

# Software Requirements Specifications (SRS) Project Tracker

Schalk

Version	Autor	Dateiname	Letzte Änderung	Druckdatum	Seite
1.0	Schalk	SRS-Project-Tracker.pdf	10.06.2023	12.06.2023	1 von 8

# Inhaltsverzeichnis

1 Vision .....	3
1.1 Vision und Kurzbeschreibung des Projekts .....	3
2 Grobe Spezifikation .....	3
2.1 Zusammenhang mit bereits bestehenden Systemen .....	3
2.2 Überblick über die geforderte Funktionalität .....	4
2.3 Wesentliche Qualitätsanforderungen und Rahmenbedingungen .....	4
3 Detaillierte Spezifikation .....	4
3.1 Systemabgrenzung, Systemarchitektur und Datenhaltung .....	4
3.2 Akteure des Systems (Personas) .....	5
3.3 Detaillierte funktionale Anforderungen (Szenarios & Screens) .....	5
3.3.1 Datenbank Login .....	5
3.3.2 User Login .....	5
3.3.3 Projekte verwalten .....	5
3.3.4 Zeit stoppen .....	5
3.3.5 CSV-Export .....	6
3.3.6 Persönliches Passwort ändern .....	6
3.3.7 Benutzerverwaltung (betrifft Administrator) .....	6
3.3.8 Datenbank wiederherstellen (betrifft Administrator) .....	6
3.4 Schnittstellen .....	6
3.4.1 Benutzerschnittstellen (GUI) .....	6
3.4.2 Systemschnittstellen .....	7
3.5 Nicht-Funktionale Anforderungen .....	7
3.5.1 Vorgaben zur Hardware und Software .....	7
3.5.2 Performance .....	7
3.5.3 Resources und Anforderungen an die Hardware .....	7
3.5.4 Security & Safety .....	7
3.5.5 Reliability .....	7
3.5.6 Maintenance .....	7
3.5.7 Portability / Skalierbarkeit / Wiederverwendbarkeit .....	7
3.5.8 Useability .....	7
3.6 Rahmenbedingungen .....	7
3.7 Lieferumfang .....	7

Version	Autor	Dateiname	Letzte Änderung	Druckdatum	Seite
1.0	Schalk	SRS-Project-Tracker.pdf	10.06.2023	12.06.2023	2 von 8

# 1 Vision

## 1.1 Vision und Kurzbeschreibung des Projekts

Das Projekt „Projekt Tracker“ zielt darauf ab, ein leistungsstarkes und benutzerfreundliches Zeitmanagement-System zu entwickeln, das es Organisationen und Einzelpersonen ermöglicht, den zeitlichen Aufwand für Projekte effizient zu erfassen, zu verwalten und auszuwerten. Die Anwendung soll eine intuitive Benutzeroberfläche bieten, die die Erstellung, Verwaltung und detaillierte zeitliche Analyse von Projekten ermöglicht.

## 2 Grobe Spezifikation

### 2.1 Zusammenhang mit bereits bestehenden Systemen

Der „Project Tracker“ ist ein eigenständiges System, das unabhängig von anderen Systemen funktioniert. Die Anwendung wird in Java entwickelt und verwendet eine MySQL-Datenbank zur Speicherung der Projektdaten. Die Systemgrenzen sind klar definiert und umfassen Funktionen wie das Anlegen und Verwalten von Projekten, die Verwaltung von Benutzerkonten, die Erfassung von Arbeitsstunden und das Exportieren von projektbezogenen Daten als CSV-Format. Diese Anwendung ist so konzipiert, dass sie in bestehende Arbeitsabläufe integriert werden kann, ohne dass eine umfangreiche Anpassung erforderlich ist. Die Spezifikation gilt für das gesamte Projekt und legt den Rahmen für die Entwicklung, Implementierung und Wartung der Anwendung fest. Diese Spezifikation dient als Leitfaden, um ein Verständnis der Ziele und Anforderungen des Projekts zu gewährleisten.

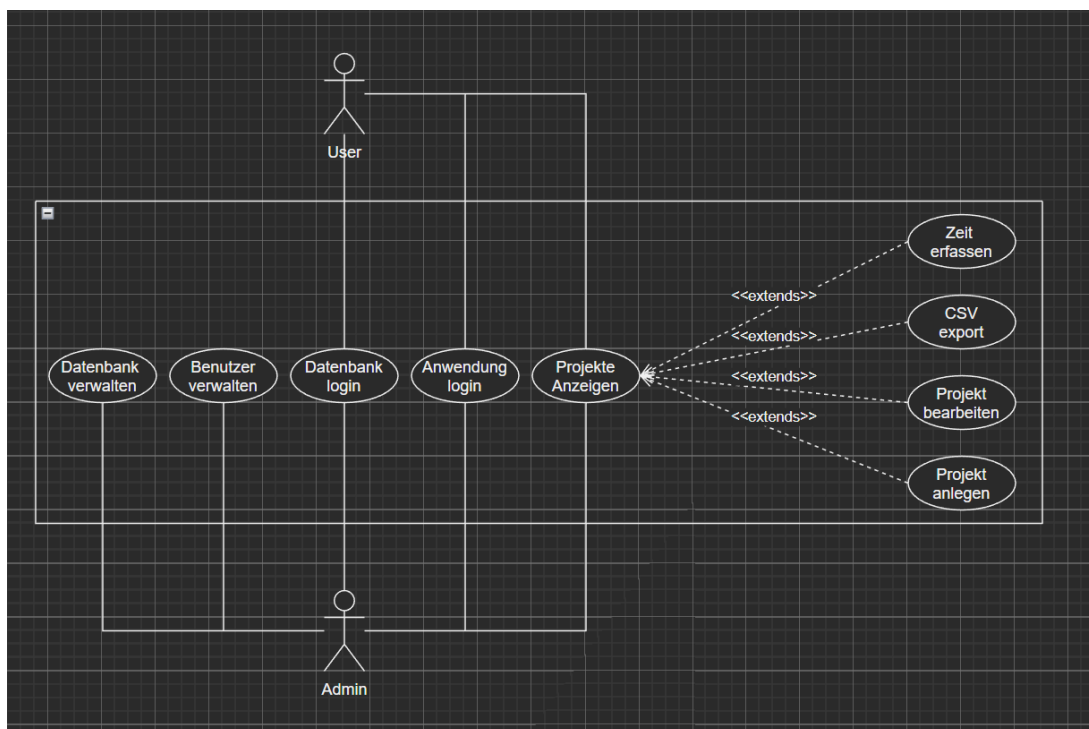


Abbildung 1: Hier ist das Use-Case-Diagramm, das die Hauptfunktionen des „Project Tracker“ darstellt.

Das Diagramm zeigt die Haupt-Use-Cases der „Project Tracker“ Anwendung und deren Beziehung zum Anwender. Die Anwendung ermöglicht es dem Anwender, Projekte anzulegen und zu verwalten, Benutzerkonten zu verwalten, Arbeitsstunden zu erfassen, Auswertungen zu erstellen sowie Daten zu importieren und zu exportieren.

Version	Autor	Dateiname	Letzte Änderung	Druckdatum	Seite
1.0	Schalk	SRS-Project-Tracker.pdf	10.06.2023	12.06.2023	3 von 8

## 2.2 Überblick über die geforderte Funktionalität

Das „Project Tracker“ System bietet eine umfassende Lösung zur Erfassung und Verwaltung von Projektinformationen und Arbeitsstunden. Die Hauptfunktionen der Anwendung umfassen:

1. *Anwendung-Login*: Ermöglicht passwortgeschütztes Anmelden von Benutzern.
2. *Benutzerkonten verwalten*: Ermöglicht das Anlegen und Verwalten von Benutzerkonten
3. *Projekte anlegen und verwalten*: Ermöglichen das Erstellen und Verwalten von Projekten mit zugehörigen Informationen wie Beschreibung, Projektkostenstelle, Verantwortlicher und Arbeitsstunden.
4. *Arbeitsstunden erfassen*: Ermöglicht das Erfassen von Arbeitsstunden für einzelne Projekte und Benutzer.
5. *Daten exportieren*: Ermöglicht das Exportieren von Projekten aus dem System in gängigen Dateiformaten wie CSV oder Excel.
6. *Datenbank*: Ermöglicht das Verbinden mit einer MySQL Datenbank und das Erstellen von Backups
7. *Datenbank importieren*: Ermöglicht das Importieren eines Backups, um die Datenbank wiederherzustellen.

## 2.3 Wesentliche Qualitätsanforderungen und Rahmenbedingungen

1. *Technologie*: Die Anwendung wird in Java entwickelt, um plattformunabhängige Portabilität zu gewährleisten. Die Datenbank verwendet MySQL, um eine zuverlässige und skalierbare Lösung für die Datenspeicherung zu bieten.
2. *Hardware*: Das System sollte auf gängigen Desktop-Computern und Laptops mit ausreichender Leistung für die reibungslose Ausführung von Java-Anwendungen laufen können. Es sind keine speziellen Hardware-Anforderungen erforderlich.
3. *Nicht-funktionale Anforderungen*: Das System sollte benutzerfreundlich, effizient und skalierbar sein. Es sollte in der Lage sein, eine große Anzahl von Projekten und Benutzern zu unterstützen, ohne die Systemleistung oder -stabilität zu beeinträchtigen.

Diese Qualitätsanforderungen bieten einen Überblick über die Erwartungen an das „Project Tracker“ System und dienen als Grundlage für die detaillierte Spezifikation und Planung der Anwendung.

## 3 Detaillierte Spezifikation

### 3.1 Systemabgrenzung, Systemarchitektur und Datenhaltung

Das zu entwickelnde System ist eine Webanwendung, die an das Model-View-Controller (MVC) Prinzip angelehnt entwickelt wird.

Die Architektur besteht aus den folgenden Paketen:

1. *Model*: Beinhaltet die Datenobjekte des Systems.
2. *View*: Beinhaltet die Benutzeroberfläche und die Präsentation der Daten.
3. *Control*: Verantwortlich für die Verarbeitung von Benutzereingaben, die Interaktion zwischen dem Model und der View.
4. *DAO*: Enthält die Datenzugriffsobjekte, die für den Zugriff auf die Datenbank verantwortlich sind.
5. *Util*: Beinhaltet Hilfsklassen und -methoden, die an verschiedenen Stellen im System verwendet werden.
6. *Database*: Enthält Verbindungsmanagement, Initialisierung, Konfiguration und Verbindung zur Datenbank.

Die Datenhaltung erfolgt mittels JDBC (Java Database Connectivity), das für die Kommunikation mit der Datenbank zuständig ist. Die Datenbankstruktur und das Datenbankschema werden zu einem späteren Zeitpunkt detaillierter definiert, nach Abschluss der Planungsphase.

Version	Autor	Dateiname	Letzte Änderung	Druckdatum	Seite
1.0	Schalk	SRS-Project-Tracker.pdf	10.06.2023	12.06.2023	4 von 8

## 3.2 Akteure des Systems (Personas)

Die Akteure, die mit dem System interagieren, können unterteilt werden in:

1. *Benutzer*: Personen, die das System verwenden, um bestimmte Aufgaben durchzuführen.
2. *Admins*: Personen mit erhöhten Berechtigungen, die zusätzlich in der Lage sind, das System und seine Benutzer zu verwalten.

## 3.3 Detaillierte funktionale Anforderungen (Szenarios & Screens)

### 3.3.1 Datenbank Login

#### 3.3.1.1 Business Use Case / Wirkungsweise

Benutzer können sich mit einer MySQL-Datenbank verbinden.

#### 3.3.1.2 Spezielle Abhängigkeiten / nicht funktionale Anforderungen

Falls es die Datenbank das Schema nicht beinhaltet, wird die Datenbank automatisch initialisiert. Bei erfolgreichem Login mit der Datenbank werden die Anmeldeinformationen für folgende Verbindungen zu Datenbank gespeichert und nur abgefragt, falls sich die Anwendung sich nicht mit der Datenbank verbinden kann.

### 3.3.2 User Login

#### 3.3.2.1

Benutzer können ihre Identität überprüfen und sich ins System einloggen, indem sie ihren Benutzernamen und ihr Passwort eingeben.

#### 3.3.2.2

Die Login-Funktion hängt von der Verfügbarkeit der Datenbank ab, da die Benutzerinformationen dort gespeichert sind. Die Funktion muss sicher sein und sollte Passwörter nicht im Klartext speichern. Nach dem ersten erfolgreichen Login an einem Arbeitstag wird außerdem ein automatisch ein Backup erstellt. Dieses Backup wird für mindestens zwei Wochen aufbewahrt und zu einem späteren Zeitpunkt automatisch entfernt. Datenbank Backups werden lokal gespeichert.

Sollte sich ein Benutzer zum ersten Mal anmelden, so muss er ein persönliches Passwort erstellen. (siehe 3.3.7.2)

### 3.3.3 Projekte verwalten

#### 3.3.3.1

Benutzer können Kostenstellen, Verantwortliche in die Datenbank hinzufügen, verändern und entfernen. Benutzer können nur für sich sichtbare neue Projekte in der Datenbank hinzufügen und entfernen.

#### 3.3.3.2

Benutzer sehen auf alle verfügbaren Kostenstellen und Verantwortliche in der Datenbank, diese lassen sich aber nicht löschen, falls sie bei einem Projekt verwendet werden. Projekte müssen folgendes beinhalten: Beschreibung, Kostenstelle und Verantwortliche Person.

### 3.3.4 Zeit stoppen

#### 3.3.4.1

Benutzer können für ein existierendes Projekt eine Stoppuhr starten und stoppen, die zeitliche Differenz wird in der Datenbank hinterlegt.

Version	Autor	Dateiname	Letzte Änderung	Druckdatum	Seite
1.0	Schalk	SRS-Project-Tracker.pdf	10.06.2023	12.06.2023	5 von 8

### 3.3.4.2

Es ist möglich, Zeiteinträge zu löschen und die verstrichene Zeit zu editieren. Es ist nicht erlaubt, das Datum zu editieren.

## 3.3.5 CSV-Export

### 3.3.5.1

Benutzer können für einen ausgewählten Zeitraum eine CSV-Datei exportieren, welche die summierte Arbeitszeit pro Projekt und pro Tag beinhaltet.

Hier ist die Form des CSV-Exports:

```
Arbeitstag;Stunden;Beschreibung;Kontierung;Ansprechpartner  
30.05.2023;1,45;Projekt Naboo;955;Frau Amidala  
31.05.2023;15,25;Projekt Naboo;955;Frau Amidala
```

### 3.3.5.2

Die erste Zeile beinhaltet die Beschreibung, wie oben gezeigt. Als Trennzeichen wird das Satzzeichen „Semikolon“ verwendet. Die Ausgabe erfolge in Stunden und wird auf zwei Kommastellen genau angegeben.

## 3.3.6 Persönliches Passwort ändern

### 3.3.6.1

Benutzer können ihr eigenes Passwort selbständig ändern

### 3.3.6.2

Das Passwort muss 8 Zeichen lang sein und mindestens ein Sonderzeichen, einen Klein und Großbuchstaben sowie eine Zahl beinhalten.

## 3.3.7 Benutzerverwaltung (betrifft Administrator)

### 3.3.7.1

Administratoren können das Passwort anderer Benutzer anlegen, löschen, vorübergehend deaktivieren und die Rolle „user“ oder „admin“ für einen User vergeben.

### 3.3.7.2

Der Benutzername darf kein Semikolon beinhalten. Der Benutzer admin kann nicht gelöscht werden. Das Passwort wird automatisch generiert und besteht aus Username und einer vierstelligen zufällig generierten Zahl in der Datenbank „verschlüsselt“ gespeichert.

## 3.3.8 Datenbank wiederherstellen (betrifft Administrator)

### 3.3.8.1

Administratoren können die Datenbank wiederherstellen.

### 3.3.8.2

Backups werden lokal im Ordner *C:\MySQL-Backups* gespeichert.

## 3.4 Schnittstellen

### 3.4.1 Benutzerschnittstellen (GUI)

Die Benutzeroberfläche besteht aus JavaFX-Komponenten.

Version	Autor	Dateiname	Letzte Änderung	Druckdatum	Seite
1.0	Schalk	SRS-Project-Tracker.pdf	10.06.2023	12.06.2023	6 von 8

### 3.4.2 Systemschnittstellen

Im Folgenden werden die wichtigsten Systemschnittstellen erläutert. Diese Anwendung interagiert mit einer lokalen SQL-Datenbank.

#### 3.4.2.1 Lokale SQL-Datenbank

1. Syntax/Semantik der Daten der Schnittstelle: MySQL
2. Datenformat / Übertragungsprotokoll: JDBC
3. Timing / Datenrate / Verschlüsselung: Timing und Datenrate, abhängig von der lokalen Hardware.  
Keine lokale Verschlüsselung der Verbindung.

### 3.5 Nicht-Funktionale Anforderungen

#### 3.5.1 Vorgaben zur Hardware und Software

Hardware: Das System soll auf einem Computer mit mindestens einem Quad-Core-Prozessor und 8 GB RAM laufen. Software: Das System soll auf Java 20 oder höher laufen.

#### 3.5.2 Performance

Die Anwendung soll in der Lage sein, bis zu 500 Benutzer gleichzeitig zu unterstützen. Die durchschnittliche Antwortzeit für Datenbankabfragen sollte unter 200 Millisekunden liegen.

#### 3.5.3 Resources und Anforderungen an die Hardware

Mindestens XX GB freier Festplattenspeicher für die Installation und die Speicherung der Daten.

#### 3.5.4 Security & Safety

Die Anwendung soll einen sicheren Login-Mechanismus haben, der Passwortverschlüsselung verwendet. Damit die Backups erstellt werden können, muss MySQL 8.0 Command Line Client installiert sein.

#### 3.5.5 Reliability

Das System sollte eine Verfügbarkeit von 99,9 % haben, mit geplanten Wartungsfenstern außerhalb der Hauptbetriebszeiten.

#### 3.5.6 Maintenance

Das System sollte so strukturiert sein, dass Wartungsarbeiten und Updates einfach durchgeführt werden können, ohne dass die Hauptfunktionen beeinträchtigt werden.

#### 3.5.7 Portability / Skalierbarkeit / Wiederverwendbarkeit

Die Anwendung sollte so entwickelt werden, dass sie auf verschiedenen Betriebssystemen laufen kann. Der Code sollte modular und gut dokumentiert sein, um die Wiederverwendbarkeit und Wartung zu erleichtern.

#### 3.5.8 Useability

Die Benutzeroberfläche soll intuitiv und einfach zu bedienen sein.

### 3.6 Rahmenbedingungen

1. Das Produkt muss in Java entwickelt werden und mit JavaFX für die Benutzeroberfläche.
2. Das System muss kompatibel mit allen Hauptversionen von Windows und macOS sein.
3. Die Entwicklungsphase muss bis zum 17. Juli abgeschlossen sein.
4. Alle verwendeten Bibliotheken müssen Open-Source sein oder eine Lizenz haben, die ihre Verwendung in kommerziellen Projekten erlaubt.

### 3.7 Lieferumfang

1. Projektdokumentation als .pdf Datei

Version	Autor	Dateiname	Letzte Änderung	Druckdatum	Seite
1.0	Schalk	SRS-Project-Tracker.pdf	10.06.2023	12.06.2023	7 von 8

2. Quellcode als .zip Datei
3. Präsentation des Projektes

Version	Autor	Dateiname	Letzte Änderung	Druckdatum	Seite
1.0	Schalk	SRS-Project-Tracker.pdf	10.06.2023	12.06.2023	8 von 8