Leistungsfähigkeit einbeziehen, um eine möglichst effiziente und realistische Planung zu ermöglichen.

## 1.3. Terminologie (Grundlage für das Glossar)

Im Kontext solcher Anwendungen werden verschiedene Fachbegriffe verwendet, die für das Verständnis der Funktionsweise relevant sind:

- Task: Eine einzelne Aufgabe, die geplant und bearbeitet wird



- Deadline: Zeitpunkt, bis zu dem eine Aufgabe abgeschlossen sein muss
- Time Blocking: Einteilung von Arbeitszeit in feste Zeitfenster für bestimmte Aufgaben
- Auto-Scheduling: Automatische Einplanung von Aufgaben in den Tagesplan durch das System
- Task Dependencies: Abhängigkeiten zwischen Aufgaben
- Capacity Planning: Abstimmung von Arbeitsaufwand und verfügbarer Zeit
- Focus Time: Zeitfenster für konzentriertes Arbeiten ohne Unterbrechungen
- Chronotyp: Individuelle Leistungspräferenz im Tagesverlauf

Diese Begriffe bilden die Grundlage für eine einheitliche Terminologie im weiteren Verlauf des Projekts und werden im Glossar detailliert aufgeführt.

## 1.4. Mögliche Leistungsmerkmale einer prototypischen Applikation

Aus der Analyse bestehender Anwendungen sowie der identifizierten Anforderungen lassen sich zentrale Leistungsmerkmale einer prototypischen Applikation ableiten.

Eine solche Anwendung ermöglicht zunächst die strukturierte Erfassung und Verwaltung von Aufgaben. Diese können mit zusätzlichen Informationen wie Priorität, Dauer und Deadline versehen werden. Darauf aufbauend erfolgt eine automatische und flexible Planung, bei der Aufgaben geeigneten Zeitfenstern zugeordnet werden.

Ein wesentliches Merkmal ist die dynamische Anpassung des Plans bei Veränderungen, beispielsweise durch neue Termine oder Verzögerungen.

Darüber hinaus bieten moderne Anwendungen Funktionen zur Visualisierung der Arbeitsbelastung, wodurch Nutzende ihre Auslastung besser einschätzen können.

Ziel einer solchen Applikation ist es, Arbeitsprozesse effizient zu organisieren, Überlastung zu vermeiden und die Produktivität durch intelligente Planung zu steigern.



## 2. Übersicht über Themenrelevante Applikationen

### 2.1. TimeFlow

#### Allgemeine Beschreibung

TimeFlow ist eine Anwendung zur visuellen Strukturierung von Arbeitsabläufen und zur Unterstützung der individuellen Tagesplanung. Der Fokus liegt auf einer übersichtlichen Darstellung von Aufgaben, um die persönliche Arbeitsbelastung besser einschätzen zu können.

#### Einsatzumfeld

Die Anwendung richtet sich vor allem an Einzelpersonen, Freelancer oder kleine Teams, die eine transparente und einfache Methode zur Tagesplanung suchen.

#### Grundsätzlicher Ablauf der Software

TimeFlow fokussiert sich auf die visuelle Darstellung von Zeitblöcken. Aufgaben werden direkt in einen Zeitstrahl integriert („Time Blocking“), wodurch Überlastung durch parallele Aufgaben vermieden werden soll.

Durch die grafische Darstellung des sogenannten „Visual Workload“ wird die aktuelle Auslastung der Nutzerinnen und Nutzer sofort sichtbar. Freie Kapazitäten können dadurch leichter erkannt und genutzt werden.

Die Anwendung bietet zudem ein benutzerfreundliches Drag-and-Drop-Interface, mit dem Aufgaben flexibel verschoben werden können. Zusätzlich besteht eine Integration in Kalenderanwendungen, wodurch bestehende Termine berücksichtigt werden. Ein integrierter Fokus-Modus unterstützt konzentriertes Arbeiten, indem Ablenkungen reduziert werden. [1]

### 2.2. Motion

#### Allgemeine Beschreibung

Motion dient als primäres Beispiel für eine softwaregestützte, algorithmische Lösung zur Vermeidung individueller Überlastung. Die Anwendung kombiniert Aufgabenmanagement mit intelligenter Zeitplanung, um Arbeitsprozesse effizient zu organisieren.

#### Einsatzumfeld

Die Anwendung richtet sich insbesondere an Wissensarbeitende sowie Teams in dynamischen Arbeitsumgebungen, wie beispielsweise Agenturen oder Start-ups. Diese zeichnen sich häufig durch eine hohe Anzahl an Meetings und kurzfristige Änderungen aus, wodurch eine flexible und adaptive Planung erforderlich ist.

#### Grundsätzlicher Ablauf der Software

<https://projekt.goaldone.de>

[team@goaldone.de](mailto:team@goaldone.de)



Motion verknüpft eine Aufgabenliste mit einem digitalen Kalender. Nutzende geben Aufgaben inklusive geschätzter Dauer und Deadline ein. Anschließend übernimmt ein Algorithmus die automatische Planung („Auto-Scheduling“), indem er die Aufgaben in freie Zeitfenster im Kalender einordnet.

Bei Änderungen, wie neuen Terminen oder Verzögerungen, passt das System den gesamten Zeitplan in Echtzeit an. Dadurch wird sichergestellt, dass Aufgaben weiterhin realistisch eingeplant bleiben. Zusätzlich berücksichtigt die Anwendung sogenannte „Focus Time“, also geschützte Zeitblöcke, die gezielt für konzentriertes Arbeiten ohne Unterbrechungen reserviert werden.

Die Software bietet darüber hinaus Funktionen zur intelligenten Priorisierung von Aufgaben sowie eine automatisierte Synchronisation mit bestehenden Kalendern. Warnmeldungen informieren die Nutzenden frühzeitig über mögliche Überschreitungen von Deadlines, wodurch rechtzeitig Gegenmaßnahmen ergriffen werden können.[2]

## 2.3. TimeHero

### Allgemeine Beschreibung

TimeHero ist eine Anwendung zur Planung und Organisation von Projekten, die sich insbesondere auf die Abbildung von Abhängigkeiten zwischen Aufgaben sowie die Kapazitätsplanung innerhalb von Teams konzentriert. Ziel der Software ist es, Arbeitsprozesse effizient zu strukturieren und die Zusammenarbeit mehrerer Beteiligter zu erleichtern.

### Einsatzumfeld

Die Anwendung richtet sich vor allem an projektorientierte Organisationen und Teams, beispielsweise Unternehmen oder Vereine, die komplexe Abläufe mit mehreren Beteiligten koordinieren müssen.

### Grundsätzlicher Ablauf der Software

In TimeHero werden Aufgaben innerhalb von Projekten erfasst und miteinander verknüpft. Dabei werden sogenannte Aufgabenabhängigkeiten („Task Dependencies“) berücksichtigt, wodurch sichergestellt wird, dass bestimmte Aufgaben erst begonnen werden können, wenn vorherige abgeschlossen sind.

Das System überprüft kontinuierlich die Verfügbarkeit der Teammitglieder und gleicht diese mit dem Arbeitsaufwand ab („Capacity Planning“). Auf dieser Basis schlägt die Anwendung automatisch den nächsten logischen Arbeitsschritt vor und unterstützt so eine strukturierte Projektabwicklung.

Darüber hinaus bietet TimeHero Funktionen zur Erstellung von Vorlagen für wiederkehrende Prozesse, wodurch Arbeitsabläufe standardisiert werden können. Die Auslastung des Teams wird visuell dargestellt, sodass Engpässe frühzeitig erkannt werden. Bei Verzögerungen passt das System automatisch die Zeitplanung an und aktualisiert die Startdaten nachfolgender Aufgaben. [3]



Basierend auf der Analyse von Motion und TimeHero haben wir uns entschieden, das Feature Auto-Scheduling als Kernfunktionalität für unser Projekt zu priorisieren, da dies den größten Mehrwert für die Zielgruppe bietet.

## 2.4. LiquidPlanner

### Allgemeine Beschreibung

LiquidPlanner ist ein Projektmanagement-Tool, das sich auf vorausschauende Planung und den Umgang mit unsicheren Zeitangaben konzentriert. Die Anwendung ermöglicht es, Projekte realistischer zu planen, indem sie Unsicherheiten in der Zeitplanung systematisch berücksichtigt.

### Einsatzumfeld

Die Software wird vor allem in größeren Organisationen sowie in IT-Teams eingesetzt, die mit komplexen Projektstrukturen, vielen Abhängigkeiten und sich häufig ändernden Prioritäten arbeiten.

### Grundsätzlicher Ablauf der Software

Im Gegensatz zu klassischen Planungssystemen arbeitet LiquidPlanner nicht mit festen Zeitangaben, sondern mit Schätzbereichen. Nutzende geben für Aufgaben einen Best-Case- und einen Worst-Case-Wert an („Ranged Estimation“).

Auf Basis dieser Angaben berechnet ein Algorithmus automatisch eine realistische Projektzeitplanung („Predictive Scheduling“). Dabei werden Wahrscheinlichkeiten berücksichtigt, um mögliche Verzögerungen einzubeziehen. Im Gegensatz dazu fokussiert sich unser Ansatz auf eine intuitive Bedienung für Einzelpersonen, um die Einstiegshürde gering zu halten.

Verändern sich Prioritäten oder treten Verzögerungen auf, passt das System die gesamte Planung dynamisch an und berechnet neue Zeitverläufe. Dadurch bleibt der Projektplan stets aktuell und realistisch.

Zusätzlich bietet die Anwendung Funktionen zur prioritätsgesteuerten Ressourcenplanung, bei der festgelegt wird, welche Aufgaben zuerst bearbeitet werden. Eine integrierte Risikoanalyse hilft dabei, mögliche Terminüberschreitungen frühzeitig zu erkennen. Dynamische Visualisierungen, wie Wasserfalldiagramme, unterstützen die Darstellung von Projektverläufen und Zeitentwicklungen. [4]



## 3. Übersicht über IT-Spezifika solcher Applikationen

### 3.1. Technologien in bestehenden Anwendungen

#### Programmiersprachen und Frameworks

Im Bereich moderner Planungs- und Projektmanagementanwendungen haben sich klare technologische Standards etabliert. Viele Anwendungen, wie beispielsweise Motion, setzen im Frontend auf Webtechnologien wie JavaScript oder TypeScript [5], da diese die Entwicklung interaktiver und dynamischer Benutzeroberflächen ermöglichen.

Für die Umsetzung solcher Benutzeroberflächen werden häufig moderne Frameworks wie React eingesetzt [5]. Für mobile Anwendungen kommt ergänzend häufig React Native zum Einsatz [6], wodurch plattformübergreifende Lösungen realisiert werden können.

Im Backend werden vor allem performante und skalierbare Technologien verwendet. Dazu zählen Node.js-basierte Frameworks wie NestJS oder Express.js sowie Plattformen wie C#/.NET oder Spring Boot [7]. Diese Technologien ermöglichen eine effiziente Verarbeitung von Anfragen und bilden die Grundlage für die Entwicklung komplexer und leistungsfähiger Webanwendungen.

#### Datenbanken

Zur Abbildung komplexer Relationen und zur Verwaltung großer Datenmengen setzen moderne Planungs- und Projektmanagementanwendungen häufig auf relationale Datenbanksysteme. Ein verbreiteter Standard ist hierbei die Nutzung von PostgreSQL [8], da diese Datenbank eine hohe Zuverlässigkeit, Skalierbarkeit und Flexibilität bietet.

Auch Anwendungen wie Motion greifen auf relationale Datenbanken zurück. Ursprünglich wurde dort seit Anfang 2022 die Datenbank CockroachDB eingesetzt, bevor Mitte 2025 eine vollständige Migration zu PostgreSQL erfolgte [9]. Durch diesen Wechsel konnte die durchschnittliche Latenzzeit um etwa 33 % reduziert werden, während gleichzeitig jährliche Infrastrukturkosten von über 110.000 US-Dollar eingespart werden konnten [8].

Dieses Beispiel verdeutlicht die praktische Relevanz und Leistungsfähigkeit relationaler Datenbanken im Bereich moderner Webanwendungen.

#### Entwicklungswerkzeuge und Infrastruktur

Moderne Softwareprojekte setzen zunehmend auf automatisierte und reproduzierbare Infrastrukturkonzepte. So verzichtet Motion bei der Verwaltung seiner Cloud-Infrastruktur bewusst auf manuelle Konfigurationen über grafische Benutzeroberflächen („ClickOps“) und nutzt stattdessen konsequent den Ansatz der Infrastructure-as-Code (IaC). Dabei werden Cloud-Ressourcen mithilfe von TypeScript und dem Tool Pulumi definiert und verwaltet [10].



Dieser Ansatz ermöglicht eine hohe Automatisierung sowie eine bessere Nachvollziehbarkeit von Änderungen. Im Falle eines Ausfalls kann die gesamte Infrastruktur automatisiert in einer neuen Cloud-Region innerhalb kurzer Zeit neu aufgebaut werden. Laut Angaben des Unternehmens ist dies in weniger als einer Stunde möglich [10].

Auch im Bereich Projektmanagement und Entwicklungsprozesse setzen moderne Unternehmen auf alternative Ansätze. Motion nutzt beispielsweise keine klassischen täglichen Stand-up-Meetings, sondern sogenannte „Project Workflow Templates“, mit denen Aufgaben, Blocker und Freigabeprozesse automatisiert und asynchron organisiert werden [11].

Darüber hinaus erfolgt die Produktentwicklung in sogenannten „Shaping“-Phasen. In diesen werden Softwareentwickler bereits frühzeitig in die Konzeption einbezogen, um technische Herausforderungen und potenzielle Probleme („technical tar pits“) frühzeitig zu identifizieren und zu vermeiden [11].

## 3.2. Technologien und Kompetenzen der Projektgruppe

### Kompetenzen im Team

Innerhalb der Projektgruppe ist bereits relevantes Vorwissen in den für die Entwicklung eingesetzten Technologien vorhanden. Etwa die Hälfte des Teams verfügt über praktische Erfahrung mit Angular, PostgreSQL sowie Spring Boot beziehungsweise der Programmiersprache Java. Dieses vorhandene Wissen reduziert den Schulungsaufwand und minimiert das Risiko von Zeitverzögerungen durch steile Lernkurven.

Gleichzeitig besteht ein Weiterbildungsbedarf, insbesondere im Bereich der Verknüpfung von Angular und Spring Boot über REST-APIs sowie bei der Integration des verwendeten UI-Frameworks. Diese Themen sollen innerhalb des Teams gemeinsam erarbeitet werden, wobei erfahrene Mitglieder ihr Wissen an andere weitergeben.

Auf Basis dieser vorhandenen Kenntnisse wurde die folgende Technologieauswahl für das Projekt getroffen.

### Programmiersprachen und Frameworks

Für die Entwicklung der eigenen Applikation wird eine moderne und praxisorientierte Technologieauswahl verwendet. Im Frontend kommt TypeScript in Kombination mit dem Angular-Framework zum Einsatz. Die statische Typisierung von TypeScript unterstützt eine bessere Strukturierung und Wartbarkeit des Codes und gilt als Standard im Angular-Umfeld.

Zur Gestaltung der Benutzeroberfläche wird zusätzlich das UI-Framework PrimeNG eingesetzt, das die Entwicklung interaktiver und benutzerfreundlicher Oberflächen erleichtert und funktionale Parallelen zu verbreiteten Frameworks wie React aufweist.



Im Backend wird die Programmiersprache Java in Kombination mit dem Spring Boot Framework genutzt. Dieses dient insbesondere der Bereitstellung einer REST-API, welche die Kommunikation zwischen Frontend und Backend ermöglicht und eine skalierbare Anwendungsarchitektur unterstützt.

Ein zentraler Bestandteil der Anwendung ist die im Backend implementierte Planungslogik. Hierbei wird ein eigener Algorithmus zur automatischen Aufgabenverteilung entwickelt, der die Organisation und Priorisierung von Aufgaben innerhalb des Systems übernimmt.

### **Datenbanken**

Für die eigene Applikation wird eine relationale Datenbank eingesetzt. Die Wahl fiel auf PostgreSQL, da innerhalb der Projektgruppe bereits Erfahrungen mit diesem Datenbanksystem vorhanden sind.

Durch die Nutzung einer bekannten Technologie kann auf bestehendes Wissen zurückgegriffen werden, wodurch der Einarbeitungsaufwand reduziert und die Entwicklung effizient gestaltet wird. Gleichzeitig bietet PostgreSQL eine stabile und leistungsfähige Grundlage für die Speicherung und Verwaltung der anfallenden Daten.

### **Entwicklungswerkzeuge und Infrastruktur**

Für die Versionskontrolle wird im Projekt Git in Kombination mit GitHub eingesetzt. Das Repository wird gemäß den Kursanforderungen verwaltet und für die Betreuer zugänglich gemacht. Diese Vorgehensweise entspricht dem aktuellen Standard in der Softwareentwicklung und ermöglicht eine strukturierte und nachvollziehbare Zusammenarbeit innerhalb des Teams.

Für das Hosting der Anwendung wird ein Virtual Private Server (VPS) genutzt, beispielsweise von Anbietern wie Hetzner. Diese bieten bereits für geringe Kosten ausreichend Ressourcen, um sowohl die Anwendung als auch die zugehörige Datenbank zu betreiben [12]. Ergänzend können Content Delivery Netzwerke (CDNs), etwa von Cloudflare, eingesetzt werden, um die Latenzzeiten zu reduzieren und eine schnellere Auslieferung der Inhalte zu gewährleisten [13]. Dadurch entsteht eine kostengünstige, leistungsfähige und zugleich einfach zu verwaltende Infrastruktur.

Zur Organisation der Projektarbeit wird das Projektmanagement- und Issue-Tracking-Tool ClickUp verwendet. Dieses unterstützt die Aufgabenverteilung, Fortschrittskontrolle und Kommunikation innerhalb des Teams.

Für die Modellierung und Dokumentation der Softwarearchitektur kommt das Werkzeug PlantUML zum Einsatz. Dieses ermöglicht die Erstellung von UML-Diagrammen in Textform und erlaubt eine einfache Integration in das Versionskontrollsystem, wodurch Änderungen effizient nachvollzogen werden können.



## 4. Genauere Beschreibung einer konkreten Applikation

Im Folgenden werden Leistungsmerkmale sowie Arbeitsabläufe des Aufgabenmanagement-Tools „Motion“ analysiert und erläutert.

### 4.1. Verbale Beschreibung der Leistungsmerkmale

Motion übernimmt die Aufgabenplanung nahezu vollständig, indem sie Aufgaben nicht nur sammelt, sondern sie automatisch in den Kalender integriert. Basierend auf Verfügbarkeit, Arbeitszeiten, Prioritäten und Deadlines werden alle Aufgaben intelligent verteilt. Sobald sich Termine verschieben oder neue hinzukommen, passt Motion den gesamten Tagesplan dynamisch an. [14]

Die Anwendung synchronisiert sich dabei mit gängigen Kalendern wie Google Calendar, Outlook oder iCloud und bündelt alle Termine an einem Ort. Änderungen, die innerhalb von Motion vorgenommen werden, werden direkt in die jeweiligen Originalkalender zurückgespielt, wodurch sich Doppelbuchungen effektiv vermeiden lassen. [14] [15]

Für eine optimale Planung benötigt jede Aufgabe einen Namen, eine geschätzte Dauer sowie eine Deadline. Größere Aufgaben können zudem in kleinere Arbeitseinheiten unterteilt werden, sodass beispielsweise aus einer zweistündigen Aufgabe mehrere 30-minütige Blöcke entstehen. [14]

Auch die Priorisierung spielt eine wichtige Rolle: Aufgaben mit hoher Dringlichkeit oder naher Deadline werden automatisch bevorzugt eingeplant. Zusätzlich gibt es das Label „ASAP“, das einzelnen Aufgaben eine besonders hohe Priorität verleiht und sie gegenüber anderen bevorzugt behandelt. [16]

Ändert sich der Tagesablauf kurzfristig, etwa durch ein länger dauerndes Meeting, reagiert Motion selbstständig. Die Webanwendung verschiebt betroffene Aufgaben automatisch und aktualisiert den gesamten Zeitplan sofort. [14], [16] Dieses adaptive Verhalten bildet das zentrale Kernfeature der Anwendung.

Darüber hinaus bietet Motion die Möglichkeit, Erinnerungen entweder per E-Mail oder als Push-Benachrichtigung auf das Smartphone zu erhalten. [17]

### 4.2. Beschreibung von Anwendungsfällen

#### Aufgabe erstellen

Der erste und womöglich wichtigste Use Case ist die Erstellung eines neuen Tasks durch den Benutzer. Dazu klickt er auf einen Knopf und es erscheint ein Eingabefeld. Dort gibt er die relevanten Informationen wie Titel, geschätzte Dauer, Priorität sowie eine Deadline ein. Optional kann der



Benutzer die Funktion „Auto-Schedule“ aktivieren.

Nach dem Absenden der Eingaben erstellt das System einen entsprechenden Task-Eintrag. Ist die Auto-Schedule-Funktion aktiviert, wird zusätzlich die Scheduling-Engine angesprochen, um automatisch einen passenden freien Kalender Slot zu finden und die Aufgabe direkt einzuplanen. Der Task erscheint anschließend zum vorgesehenen Zeitpunkt im Kalender. Alternativ kann der Benutzer die Aufgabe auch ohne Auto-Schedule speichern. In diesem Fall wird der Task als offenes Element hinterlegt und kann zu einem späteren Zeitpunkt manuell eingeplant werden. [14], [16]

### **Auto Scheduling**

Ein weiterer Use Case ist die Durchführung des automatischen Scheduling durch das System, genauer gesagt durch die Motion-Engine. Dieser Prozess wird ausgelöst, sobald eine neue oder geänderte Aufgabe beziehungsweise Deadline vorliegt oder ein Rescheduling erforderlich ist, etwa aufgrund von Änderungen im Kalender.

Im Hauptablauf analysiert die KI zunächst alle verfügbaren Zeitfenster im Kalender und berücksichtigt dabei Faktoren wie Prioritäten und Deadlines der einzelnen Tasks. Anschließend weist sie die Aufgaben den jeweils optimalen Zeitfenstern zu. Dabei ist das System in der Lage, Aufgaben in kleinere Einheiten zu unterteilen (Chunking) oder flexibel auf mehrere Tage zu verteilen, sofern dies mit den zeitlichen Anforderungen der Aufgabe vereinbar ist.

Sollte die Auslastung des Kalenders sehr hoch sein und keine passende Planung möglich erscheinen, reagiert das System mit Alternativvorschlägen. Dazu gehört beispielsweise die Empfehlung, eine Deadline anzupassen oder zusätzliche Zeitfenster zu nutzen, gegebenenfalls auch außerhalb der üblichen Arbeitszeiten. [16]

### **Team Scheduling**

Ein weiterer Use Case ist das Team- und gemeinsame Kalender-Scheduling, bei dem ein Teamleiter oder Mitarbeiter Termine und Aufgaben für mehrere Personen koordiniert.

Das System greift dabei auf die Kalender aller Beteiligten zu, prüft deren Verfügbarkeiten und plant automatisch einen passenden Termin, der für alle passt. Bei gemeinsamen Aufgaben berücksichtigt es zudem die Kapazitäten der einzelnen Personen und verteilt die Arbeit entsprechend. Alternativ können Aufgaben auch gezielt an einzelne Teammitglieder delegiert und direkt in deren Kalender eingetragen werden. [18]

## **4.3. Sichtbare Aspekte der inneren Logik**

### **Aufgabe anlegen**

In der Webanwendung bilden die Benutzeroberflächen-Elemente wie Buttons, Formulare oder Dropdown-Menüs die Schnittstelle zwischen dem Benutzer und der internen Logik. Wenn ein Benutzer beispielsweise eine neue Aufgabe anlegt, erfasst er alle relevanten Informationen wie Titel, Dauer, Priorität, Deadline und Wiederholungsoptionen über die Eingabefelder. Die Anwendung nimmt diese Eingaben entgegen und erzeugt daraus einen neuen Aufgaben-Eintrag, der dem Benutzer zugeordnet wird und optional einem Projekt zugeordnet werden kann.



Sobald die Aufgabe angelegt ist, prüft die interne Logik verfügbare Zeitfenster im Kalender und plant die Aufgabe automatisch ein, sofern die Bedingungen erfüllt sind. Der Kalender verwaltet dabei alle geplanten Einträge zentral, sodass eine konsistente Übersicht über alle Termine des Benutzers gewährleistet ist.

Wenn ein Benutzer eine Aufgabe als erledigt markiert oder den Status einer Aufgabe ändert, aktualisiert die Logik automatisch die Verfügbarkeit der Zeit im Kalender und berücksichtigt dies bei der Planung weiterer Aufgaben. Sollte es Konflikte geben oder die Auslastung des Kalenders hoch sein, generiert das System alternative Vorschläge oder Warnungen, die dem Benutzer über Benachrichtigungen angezeigt werden.

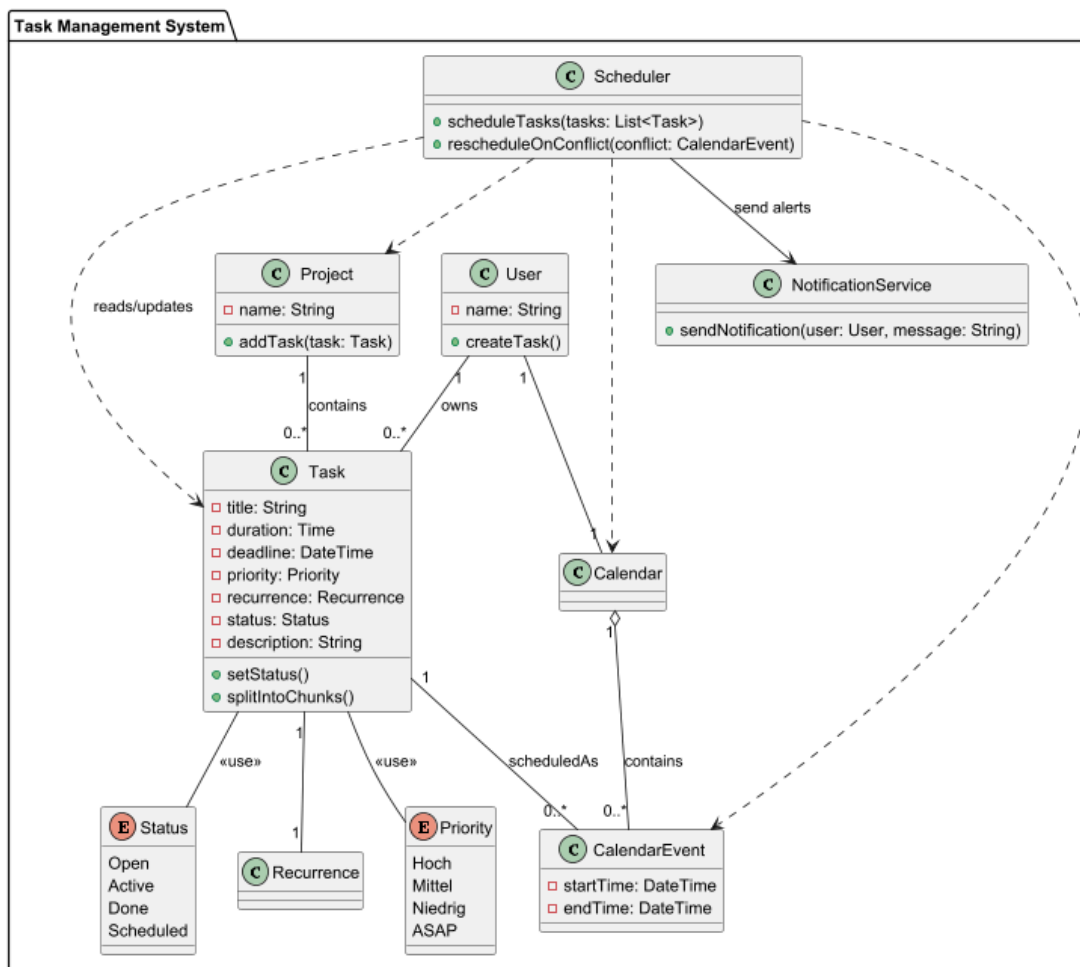


Abbildung 1: UML-Klassendiagramm, welches annähernd beschreibt, wie die Komponenten der Aufgabenmanagement-Plattform "Motion" zusammenhängen.

### Teameeting einstellen

Wenn ein Benutzer ein Teameeting anlegen möchte, interagiert er über die Benutzeroberfläche mit der Anwendung, beispielsweise durch einen Button oder ein Eingabeformular, in dem das Datum, die



Uhrzeit, die Teilnehmer und die Dauer des Meetings festgelegt werden. Die Anwendung nimmt diese Eingaben entgegen und versucht, das Meeting im Kalender aller beteiligten Teilnehmer einzutragen.

Dabei prüft die interne Logik automatisch, ob bereits bestehende Aufgaben oder Termine mit dem gewünschten Zeitfenster kollidieren. Falls Überschneidungen festgestellt werden, sucht das System nach alternativen freien Zeitfenstern, um das Meeting möglichst ohne Konflikte einzuplanen. Wenn ein geeigneter Slot gefunden wird, wird das Meeting in den Kalendern der Teilnehmer eingetragen und alle Beteiligten erhalten entsprechende Benachrichtigungen.

Falls kein passender Termin verfügbar ist, generiert die Logik Warnungen oder Vorschläge, sodass der Benutzer informiert wird und gegebenenfalls einen anderen Zeitpunkt wählen kann. Gleichzeitig können betroffene Aufgaben automatisch neu geplant werden, um die Übersichtlichkeit und Verfügbarkeit im Kalender zu gewährleisten.

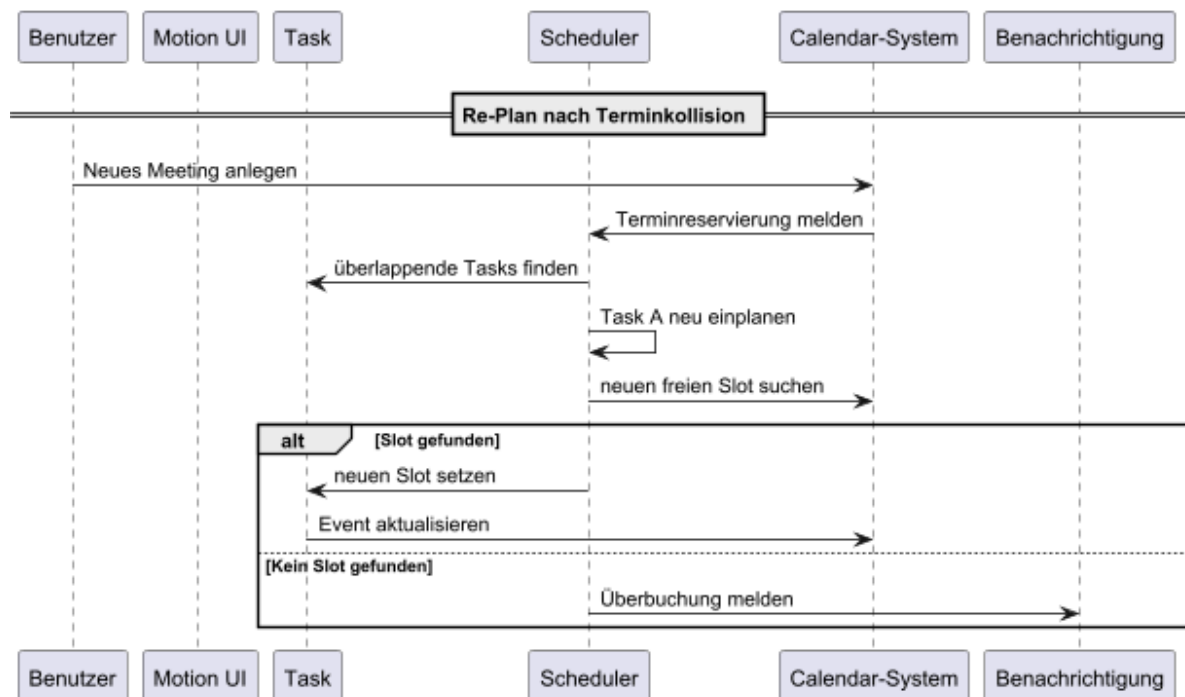


Abbildung 2: Sequenzdiagramm, welches annähernd beschreibt, wie ein Meeting in der Aufgabenmanagement-Plattform "Motion" angelegt wird.

## 5. Nachforschungen für den Scheduling- Algorithmus

### Circadiane Rhythmen

Der menschliche circadiane Rhythmus steuert über einen etwa 24-Stunden-Zyklus zentrale körperliche und psychologische Prozesse, darunter auch die kognitive Leistungsfähigkeit. Diese umfasst unter anderem Konzentration, Aufmerksamkeit und die Fähigkeit, Informationen zu verarbeiten sowie Probleme zu lösen.



Das Paper „*A time to think: Circadian rhythms in human cognition*“ von Schmidt et al. (2007) zeigt, dass kognitive Leistungen systematisch über den Tag hinweg schwanken und stark durch biologische Rhythmen beeinflusst werden.

### **Kognitive Leistungsfähigkeit**

Im Tagesverlauf zeigt sich, dass einfache kognitive Leistungen oft am Morgen ansteigen, während komplexere Denkprozesse ihren Höhepunkt eher am späten Vormittag oder um die Mittagszeit erreichen. Gegen Nachmittag kann die Leistung in einigen Bereichen abnehmen, auch wenn einfache oder motorische Fähigkeiten teilweise stabil bleiben.

Diese Muster sind jedoch nicht für alle gleich: Je nach Chronotyp („Morgenmensch“ vs. „Nachteule“) verschiebt sich der individuelle Leistungshorizont. Das Paper betont, dass kognitive Aufgaben besonders dann gut gelingen, wenn sie mit der persönlichen Hochphase der inneren Uhr übereinstimmen.[19], [20]

### **Relevanz für das Produkt**

Die Nachforschung zur kognitiven Leistungsfähigkeit hat direkten Einfluss auf den Scheduling-Algorithmus, welcher für die Planung der einzelnen Aufgaben innerhalb der Software zuständig ist.

Anhand der Arbeitszeiten des Nutzers wird geschätzt, wann die kognitiven Hochphasen des Mitarbeiters liegen. In diesen Zeitfenstern werden kognitiv anspruchsvolle Aufgaben (z. B. Datenanalyse, das Lösen komplexer Problemstellungen oder das Erarbeiten von Konzepten und Strategien) eingeplant. Umgekehrt werden kognitiv wenig anspruchsvolle Aufgaben (z. B. das Übertragen von Daten, das Sortieren von Dokumenten oder das Bearbeiten standardisierter E-Mails) in Phasen eingeordnet, in denen die kognitive Leistungsfähigkeit abnimmt.



## Quellen

- [1] „Timeflow | Smart calendar & task manager“. Zugegriffen: 17. März 2026. [Online]. Verfügbar unter: <https://timeflow.site/>
- [2] „The AI Powered SuperApp for Work | Motion“. Zugegriffen: 17. März 2026. [Online]. Verfügbar unter: <https://www.usemotion.com/>
- [3] „Smart Task Planning and Work Management | timehero.com“. Zugegriffen: 17. März 2026. [Online]. Verfügbar unter: <https://www.timehero.com/>
- [4] „LiquidPlanner“. Zugegriffen: 17. März 2026. [Online]. Verfügbar unter: <https://app.liquidplanner.com/>
- [5] C. Ramesh, „Moving off of TypeScript“, Medium. Zugegriffen: 17. März 2026. [Online]. Verfügbar unter: <https://engineering.usemotion.com/moving-off-of-typescript-e7bb1f3ad091>
- [6] M. Sellars, „We Thought React Native Was the Answer — Until Our App Hit 190% CPU“, Medium. Zugegriffen: 17. März 2026. [Online]. Verfügbar unter: <https://engineering.usemotion.com/we-thought-react-native-was-the-answer-until-our-app-hit-190-cpu-f0e849073334>
- [7] Z. Tarif, „Top 10 Backend-Frameworks des Jahres 2025“. Zugegriffen: 17. März 2026. [Online]. Verfügbar unter: <https://blog.back4app.com/de/die-besten-backend-frameworks-im-jahr-2025/>
- [8] „How One Engineer Saved Us \$110K and Ended Our Database Nightmare 🚀“, DEV Community. Zugegriffen: 17. März 2026. [Online]. Verfügbar unter: <https://dev.to/usemotion/how-one-engineer-saved-us-110k-and-ended-our-database-nightmare-1g1o>
- [9] S. Callahan, „Migrating to Postgres“, Medium. Zugegriffen: 17. März 2026. [Online]. Verfügbar unter: <https://engineering.usemotion.com/migrating-to-postgres-3c93dff9c65d>
- [10] S. Kowalczyk, „Replacing ClickOps with Pulumi“, Medium. Zugegriffen: 17. März 2026. [Online]. Verfügbar unter: <https://engineering.usemotion.com/replacing-clickops-with-pulumi-d21f3e80b851>
- [11] S. Walters, „Why I Never Want Another Standup“, Medium. Zugegriffen: 17. März 2026. [Online]. Verfügbar unter: <https://engineering.usemotion.com/why-i-never-want-another-standup-f858238b4bed>
- [12] „Was ist ein Virtual Private Server? – Funktion und Vorteile“. Zugegriffen: 17. März 2026. [Online]. Verfügbar unter: <https://www.badencloud.de/blog/virtual-private-server-vps/>
- [13] „Nutzen Sie die Vorteile von CDN | Server-Security“. Zugegriffen: 17. März 2026. [Online]. Verfügbar unter: <https://www.cloudflare.com/de-de/learning/cdn/cdn-benefits/>
- [14] „Welcome to Motion | Help Center“. Zugegriffen: 17. März 2026. [Online]. Verfügbar unter: <https://www.usemotion.com/help>
- [15] PrimeProductiv4, „Motion Review 2025: AI Calendar & Task Manager That Automates Your Perfect Day | Features & Pricing“, PrimeProductiv4.com. Zugegriffen: 17. März 2026. [Online]. Verfügbar unter: <https://www.primeproductiv4.com/apps-tools/motion-review>
- [16] „Auto-scheduling | Help Center“. Zugegriffen: 17. März 2026. [Online]. Verfügbar unter: <https://www.usemotion.com/help/time-management/auto-scheduling>
- [17] „Managing Notifications | Help Center“. Zugegriffen: 17. März 2026. [Online]. Verfügbar unter: <https://www.usemotion.com/help/settings/managing-notifications>
- [18] „Capacity Planning | Help Center“. Zugegriffen: 17. März 2026. [Online]. Verfügbar unter: <https://www.usemotion.com/help/project-management/capacity-planning>
- [19] C. Schmidt, F. Collette, C. Cajochen, und P. Peigneux, „A time to think: circadian rhythms in



human cognition“, *Cogn. Neuropsychol.*, Bd. 24, Nr. 7, S. 755–789, Okt. 2007, doi:  
10.1080/02643290701754158.

- [20] P. Valdez, C. Ramírez, und A. García, „Circadian rhythms in cognitive performance: implications for neuropsychological assessment“, *ChronoPhysiology Ther.*, Bd. 2, S. 81–92, Dez. 2012, doi:  
10.2147/CPT.S32586.