

**IU Internationale Hochschule**

Studiengang: Informatik M.Sc.

# **Phase 1 Konzeptionsphase: Projektdokumentation, Anforderungsdokument, Spezifikationsdokument**

Portfolio zur Prüfung im Kurs Projekt: Software Engineering (DLMCSPSE01\_D)

eingereicht von: Kevin Walter

Matrikel-Nr.: 92212082

Tutor: Prof. Dr.-Ing. David Kuhlen

Datum: 03.07.2025

# Inhaltsverzeichnis

1. Projektidee und angestrebtes Ergebnis.....	3
2. Potenzielle Risiken und Gegenmaßnahmen .....	3
2.1 Skala: .....	3
2.2 Tabelle potenzielle Risiken und Gegenmaßnahmen.....	3
3. Zeitplanung .....	4
4. Stakeholder (Ziel- und Benutzergruppe).....	5
5. Funktionale Anforderungen .....	5
5.1 Funktionsliste .....	5
5.2 User Stories.....	5
6. Nicht-funktionale Anforderungen .....	6
7. Glossar .....	6
8. Datenmodell.....	7
8.1 Aufgabe.....	7
8.2 UML-Klassendiagramm .....	7
9. Geschäftsprozesse .....	8
10. Geschäftsregeln.....	10
11. Systemschnittstellen .....	10
12. Benutzerschnittstellen .....	11
12.1 Struktur der Oberfläche .....	11
12.2 Wichtigste Dialoge & Abläufe .....	11
12.3 Skizze der Anwendung .....	12

## Phase 1: Konzeptionsphase - Projektdokument

### 1. Projektidee und angestrebtes Ergebnis

Bevor der Zweck dieser Anwendung genannt wird, möchte ich ein Problem aus meinem Alltag nennen. Ich plane meinen Tag bisher noch auf einem klassischen Notizblock, was auf lange Sicht nicht nur unpraktisch, sondern auch ressourcenverschwendend ist. Zwar gibt es die „Erinnerungen“-App für das iPhone, jedoch fehlt mir eine einfache und schlanke Anwendung für den PC, die mich in meinem Alltag unterstützt. Die gängigen digitalen Tools sind häufig entweder zu kompliziert oder mit unnötigen Zusatzfunktionen überladen.

Die Zielsetzung dieses Portfolios ist es daher, eine schlanke Desktop-Anwendung für Windows-PCs zu entwickeln, mit der man seine Aufgaben für den nächsten Tag (oder auch die gesamte Woche) möglichst effizient und übersichtlich aufschreiben kann. Die Bedienung soll dabei so einfach wie möglich gehalten werden, also ohne unnötig viele Menüs oder Reiter, sondern mit einem klaren Fokus auf tägliche Aufgabenplanung, Deadlines und einer Kalenderansicht.

### 2. Potenzielle Risiken und Gegenmaßnahmen

#### 2.1 Skala:

Wahrscheinlichkeit	niedrig	mittel	hoch
Schadensausmaß	niedrig	mittel	hoch

#### 2.2 Tabelle potenzielle Risiken und Gegenmaßnahmen

Risiko	Eintrittswahrscheinlichkeit	Schadensausmaß	Gegenmaßnahme
Einarbeitung in Qt und CMake dauert zu lange	mittel	mittel	Früh mit Einarbeitung beginnen, Beispiele als Orientierung verwenden
Release-Package funktioniert nicht auf anderen Rechnern	hoch	hoch	windeployqt.exe verwenden, damit alle DLLs und Dateien vorhanden sind
Projektumfang zu groß / Zeitmangel	gering	mittel	Realistischer Zeitplan, Fokus auf Kernfunktionen (sh. Abschnitt 1)
Fehler durch Pointer / Referenzen	mittel	hoch	Einfacher Code, Debugging-Tools und Testing
Codeverlust / verschiedene Versionen	gering	hoch	Regelmäßige Pushs zu GitHub

### 3. Zeitplanung

[illegible]

## **Phase 1: Konzeptionsphase - Anforderungsdokument**

### **4. Stakeholder (Ziel- und Benutzergruppe)**

Die Anwendung richtet sich an alle Nutzer, die ihren Tag gerne strukturiert planen möchten, unabhängig davon ob sie Studenten, Berufstätige oder Privatpersonen sind. Das Programm bietet einen schnellen und unkomplizierten digitalen Ersatz für einen analogen Notizblock und unterstützt dabei, einen Überblick über anstehende Aufgaben und Deadlines zu behalten. Das Ziel ist es, die Vorteile moderner Aufgabenverwaltung am PC mit der Einfachheit eines Papier-Notizblocks zu verbinden, um papierlos und effizienter planen zu können.

### **5. Funktionale Anforderungen**

#### **5.1 Funktionsliste**

- Aufgaben anlegen, bearbeiten, löschen
- Deadlines setzen
- Aufgaben als erledigt markieren
- Kalender-/Tagesansicht
- Daten speichern/laden

#### **5.2 User Stories**

Als Benutzer möchte ich einfach per Texteingabe mit der Tastatur Aufgaben anlegen, bearbeiten und löschen können.

Als Benutzer möchte für meine Aufgabe auch ein Fälligkeitsdatum setzen können, damit ich mir nicht selbst notieren muss, bis wann ich die Aufgabe erledigen will. Es ist vorerst nicht vorgesehen, eine Benachrichtigung zu versenden, dass die Aufgabe überfällig ist. Über das heutige Datum und dem notierten Fälligkeitsdatum kann ich mir das selbst herleiten.

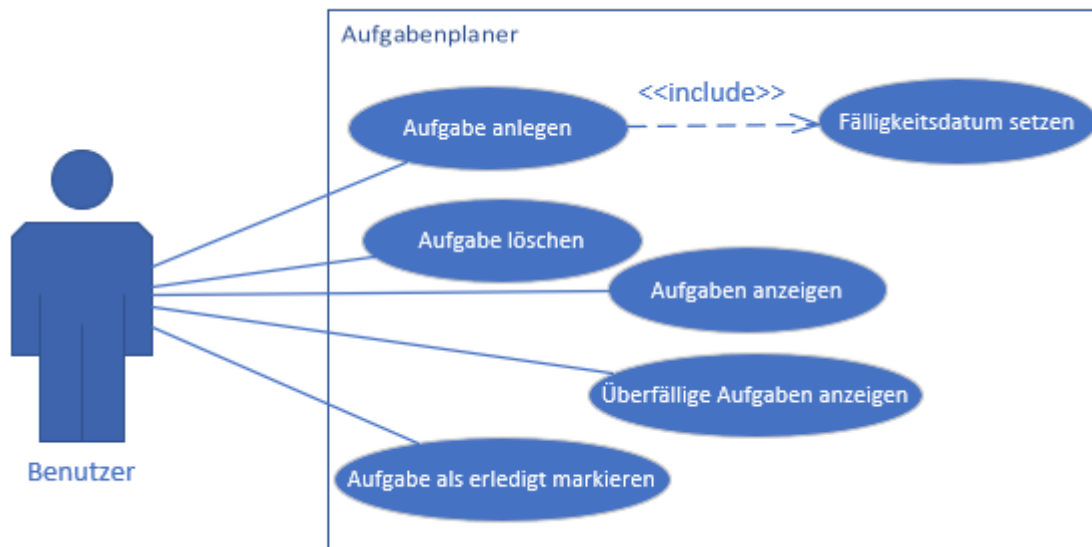
Als Benutzer möchte ich die Aufgabe als erledigt markieren können, damit ich direkt sehen kann, welche Aufgaben ich heute noch vor habe.

Als Benutzer möchte ich eine Kalender- und Tagesansicht sehen können, damit ich so beispielsweise auch eine ganze Woche planen kann.

Als Benutzer möchte ich die Daten speichern können, das heißt entweder per „save“ Button, oder mit automatischer Speicherung.

Als Benutzer muss ich mich nicht anmelden oder registrieren. Die Anwendung funktioniert wie ein klassisches Notizbuch: Ich kann direkt starten, ohne Account oder Internetverbindung. Die Aufgaben werden lokal auf meinem Windows-PC gespeichert. Eine Synchronisierung mit einer Cloud ist nicht vorgesehen.

Abbildung 1: Use-Case-Diagramm



Quelle: Eigene Darstellung mit Hilfe von Visio

## 6. Nicht-funktionale Anforderungen

- **Benutzerfreundlichkeit:** Die Anwendung soll eine einfache Bedienung und eine übersichtliche Oberfläche besitzen
- **Performance:** Die Anwendung ist schlank und überzeugt durch schnelle Ladezeiten, z. B. keine langen Ladezeiten für Reiter oder Seiten
- **Datensicherheit:** Alle Daten werden lokal auf dem PC gespeichert, es besteht keine Internetverbindung und keine Cloud-Synchronisierung. Dadurch sind die Daten bestmöglich vor externen Angriffen geschützt
- **Verschlüsselung:** Eine Verschlüsselung der Daten ist vorerst nicht vorgesehen
- **Plattform:** Die Anwendung wird als Windows Desktop-Anwendung bereitgestellt und nicht als Web-App
- **Eingabevalidierung:** Es wird eine Eingabevalidierung für Pflichtfelder und sinnvolle Werte umgesetzt

## 7. Glossar

- **Aufgabe:** Ein Eintrag mit Titel, Beschreibung und Fälligkeitsdatum
- **Tagesansicht:** Auflistung aller an diesem Tag zu erledigenden/eingetragenen Aufgaben
- **Deadline:** Datum, bis zu dem die Aufgabe abgeschlossen sein soll
- **GUI:** Grafische Benutzeroberfläche
- **User Story:** Beschreibung aus Anwendersicht
- **UML:** Unified Modeling Language

## Phase 1: Konzeptionsphase - Spezifikationsdokument

### 8. Datenmodell

#### 8.1 Aufgabe

- Es ist das wichtigste Objekt
- Attribute einer Aufgabe sind: Titel, Beschreibung, Fälligkeitsdatum, Status (offen/erledigt) mit Anzeigen in grün/rot
- Benötigt mindestens einen Titel (Pflichtfeld), ein optionales Fälligkeitsdatum und einen Status (offen/erledigt), kann zusätzliche Beschreibung haben

#### 8.2 UML-Klassendiagramm

Abbildung 2: UML-Klassendiagramm



Quelle: Eigene Darstellung mit Hilfe von Visio

## **9. Geschäftsprozesse**

### **Geschäftsprozess 1: Aufgabe anlegen**

1. Der Benutzer startet die Anwendung
2. Im Hauptfenster klickt er auf „Aufgabe hinzufügen“
3. Es öffnet sich ein Modal
  - Der Benutzer gibt Titel, Beschreibung, Fälligkeitsdatum ein
  - Mit „Speichern“ wird die Aufgabe übernommen
4. Die neue Aufgabe erscheint in der Aufgabenliste

### **Geschäftsprozess 2: Aufgabe als erledigt markieren**

1. Benutzer sieht die Aufgabenliste
2. Bei einer Aufgabe klickt er auf „Aufgabe als erledigt markieren“
3. Die Aufgabe wird als erledigt markiert und ggf. anders dargestellt (je nach Qt Funktionen)

### **Geschäftsprozess 3: Überfällige Aufgaben anzeigen**

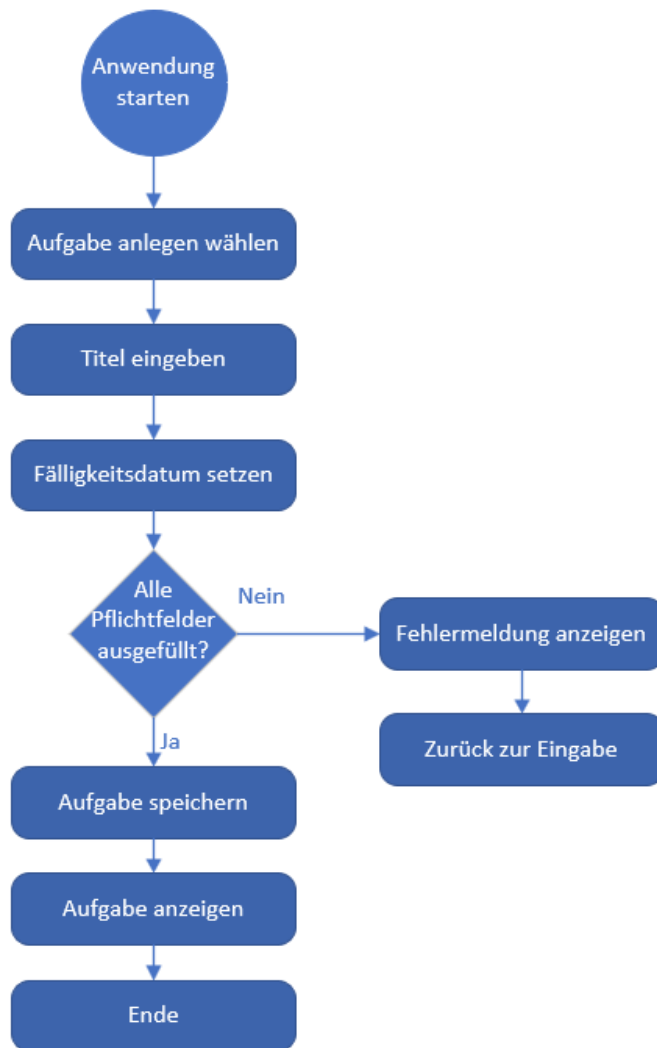
1. Beim Start prüft das Programm alle Aufgaben
2. Aufgaben mit Fälligkeitsdatum < wie aktuelles Datum werden als „überfällig“ markiert/angezeigt

### **Geschäftsprozess 4: Aufgabe löschen**

1. In der Aufgabenliste wählt der Nutzer eine Aufgabe aus
2. Mit Klick auf „Aufgabe löschen“ wird die Aufgabe entfernt (komplett gelöscht)



Abbildung 3: UML-Aktivitätsdiagramm für Kernprozess "Aufgabe anlegen"



Quelle: Eigene Darstellung mit Hilfe von Visio

## **10. Geschäftsregeln**

- Eine Aufgabe muss immer mindestens einen Titel besitzen
- Das Fälligkeitsdatum einer Aufgabe darf nicht in der Vergangenheit liegen
- Eine Aufgabe kann erst gespeichert werden, wenn alle Pflichtfelder (Titel) ausgefüllt sind
- Jede Aufgabe ist eindeutig identifizierbar (z.B. durch eine ID)
- Bereits erledigte Aufgaben können nicht bearbeitet werden (nur noch löschen möglich)
- Das Fälligkeitsdatum von bereits erledigten Aufgaben führt nicht mehr zu einem Zustandswechsel (bspw. Verändern der Schrift von grün auf rot o.Ä.)
- Beim Löschen einer Aufgabe öffnet sich ein Modal, um unbeabsichtigtes löschen zu verhindern
- Alle angegebenen Daten werden beim Schließen der Anwendung gespeichert
- Das Fälligkeitsdatum ist optional, aber wenn es gesetzt wird, muss es nach dem Erstellungsdatum liegen

## **11. Systemschnittstellen**

In diesem Projekt sind keine externen technischen Schnittstellen (z.B. HTTP, FTP) vorgesehen. Die Anwendung läuft komplett lokal auf dem Rechner des Benutzers. Der Datenformat der Aufgaben wird zum Beispiel TXT, CSV oder JSON sein.

## **12. Benutzerschnittstellen**

Die Anwendung bietet eine schlanke grafische Benutzeroberfläche (GUI), die für einfache Bedienung und Übersichtlichkeit ausgelegt ist, ohne viele Reiter / Dialoge.

### **12.1 Struktur der Oberfläche**

- Hauptfenster: Zeigt alle Aufgaben übersichtlich als Liste
- Neue Aufgabe anlegen: Button „+ Aufgabe anlegen“ öffnet Dialog
- Aufgabendetails: Im Dialog kann der Nutzer die Attribute eingeben
- Löschen/Erledigt: Neben jeder Aufgabe befinden sich Buttons für „Löschen“ oder „als erledigt markieren“
- Kalender: Der Kalender soll im Hauptfenster angezeigt werden

### **12.2 Wichtigste Dialoge & Abläufe**

#### **Neue Aufgabe anlegen**

- Klick auf „+ Aufgabe hinzufügen“
- Eingabefeld für Titel (Pflichtfeld)
- Eingabefeld für Fälligkeitsdatum (optional)
- Eingabefeld für Beschreibung (optional)
- „Speichern“ Button, der nur aktiv ist, wenn das Pflichtfeld ausgefüllt ist
- Fehlermeldung, falls Pflichtfeld leer bleibt oder Datum ungültig ist

#### **Aufgabe löschen**

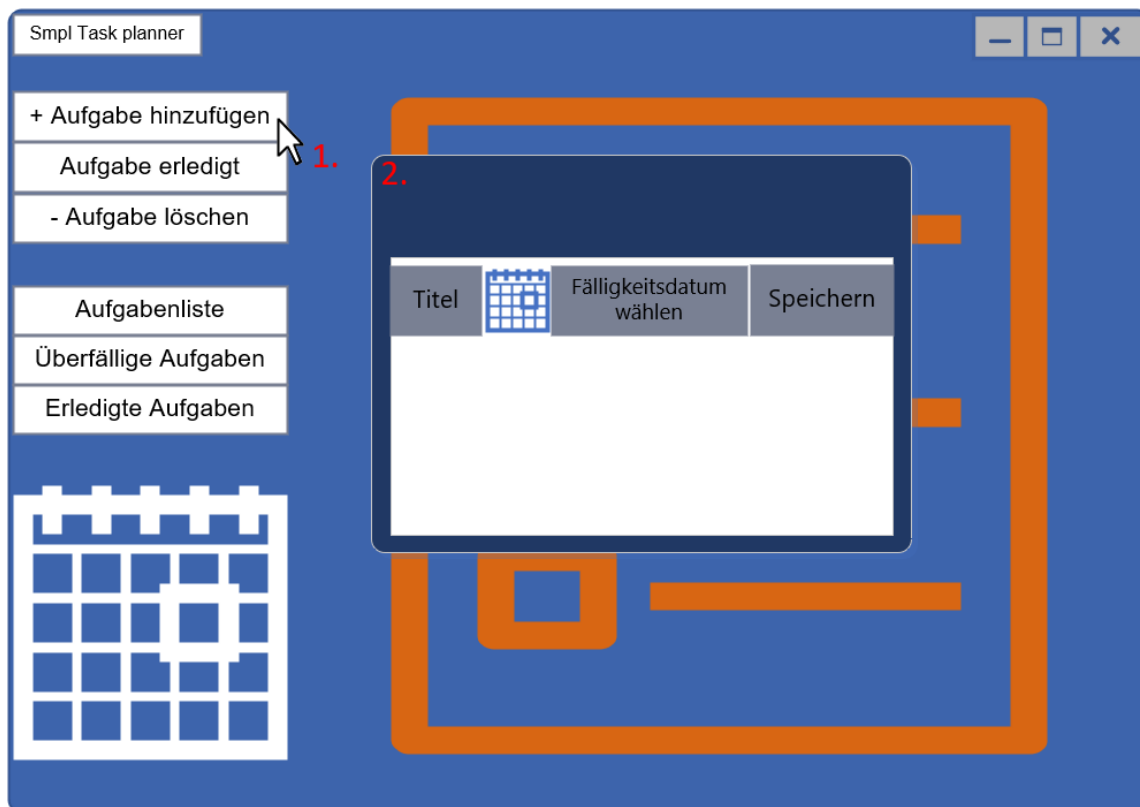
- Klick auf „Löschen“-Button öffnet ein Modal/Bestätigungsdiallog und erst danach wird die Aufgabe gelöscht

#### **Eingabevalidierung**

- Titel muss ausgefüllt sein
- Fälligkeitsdatum darf nicht in der Vergangenheit liegen

### 12.3 Skizze der Anwendung

Abbildung 4: Skizze Aufgabenplaner



Quelle: Eigene Darstellung mit Hilfe von Visio