

# **Java Programm - Bücherverwaltung**

Marco Bräuer

04.10.2021

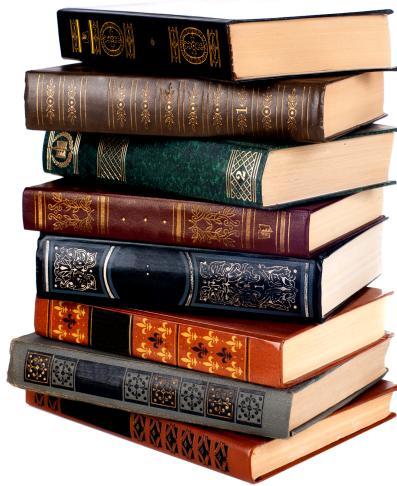


Abbildung 1: Logo Büchersammlung

# **Inhaltsverzeichnis**

<b>1</b>	<b>Analyse</b>	<b>3</b>
1.1	Anforderung an das Programm . . . . .	3
1.1.1	Eigenschaften eines Buches . . . . .	3
1.2	Anwendungsfälle . . . . .	3
1.3	Projektbegründung . . . . .	4
1.4	Programmschnittstellen . . . . .	4
1.5	Projektphasen . . . . .	4
1.6	Entwicklungsprozess . . . . .	4
<b>2</b>	<b>Entwurf</b>	<b>5</b>
2.1	Zielplattform . . . . .	5
2.2	Architekturdesign . . . . .	5
2.3	Entwurf der Benutzeroberflächen . . . . .	5
2.4	Datenmodell . . . . .	6
<b>3</b>	<b>Implementierung</b>	<b>8</b>
3.1	Datenstruktur . . . . .	8
3.2	Logik . . . . .	8
3.3	Benutzeroberfläche . . . . .	8
3.4	Test . . . . .	8
<b>4</b>	<b>Dokumentation</b>	<b>9</b>
4.1	Benutzerdokumentation . . . . .	9
4.1.1	Art der Dokumentation . . . . .	9
4.1.2	Beispiel . . . . .	9
4.2	Entwicklerdokumentation . . . . .	9
4.2.1	Art der Dokumentation . . . . .	9
4.2.2	Beispiel . . . . .	9
<b>5</b>	<b>Fazit</b>	<b>10</b>
5.1	Soll/Ist-Vergleich . . . . .	10
5.2	Lesson Learned . . . . .	10

# **1 Analyse**

## **1.1 Anforderung an das Programm**

Es wird ein neues Programm von Beginn an entwickelt.

Das Programm soll eine Bücherliste tabellarisch darstellen. Jedes Buch soll mehrere Eigenschaften besitzen und diese sollen mit dargestellt werden.

Dem Programm soll eine Datenbank zugrunde liegen aus der zu Programmstart geladen wird.

Über eine Maske sollen neue Bücher hinzugefügt werden können und ausgewählte Bücher müssen gelöscht werden können. Die einzelnen Eigenschaften der Felder müssen geändert werden können.

Über ein Eingabefeld soll eine Suche nach der ISBN Nummer möglich sein.

Gefundene Ergebnisse sollen angezeigt werden und bei Bedarf der Bücherliste hinzugefügt werden können.

### **1.1.1 Eigenschaften eines Buches**

- Nummer in der Liste
- Name des Buches
- Name des Autors
- Jahr der Erstauflage
- Ort/Platz im Regal
- ISBN Nummer

## **1.2 Anwendungsfälle**

Es soll für einen Einzelplatz für einen Anwender mit einer privaten Büchersammlung entwickelt werden.

### **1.3 Projektbegründung**

Bei einer Büchersammlung ist es Sinnvoll einen geordneten Überblick zu haben. Vor allem um Bücher schnell zu finden sowohl für Bücher im Bestand als auch für neue Bücher, die hinzugefügt werden sollen.

### **1.4 Programmschnittstellen**

Die Datenbank als Speicher ist die erste Schnittstelle.

Eine geeignete online API zu einer Bücherdatenbank soll für die ISBN-Suche verwendet werden.

### **1.5 Projektphasen**

<b>Projektphase</b>	<b>geplante Zeit</b>
Analyse	8h
Entwurf	8h
Implementierung	16h
Dokumentation	8h
<b>Gesamt</b>	<b>40h</b>

Tabelle 1: Projektphasen

### **1.6 Entwicklungsprozess**

Bei der Bearbeitung des Projekts wird das Wasserfallmodell verfolgt. Vom Problem zum Programm in folgenden Schritten:

- Analyse
- Entwurf
- Implementierung
- Test

## 2 Entwurf

### 2.1 Zielplattform

Die Anwendung soll auf Windows PC lauffähig sein.

Windows 10 wird als Standard System vorausgesetzt.

Als Programmiersprache wird Java 8 verwendet und mit der IDE Eclipse wird der Quellcode geschrieben.

Als Datenbank wird eine lokale SQLite Datenbank eingesetzt.

### 2.2 Architekturdesign

Ich teile die Entwicklung des Programms in 3 Ebenen.

- Datenbank
- Logik
- Oberfläche

Die Einteilung dient der Übersicht in der Erstellung des Quellcodes und für eine bessere Dynamik.

So kann z.B. durch die Abgrenzung der Datenbankebene von der Logikebene schnell ein anderes Datenbanksystem als Grundlage gewechselt werden.

Zum Beispiel von **SQLite** zu **MySql Server**.

Die Trennung von Logik zu Oberfläche bietet den Vorteil, dass spätere Anpassungen in der Bedienung oder ein komplett neues Design zügig geändert werden kann ohne die Logik umschreiben zu müssen. Außerdem könnten die Ebenen auch separat in anderen Projekten wiederverwendet werden.

### 2.3 Entwurf der Benutzeroberflächen

Ein GUI ist geeignet, um schnell und bedienerfreundlich die Anforderungen an das Programm zu erfüllen.

Zum Erstellen der Benutzeroberfläche verwende ich den Window Builder, der als Addon in der Eclipse IDE kostenlos dazu geladen werden kann.

Die Informationen über die einzelnen Bücher werden, in einem Grid gelistet, dargestellt.

Bücherverwaltung 1.0							X							
<a href="#">löschen</a> <a href="#">bearbeiten</a> <a href="#">hinzufügen</a>	Liste durchsuchen ...													
	Nummer	Titel	Autor	Jahr	Genre	Verlag	ISBN							
	1	Testtitel1	AutorXY	1988	Horror	V1	564684848							
	2	Testtitel2	AutorKP	1972	Roman	V2	669684968							
	3	Testtitel3	AutoPZ	2005	Biografie	V3	987968749							
	...													
	99	Testtitel98	AutorKP	2011	Novelle	V5	846813584							
	100	Testtitel100	AutoPZ	2019	Biografie	V1	541554856							
	Im Internet suchen ...													
Neues Buch erstellen														
Nummer														
1 Gefunden 1 AutorXY														
Nummer														
1 NEU NEUNEU														
Nummer														
1 NEU NEUNEU														

Abbildung 2: Entwurf der Benutzeroberfläche

## 2.4 Datenmodell

Zu Beginn des Programms wird eine Verbindung zur Datenbank hergestellt. Danach werden Listen erzeugt und diese mit den Daten aus der Datenbank gefüllt. Während der Laufzeit werden alle Darstellungen und interne Suchen über die Listen vollzogen. Änderungen, wie das bearbeiten einzelner Felder, das Löschen oder das Hinzufügen werden nach aktualisieren der Listen in die Datenbank geschrieben. Zum Programm Ende wird die Verbindung zur Datenbank getrennt. In der folgenden Abbildung wird ein Modell des Programms dargestellt.

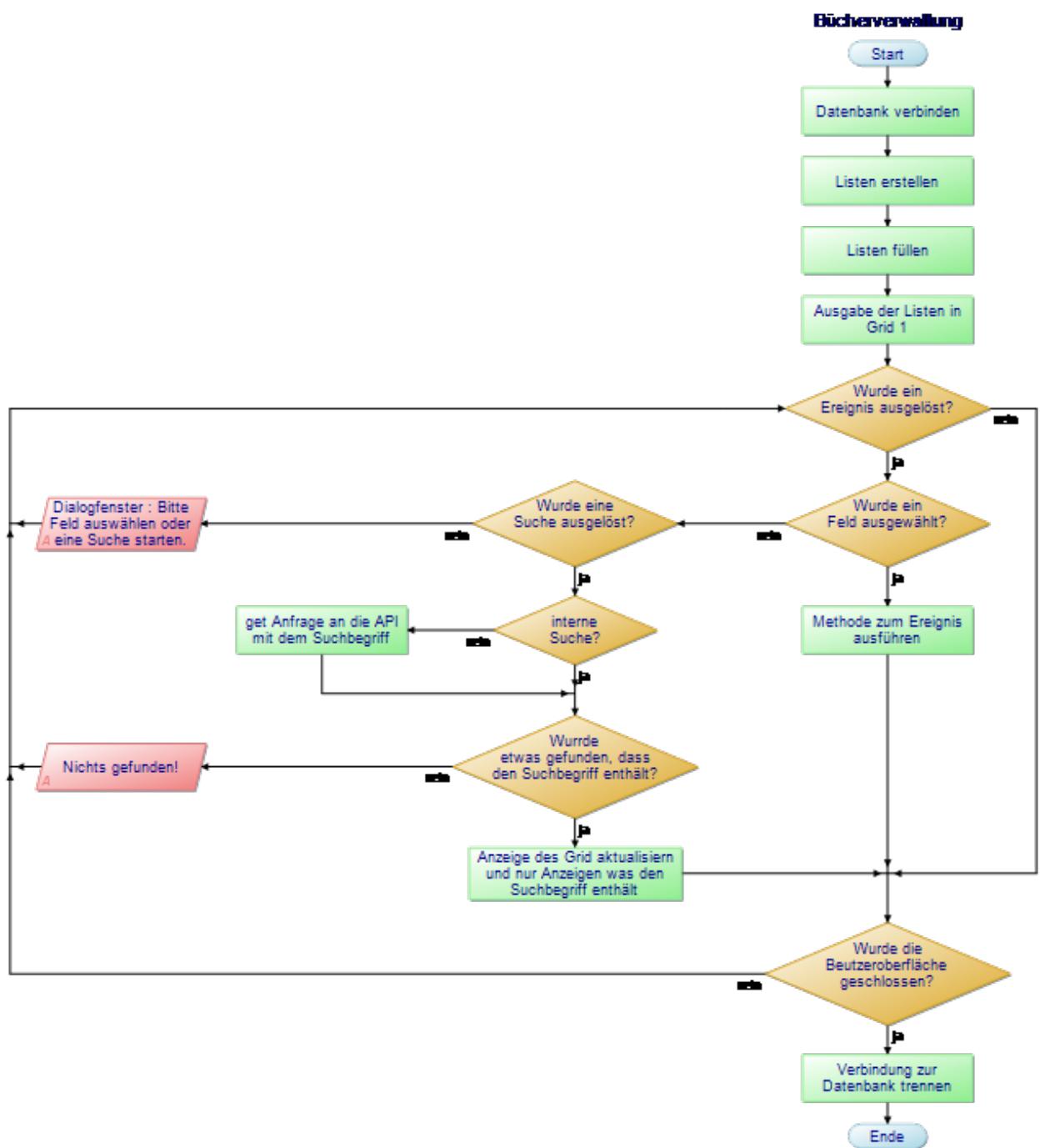


Abbildung 3: Programmablaufplan

### **3 Implementierung**

#### **3.1 Datenstruktur**

#### **3.2 Logik**

#### **3.3 Benutzeroberfläche**

#### **3.4 Test**

## **4 Dokumentation**

### **4.1 Benutzerdokumentation**

#### **4.1.1 Art der Dokumentation**

#### **4.1.2 Beispiel**

### **4.2 Entwicklerdokumentation**

#### **4.2.1 Art der Dokumentation**

#### **4.2.2 Beispiel**

## **5 Fazit**

### **5.1 Soll/Ist-Vergleich**

### **5.2 Lesson Learned**