Inhalt

[1. Einleitung 2](#_Toc447710453)

[1.1. Projektbegründung 2](#_Toc447710454)

[1.2. Projektziel 2](#_Toc447710455)

[1.3. Projektschnittstellen 2](#_Toc447710456)

[2. Projektplanung 2](#_Toc447710457)

[2.1. Projektzeitplan 2](#_Toc447710458)

[2.2. Modelle 3](#_Toc447710459)

[2.2.1. Spiralmodell 3](#_Toc447710460)

[2.2.2. Test-Driven Development (TDD) 3](#_Toc447710461)

[3. Analysephase 4](#_Toc447710462)

[3.1. Ist-Analyse 4](#_Toc447710463)

[3.2. Lastenheft 4](#_Toc447710464)

[4. Entwurfsphase 4](#_Toc447710465)

[4.1. Zielplattform 4](#_Toc447710466)

[4.2. Architekturdesign 4](#_Toc447710467)

[4.3. Benutzeroberfläche 4](#_Toc447710468)

[4.4. Datenbankentwurf 4](#_Toc447710469)

[4.5. Pflichtenheft 4](#_Toc447710470)

[5. Implementierungsphase 4](#_Toc447710471)

[5.1. Implementierung der Datenstrukturen 4](#_Toc447710472)

[5.2. Implementierung der Benutzeroberfläche 4](#_Toc447710473)

[5.3. Zwischenstand 4](#_Toc447710474)

[6. Dokumentation 4](#_Toc447710475)

[7. Fazit 4](#_Toc447710476)

[7.1. Soll- / Ist- Vergleich 4](#_Toc447710477)

[7.2. Aus dem Projekt Gewonnenes 4](#_Toc447710478)

[7.3. Ausblick 4](#_Toc447710479)

[I. Anhang 4](#_Toc447710480)

[I.I. Detaillierter Zeitplan 4](#_Toc447710481)

# Einleitung

Die folgende Projektdokumentation schildert den Ablauf des Datenbankprojekts, welches wir im Rahmen einer gestellten Aufgabe im Lernfeld 6 durchgeführt haben.

## Projektbegründung

Aufgrund des Verlustes des Überblicks über Komponente für Computer in einzelnen Betrieben, wird durch das ständige nachfragen von Teilen in der Lagerhalle der Arbeitsfluss ständig unterbrochen oder sogar nicht benötigte teile eingekauft werden und somit mehr Geld verbraucht wird. Ein Lösungsbeispiel wäre das Führen einer Excel Liste zum Erfassen der Werte. Hierbei kann es jedoch schnell zu Flüchtigkeitsfehlern kommen, die dann wiederum zu Folgefehlern führen. Außerdem könnte es sein, dass einzelne Eintragungen komplett vergessen werden.

Aufgrund dieser Probleme und der dadurch steigenden Kosten haben wir uns dazu entschieden, die Entwicklung eines Inventar Management Systems (IMS) zu übernehmen, damit die Lagerverwaltung und die Mitarbeiter einen Einfachen Überblick behalten.

## Projektziel

Ziel des Projektes ist es die einzelnen Komponenten zu erfassen und den Mitarbeitern zuzuordnen. Dazu soll eine graphische Benutzeroberfläche mit eigener Datenbank erstellt werden. Die Anwendung soll das Auslesen und Eintragen der Teile für jeden Benutzer möglich machen, sodass alle einen Überblick über die vorhandenen Bestandteile erhalten. Dadurch soll eine Kosten Minimierung gewährleistet sein.

## Projektschnittstellen

# Projektplanung

## Projektzeitplan

|  |  |
| --- | --- |
| Projektphase | Geplante Zeit |
| Analysephase | 2 Stunden |
| Entwurfsphase | 4 Stunden |
| Implementierungsphase | 13 Stunden |
| Abnahmetest der Gruppe | 1 Stunde |
| Erstellen der Dokumentation | 4 Stunden |
| Gesamt | 24 Stunden |

Eine Detaillierte Zeitplanung findet sich im [Anhang I.I](#_Detaillierter_Zeitplan)

## Modelle

### Spiralmodell

Das Spiralmodell ist ein risikogetriebenes Vorgehensmodell. Da die Risikobetrachtung dadurch ein wesentlicher Aspekt ist gilt ein Projekt als gescheitert, wenn die Beseitigung eines Risikos fehlschlägt. Andererseits gilt ein Projekt als Abgeschlossen wenn es keine Risiken mehr gibt.

Das Spiralmodell ist eine Weiterentwicklung des Wasserfallmodells, in der die Phasen mehrfach Spiralförmig durchlaufen werden.

Beim Projektablauf ist eine Zyklische Wiederholung der Phasen vorgesehen wobei sich bei jedem Zyklus eine Annäherung an die Ziele erkennbar ist, auch wenn sich die Ziele während des Projektfortschritts verändert wurden.

In jedem einzelnen Zyklus werden dann folgende vier Schritte durchlaufen.

* Risikoanalyse; Beschreibung von Rahmenbedingungen, Zielen und Alternativen
* Evaluierung von Alternativen und Reduzieren von Risiken
* Realisieren und Überprüfen des Zwischenproduktes
* Planen des nächsten Zyklus

### Test-Driven Development (TDD)

Bei der testgetriebenen Entwicklung (engl. Test-Driven Development, TDD) werden Tests dazu benutzt, um die Software Entwicklung zu steuern. Hierbei wird folgender Ablauf bei jedem Test eingehalten.

* Ein Test wird geschrieben, der zunächst fehlschlägt.
* Genau so viel Produktivcode wird implementiert, dass der Test erfolgreich durchläuft.
* Test und Produktivcode werden refaktorisiert.

Da erfolgreiche durchgelaufene Tests mit Grün und nicht erfolgreiche Tests Rot gekennzeichnet werden, spricht man auch vom ‘‘ Red - Green - Refactor ‘‘ - Zyklus.

# Analysephase

## Ist-Analyse

|  |  |  |
| --- | --- | --- |
| Für welche Arbeitsabläufe wird das Inventar Management System benötigt? |  |  |
| Wie wird das Management bisher erledigt? |  |  |
| Welche Probleme sind bei dem bisherigen Vorgehen Aufgetreten? |  |  |
| Wie viele Personen nutzen das Programm? |  |  |
| Wie hoch ist der Gebrauch des bisherigen Management Programms? |  |  |

## Lastenheft

Das Lastenheft wurde am Ende der Analysephase erstellt. Dieses enthält die Kriterien, die die Entwickler aus Sicht der Anwender formuliert haben. Die Anforderungen wurden der Priorität absteigend nach Muss-, Soll- und Kann-Kriterien gestaffelt. Ein Auszug des Lastenhefts findet sich im Anhang unter [A.2: Lastenheft](#_Lastenheft).

## Pflichtenheft

# Entwurfsphase

## Zielplattform

## Architekturdesign

## Benutzeroberfläche

## Datenbankentwurf

## Pflichtenheft

Das Pflichtenheft wurde basierend auf den Anforderungen im Lastenheft mit Hilfe der in der Entwurfsphase getroffenen Entscheidungen erstellt. Es enthält die konkrete Umsetzung der in   
[3.2 Lastenheft](#_Lastenheft_1) definierten Anforderungen. Ein Auszug des Pflichtenhefts ist im Anhang unter [A.3 Pflichtenheft](#_Pflichtenheft) zu finden.

# Implementierungsphase

## Implementierung der Datenstrukturen

## Implementierung der Benutzeroberfläche

## Zwischenstand

# Dokumentation

# Fazit

## Soll- / Ist- Vergleich

## Aus dem Projekt Gewonnenes

## Ausblick

# Anhang

## Detaillierter Zeitplan

|  |  |  |  |
| --- | --- | --- | --- |
| Phasen | Stunden | | |
| Analysephase |  |  | 2 |
| 1. Ist - Analyse |  | 1 |  |
| 1. Erstellen der Muss-, Kann- und Wunsch Kriterien |  | 1 |  |
| Entwurfsphase |  |  | 4 |
| 1. Datenbank Entwurf |  | 2 |  |
| * 1. Festlegen der Entitäten | 1 |  |  |
| * 1. Datenbank – Modellierung | 1 |  |  |
| 1. Benutzeroberfläche entwerfen und abstimmen |  | 1 |  |
| 1. Erstellen des Pflichtenhefts |  | 1 |  |
| Implementierungsphase |  |  | 13 |
| 1. Anlegen der Datenbank |  | 2 |  |
| 1. Unit-Tests Schreiben |  | 2 |  |
| 1. Programmierung des C# Codes |  | 7 |  |
| 1. Umsetzung der Oberfläche |  | 2 |  |
| Abnahmetest der Gruppe |  |  | 1 |
| 1. Abnahmetest der Gruppe |  | 1 |  |
| Erstellen der Dokumentation |  |  | 4 |
| 1. Erstellen der Projektdokumentation |  | 4 |  |
| Gesamt |  |  | 24 |

## Lastenheft‘

**Muss-Kriterien:**

* Dauerhaftes Speichern von Komponenten mit allen definierten Merkmalen (Arbeitsspeicher, Festplatte, Grafikkarte, Hauptplatine, Monitor, Prozessor)
* Anzeigen von Komponenten in einer Liste
* Bearbeiten von bereits gespeicherten Komponenten
* Löschen von gespeicherten Komponenten
* Grafische Oberfläche

**Soll-Kriterien:**

* Filtern von Komponenten nach definierten Merkmalen
* Umrechnung in andere Einheiten (z.B. Byte -> GB)
* Löschen von mehreren Komponenten gleichzeitig

**Kann-Kriterien:**

* Ansprechende grafische Oberfläche
* Konfigurieren der Datenbankverbindung über die Oberfläche
* Exportieren einer gefilterten Tabelle als CSV-Datei

## Pflichtenheft

**Muss-Kriterien:**

* Speichern vom Komponenten in einer relationalen Datenbank (MariaDB)
* Datenbankverbindung mittels MySQL-Treiber für .NET (C#)
* Selektieren von Datensätzen aus der Datenbank mittels SQL
* Anzeigen von Datensätzen in einer Tabelle auf der GUI
* Löschen von gespeicherten Komponenten aus der Datenbank mittels SQL
* Gestalten der grafischen Oberfläche (GUI) mit dem Windows Presentation Foundation Framework (WPF)

**Soll-Kriterien:**

* Selektieren von gefilterten Listen aus der relationalen Datenbank mittels SQL
* Schreiben eines Algorithmus für die Umrechnung von Einheiten in C#
* Selektieren von mehreren Einträgen mittels Auswahlboxen auf der WPF-Oberfläche

**Kann-Kriterien:**

* Entwicklung einer ergonomischen GUI mittels Mockups
* Konfigurieren der Datenbankverbindung über ein Konfigurationsfenster in C# mittels WPF
* Exportieren einer gefilterten Tabelle als CSV-Datei mittels File Stream