
Software Engineering

**Anforderungsanalyse zur Entwicklung eines SW-Systems zur
Unterstützung der Einführung von Gleitarbeitszeit**

vorgelegt von

Tom Graupner
Markus Klemm
Leonard Hecker

Inhaltsverzeichnis

| | | |
|-------|--|----|
| 1 | Einführung | 3 |
| 2 | Dokumentation der Anforderungen | 4 |
| 2.1 | Funktionale Anforderungen | 4 |
| 2.1.1 | Tabellarischer Überblick. | 5 |
| 2.1.2 | Struktur der Eingangs- und Ausgangedaten | 7 |
| 2.2 | Qualitätsanforderungen. | 10 |
| 2.3 | Rahmenbedingungen | 11 |
| 3 | Kontextdiagramm | 12 |
| 4 | Anwendungsfalldiagramme | 13 |
| 4.1 | AWD der groben Funktionalität | 13 |
| 4.2 | AWD der Funktionalität XY. | 13 |
| 4.3 | Detaillierte Beschreibung der essenziellen Funktionalität XY | 13 |
| 5 | Zustandsdiagramm eines Urlaubsantrages | 14 |
| 6 | Entity Relationship Model | 15 |
| 7 | Glossar | 16 |

1 Einführung

Das Unternehmen EKS¹ evaluiert aktuell die Umstellung ihres Arbeitszeitmodells zur Gleitzeit. Die Erfassung und Auswertung der Arbeitszeit soll dabei durch ein Software-System unterstützt werden. Die vorliegende Anforderungsanalyse beschäftigt sich zunächst mit den Rahmenbedingung und den Funktionen, die vom System übernommen werden sollen. Neben der Zusammenfassung aller funktionalen Anforderungen und der Struktur der Eingangs- und Ausgangsdaten, enthält diese Analyse verschiedene Anwendungsfalldiagramme², sowie ein Entity Relationship Model, welches die Speicherung der Daten veranschaulicht.

1 Abkürzung für ENTWICKLUNG VON KUNDENSPEZIFISCHER SOFTWARE

2 Als Abkürzung wird im folgenden AWD verwendet. Daran angelehnt ist die Abkürzung AWF für einen Anwendungsfall

2 Dokumentation der Anforderungen

Anforderungen an ein Software-Produkt werden im Allgemeinen zunächst in funktionale und nicht-funktionale Anforderungen unterteilt. Erstere decken dabei die Fähigkeiten und die Beschaffenheiten ab, die der Benutzer der Software zur Problemlösung oder zur Erreichung seines Zieles benötigt. Nicht-funktionale Anforderungen unterteilen sich weiterhin in Rahmenbedingungen und Qualitätsanforderungen.

2.1 Funktionale Anforderungen

Die folgende Auflistung enthält die groben Funktionen, die vom Software-System erfüllt werden sollen. Bei einigen handelt es sich dabei um *abstrakte Funktionen*, welche sich im weiteren Verlauf der Analyse feiner aufgliedern werden.

- **Anwesenheit erfassen** « *abstrakt* »
- **Urlaub verwalten - Mitarbeiter** « *abstrakt* »
- **Urlaub planen - Abteilungsleiter** « *abstrakt* »
- **Krankheitsdaten erfassen**
- **Anwesenheit auswerten**
- **Zeitauswertung für Abteilungsleiter** « *abstrakt* »

2.1.1 Tabellarischer Überblick

Die folgenden Tabellen fassen nun alle voneinander unabhängigen funktionalen Anforderungen an das Software-System zusammen. Im Rahmen der Anforderungsanalyse verwendet man für unabhängige funktionale Anforderungen ebenfalls den Begriff *essentielle Funktionen*.

| Funktion | Eingangsdaten | Ausgangsdaten | Bemerkungen | abstrakter AWD |
|-------------------------------|-------------------|---|---|-----------------------------|
| <i>Betreten</i> | MA-ID und Uhrzeit | Zutritt und Speicherung der Zeit, alternativ Zutrittsverweigerung | Bei einer ungültigen MA-ID kann der Zutritt verweigert werden | Anwesenheit erfassen |
| <i>Verlassen</i> | MA-ID und Uhrzeit | Verlassen und Speicherung der Zeit, alternative Fehlermeldung | | |
| <i>Wachdienst informieren</i> | Mitarbeiterliste | Detaillierte Information an den Wachdienst | Der Wachdienst wird stündlich darüber informiert, welche Mitarbeiter sich im Gebäude befinden | |

| | | | | |
|--------------------------------------|---|---|---|--------------------------------------|
| <i>Urlaub beantragen</i> | Urlaubswunsch | Urlaubsantrag | Urlaub wird unter Verwendung der eigenen MA-ID beim jeweiligen Abteilungsleiter beantragt | Urlaub verwalten, Mitarbeiter |
| <i>Urlaubsinformationen anzeigen</i> | Wunsch nach Urlaubsinformationen | Informationen zu Urlaubsterminen, Beantragungsstatus, verbrauchten und verbleibenden Urlaubstagen | | |
| <i>Urlaubsantrag stornieren</i> | Storno-Wünsche | Storno-Bestätigung mit Aktualisierung der Urlaubsdaten | Mitarbeiter kann offene, abgelehnte und genehmigte (noch nicht angetretene) Urlaubsanträge stornieren | |
| <i>Urlaubsvorschlag annehmen</i> | Urlaubsvorschlag des Abteilungsleiter | Aktualisierung der Urlaubsinformationen | Abteilungsleiter können Mitarbeiter ihrer Abt. Vorschläge unterbreiten | |
| <i>Urlaubsvorschlag stornieren</i> | Urlaubsvorschlag des Abteilungsleiter und Storno-Wunsch | Stornierungsmitteilung und Aktualisierung der Urlaubsinformationen | Abteilungsleiter können Mitarbeitern ihrer Abt. Vorschläge unterbreiten | |

| Funktion | Eingangsdaten | Ausgangsdaten | Bemerkungen | abstrakter AWD |
|---|---|---|--|-----------------------------------|
| <i>Urlaubsantrag genehmigen</i> | Urlaubsantrag eines MA | Aktualisierung der Urlaubsdaten und Bestätigung | Abteilungsleiter müssen Anträge ihrer Mitarbeiter genehmigen | Urlaub planen, Abt.-Leiter |
| <i>Urlaubsantrag ablehnen</i> | Urlaubsantrag eines MA | Aktualisierung der Urlaubsdaten und Absage | Abteilungsleiter können Anträge ihrer Mitarbeiter ablehnen | |
| <i>Vorschlag unterbreiten</i> | Urlaubsvorschlag des Abteilungsleiters | Urlaubsvorschlag an Mitarbeiter und Aktualisierung der Urlaubsinformationen | Abteilungsleiter können Mitarbeitern Urlaubsvorschläge unterbreiten | |
| <i>Urlaubsinformationen eines Mitarbeiters anzeigen</i> | Wunsch des Abteilungsleiters nach Urlaubsinformationen eines Mitarbeiters | Detaillierte Informationen zur Abteilung | Abteilungsleiter können sich zur Entscheidungsunterstützung die Urlaubsinformationen eines Mitarbeiters anzeigen lassen | |
| <i>Krankmeldung erfassen</i> | Krankenschein eines Mitarbeiters | Aktualisierung der Urlaubsinformationen | Sachbearbeiter (HR) erfasst Krankmeldungen von Mitarbeitern und betroffene Urlaubsinformationen werden sofort aktualisiert | |
| <i>Anwesenheit auswerten</i> | Anwesenheitsinformationen eines Mitarbeiters | Detaillierte Arbeitszeitauswertung des Mitarbeiters | Die Auswertung wird wöchentlich automatisch erstellt und dem Mitarbeiter per Email zugesandt | |

| Funktion | Eingangsdaten | Ausgangsdaten | Bemerkungen | abstrakter AWD |
|------------------------------------|-------------------------------|---|--|---------------------------------------|
| <i>Gesamtbilanz anfordern</i> | Wunsch nach Gesamtbilanz | Gesamtbilanz enthält detaillierte Informationen zur Arbeitszeitauswertung der Abteilung | Die Kennzahlen sind absolut und prozentual angegeben und betreffen einen beliebigen, abgelaufenen Zeitraum | Zeitauswertung für Abt.-Leiter |
| <i>Urlaubszeitbilanz anfordern</i> | Wunsch nach Urlaubsbilanz | Urlaubszeitbilanz enthält beantragte Urlaubstage der Abteilung in einem vorausschauenden Zeitraum | Anträge werden absolut und prozentual bezogen auf die Gesamtarbeitszeit dargestellt | |
| <i>Anwesenheitsliste anfordern</i> | Wunsch nach Anwesenheitsliste | Liste enthält alle momentan anwesenden Mitarbeiter der eigenen Abteilung | | |

2.1.2 Struktur der Eingangs- und Ausgangedaten

Jede essentielle Funktion besitzt definierte Eingangs- und Ausgangsdaten. Die folgende Auflistung stellt die Struktur der entsprechenden Daten aller zuvor genannten essentiellen Funktionen dar.

| Funktion | Struktur der Eingangsdaten | Struktur der Ausgangsdaten |
|-------------------------------|---|---|
| <i>Betreten</i> | + MA-ID + Uhrzeit | Bestätigung des Zutritts und Datensatz := {MA-ID, Uhrzeit}, alternativ Fehlermeldung |
| <i>Verlassen</i> | + MA-ID + Uhrzeit | Bestätigung des Verlassen und Datensatz := {MA-ID, Uhrzeit}, alternativ Fehlermeldung |
| <i>Wachdienst informieren</i> | Stündlicher Trigger zum Auslösen der Benachrichtigung | Detaillierte Mitarbeiterliste mit den Spalten {MA-ID, Nachname, Vorname, Büro} |

| Funktion | Struktur der Eingangsdaten | Struktur der Ausgangsdaten |
|---|---|--|
| <i>Urlaub beantragen</i> | Urlaubswunsch := {MA-ID, Liste: zu beantragende Urlaubstage} | Urlaubsantrag := {MA-ID, Nachname, Vorname, Liste: zu beantragende Urlaubstage} |
| <i>Urlaubsinformationen anzeigen</i> | Wunsch nach Urlaubsinformationen | Urlaubsinformationen := {verbrauchte Urlaubstage, verbleibende Urlaubstage, Liste: Urlaubstermine inkl. Status (offen, genehmigt, abgelehnt)} |
| <i>Urlaubsantrag stornieren</i> | Urlaubsantrag s.o. (Status: offen / genehmigt und noch nicht angetreten). | Bestätigung der Stornierung und aktualisierte Urlaubsinformationen := {verbrauchte Urlaubstage, verbleibende Urlaubstage, Liste: Urlaubstermine inkl. Status (offen, genehmigt, abgelehnt)} |
| <i>Urlaubsvorschlag annehmen</i> | Urlaubsvorschlag := {MA-ID, Liste: vorgeschlagener Urlaubstage } | Bestätigung des Urlaubsvorschlages und aktualisierte Urlaubsinformationen := {verbrauchte Urlaubstage, verbleibende Urlaubstage, Liste: Urlaubstermine inkl. Status (offen, genehmigt, abgelehnt)} |
| <i>Urlaubsvorschlag stornieren</i> | Urlaubsvorschlag s.o. | Bestätigung des Urlaubsvorschlages und aktualisierte Urlaubsinformationen := {verbrauchte Urlaubstage, verbleibende Urlaubstage, Liste: Urlaubstermine inkl. Status (offen, genehmigt, abgelehnt)} |
| <i>Urlaubsantrag genehmigen</i> | Urlaubsantrag s.o. | Bestätigung der Genehmigung und aktualisierte Urlaubsinformationen := {verbrauchte Urlaubstage, verbleibende Urlaubstage, Liste: Urlaubstermine inkl. Status (offen, genehmigt, abgelehnt)} |
| <i>Urlaubsantrag ablehnen</i> | Urlaubsantrag s.o. | Bestätigung der Ablehnung und aktualisierte Urlaubsinformationen := {verbrauchte Urlaubstage, verbleibende Urlaubstage, Liste: Urlaubstermine inkl. Status (offen, genehmigt, abgelehnt)} |
| <i>Urlaubsvorschlag unterbreiten</i> | Wunsch einen Vorschlag zu unterbreiten oder konkreter Urlaubsantrag s.o. | Urlaubsvorschlag := {MA-ID, Nachname, Vorname, Liste: vorgeschlagener Urlaubstage} und aktualisierte Urlaubsinformationen := {verbrauchte Urlaubstage, verbleibende Urlaubstage, Liste: Urlaubstermine inkl. Status (offen, genehmigt, abgelehnt)} |
| <i>Urlaubsinformationen eines Mitarbeiters anzeigen</i> | Wunsch nach Urlaubsinformationen | Urlaubsinformationen := {verbrauchte Urlaubstage, verbleibende Urlaubstage, Liste: Urlaubstermine inkl. Status (offen, genehmigt, abgelehnt)} |

| Funktion | Struktur der Eingangsdaten | Struktur der Ausgangsdaten |
|------------------------------------|---|---|
| <i>Krankmeldung erfassen</i> | Krankenschein eines Mitarbeiters := {Nachname, Vorname, Geburtsdatum, Zeitraum der Bescheinigung} | Entsprechende Aktualisierung der Mitarbeiterdatensätze für Urlaubstage, Soll- und Ist-Arbeitszeit |
| <i>Anwesenheit auswerten</i> | Arbeitsstunden, Urlaubstage und Krankmeldungen des Mitarbeiters | Detaillierte Arbeitszeitauswertung := {Soll-Arbeitszeit, Ist-Arbeitszeit, Stand des Arbeitszeitkontos } |
| <i>Gesamtbilanz anfordern</i> | Wunsch nach Gesamtbilanz | Gesamtbilanz zu Abteilung := {Sollarbeitszeit, tatsächliche Arbeitsstunden, Urlaubstage, Krankheitstage, Überstunden} |
| <i>Urlaubsbilanz anfordern</i> | Wunsch nach Urlaubsbilanz | Urlaubsbilanz der Abteilung := {beantragte Urlaubstage (absolut und prozentual bezogen auf Gesamtarbeitszeit)} |
| <i>Anwesenheitsliste anfordern</i> | Wunsch nach Anwesenheitsliste | Anwesenheitsliste der Abteilung := {MA-ID, Nachname, Vorname, Arbeitsplatz} |

2.2 Qualitätsanforderungen

Nachdem weder interne, noch externe Qualitätsanforderungen explizit in den vorliegenden Rahmenbedingungen genannt sind, lautet die Aufgabe hier globale Anforderungen zu formulieren und eigene Gedanken zu entwickeln.

Ein allgemeiner Punkt herausragender Bedeutung ist beispielsweise *Datensicherheit und Integrität*. Aufgrund der Sensibilität der zu verarbeitenden Daten und der mit ihnen verbundenen Business-Prozesse (e.g. Buchhaltung) ist unbedingt dafür zu sorgen, dass jegliche Daten *zugriffssicher, redundant* und unter *definierten Integritätsbestimmungen* gespeichert und verarbeitet werden.

Für die spätere Erweiterung oder Wartung der Software ist es außerdem von großer Bedeutung, alle Funktionen und Komponenten des Systems lückenlos zu dokumentieren.

Geht man etwas ins Detail und betrachtet die essentiellen Funktionen, so gibt es Punkte an denen die Benutzerfreundlichkeit deutlich verbessert werden kann. Empfohlen wären unter anderem *Interaktionen mit der Software zu bestätigen*. Gemeint ist damit, dem Benutzer Rückmeldung zu erfolgreich oder nicht erfolgreich abgeschlossenen Interaktionen zu geben.

Weitere vorstellbare Qualitätsanforderungen werden nach Bedarf mit dem Auftraggeber abgesprochen.

2.3 Rahmenbedingungen

Als abschließender Punkt der schriftlichen Formulierung der Anforderungen werden die Rahmenbedingungen festgehalten. Hierbei unterscheidet man zwischen *technologischen*, *rechtlichen* und *organisatorischen Rahmenbedingungen*.

Zu den *technischen Rahmenbedingungen* gehört dabei, dass das System Zugriff auf den betriebsinternen Jahreskalender benötigt. Dies ist notwendig, um Feiertage und Betriebsruhetage automatisch in die Bilanz der Arbeitszeitkonten einbeziehen zu können. Weiterhin sollen Urlaubstage und Krankmeldungen unmittelbar in die Bilanz einfließen.

Wichtigster Teil der *rechtlichen Rahmenbedingungen* ist zweifelsohne das Thema Datensicherheit. Die Vollständigkeit und Integrität der personenbezogenen Daten muss zu jedem Zeitpunkt gewährleistet sein. Dies ist notwendig um Rechtssicherheit zu schaffen, für den Arbeitgeber und den Arbeitnehmer.

Die *organisatorischen Rahmenbedingungen* beinhalten vor allem Details zu den Arbeitszeitmodellen im Unternehmen. So besitzt ein Standard-Arbeitstag 8 Stunden und eine Arbeitswoche dementsprechenden 40 Stunden. Das Arbeitszeitkonto eines jeden Mitarbeiters wird dabei vom Beginn des Arbeitsverhältnisses an kumulativ geführt.

3 Kontextdiagramm

Todo: Markus

4 Anwendungsfalldiagramme

Todo: Markus

4.1 AWD der groben Funktionalität

Todo: Markus

4.2 AWD der Funktionalität XY

Todo: Markus

4.3 Detaillierte Beschreibung der essenziellen Funktionalität XY

Todo: Markus

5 Zustandsdiagramm eines Urlaubsantrages

Todo: Leonard

6 Entity Relationship Model

Todo: Leonard

7 Glossar

Todo: Tom und Markus und Leonard

- Anwendungsfall
- Anwendungsfalldiagramm
- abstrakte Funktion
- essentielle Funktion
- Entity Relationship Model
- Unified Modeling Language
- Mitarbeiter-ID
- t.b.c
-
-