

Buchaustellungshelfer



Inhaltsverzeichnis

1 Lastenheft für eine Software zum Abwickeln einer Buchausstellung	3
1.1 Ist – Zustand	3
1.2 Soll – Zustand	3
1.3 Aufgabe	3
2 Mein Buch Weihnachts-Buchausstellung 2019 Gemeinde XYZ	4
3 Benutzerdiagramm für den Administrator	5
4 Benutzerdiagramm für den Erfasser	5
5 Anwendungsfälle	6
5.1 Erfasser	6
5.2 Administrator	7
6 Sequenzdiagramme	8
6.1 Allgemein	8
6.2 Kunde	8
6.3 Kunde	9
6.4 Bestellungen drucken	10
7 Klassen	11
7.1 Klassendiagramm Basisframework	11
7.2 Klassendiagramm Datenframework	12
7.3 Ausstellungshelfer Diagramm	13
7.4 Ausstellungshelfer Model Diagramm	14
7.5 ER – Diagramm	15
8 Klassen	16
8.1 Viewmodel	16
8.1.1 BestellungenViewModel	16
8.1.2 AdministratorViewModel	17
8.1.3 KundenViewModel	18
8.2 Models	19
8.2.1 KundenManager	19
8.2.2 Bestellungen Manager	20
8.2.3 Dokument Manager	21
9 Zusatzinformationen	22
9.1 Mögliche zukünftige Erweiterungen	22

1 Lastenheft für eine Software zum Abwickeln einer Buchausstellung

1.1 Ist – Zustand

- Die Ausstellung findet einmal im Jahr an einem Samstag und Sonntag statt. Durchgeführt wird die Ausstellung von örtlichen Jugendgruppen (Jungschar, Pfadfinder), die entsprechend der Rabattgruppe (Rab.Gr.) umsatzbeteiligt sind.
- Die Bücher mit den skizzierten Buchlisten (siehe Anlage) werden am Donnerstag davor angeliefert.
- Die Bücher werden im Veranstaltungssaal aufgelegt gruppiert nach den Listen.
- Während der Veranstaltung werden die Daten der Besteller in die Bestell-Listen eingetragen.
- Direkt nach der Veranstaltung werden die Bestelllisten ausgewertet und auf den Buchlisten die benötigte Menge je Buch eingetragen.
- Ca. 2 Wochen nach der Veranstaltung werden die Bücher geliefert und von den Kunden abgeholt (oder auch nicht)

1.2 Soll – Zustand

- Für das Erfassen der Besteller-Daten und gewünschten Bücher soll eine Software benutzt werden.
- Direkt nach der Veranstaltung sollen die Buchlisten mit der Anzahl der benötigten Bücher gedruckt werden können.
- Bis zur Lieferung der Bücher soll je Besteller eine „Rechnung“ gedruckt werden können. Zum Papiersparen je A4-Seite ZWEI Rechnungen.
- Entsprechend der Rechnungen sollen die Bücher aufgelegt werden, damit der Abholvorgang beschleunigt werden kann.
- Es kommt vor, dass Bücher nicht geliefert werden. In diesem Fall soll die Bestellung korrigiert, mit einem Verweis versehen und neu gedruckt werden.

1.3 Aufgabe

- Durch Analyse entsprechend der UML (wie im C# Teil 1) das Lastenheft zu einem Pflichtenheft ausarbeiten (Akteure, Anwendungsfälle, Szenarien, Klassendiagramm)
- Datenbankdesign
- Implementierung

Das Pflichtenheft ist (mit Deckblatt, Inhaltsverzeichnis, Kopf- und Fußzeile) in gedruckter Form mit der Software auf einem Datenträger je Gruppe am letzten Prüfungstag abzugeben.

2 Mein Buch Weihnachts-Buchausstellung 2019 Gemeinde XYZ

Gruppe 0 – Kinderbücher

Anzahl	Nr.	Titel	Autor	Verlag	Preis	Rab.Gr.
.....	010	Hänsel und Gretl	Grimm	Childsbook	10.--	1
.....	020	Mein erstes Malbuch	Maler	Color	4.--	1
.....	030	Bimbo, der Elefant	Jäger	Childsbook	8.--	1

Gruppe 1 – Schüler, 6-8 Jahre

Anzahl	Nr.	Titel	Autor	Verlag	Preis	Rab.Gr.
.....	110	1x1 der Mathematik,	Einstein	Schoolprint	10.--	1
.....	111	Lesen lernt lesen	Gutenberg	Gaudete	8.--	1
.....	112	Mal etwas lustiges	Clown u.Brow	Gaudete	8.--	1
.....	113	Schreiben und Lesen,	Gutenberg	Primary	6.--	1

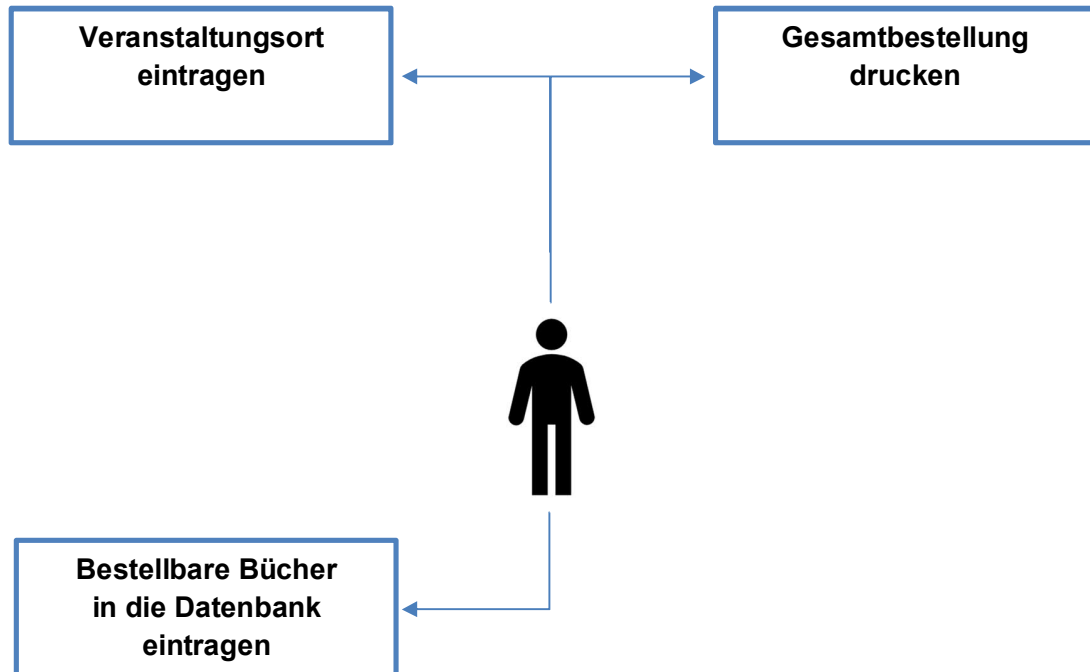
Gruppe 1 – Schüler, 9-10 Jahre

Anzahl	Nr.	Titel	Autor	Verlag	Preis	Rab.Gr.
.....	151	Technik verstehen	Einstein	Schoolprint	10.--	1
.....	152	Sachkunde	Gutenberg	Schoolprint	14.--	1
.....	153	Geschichte einfach	Clown u.Brow	Mybook	8.--	1

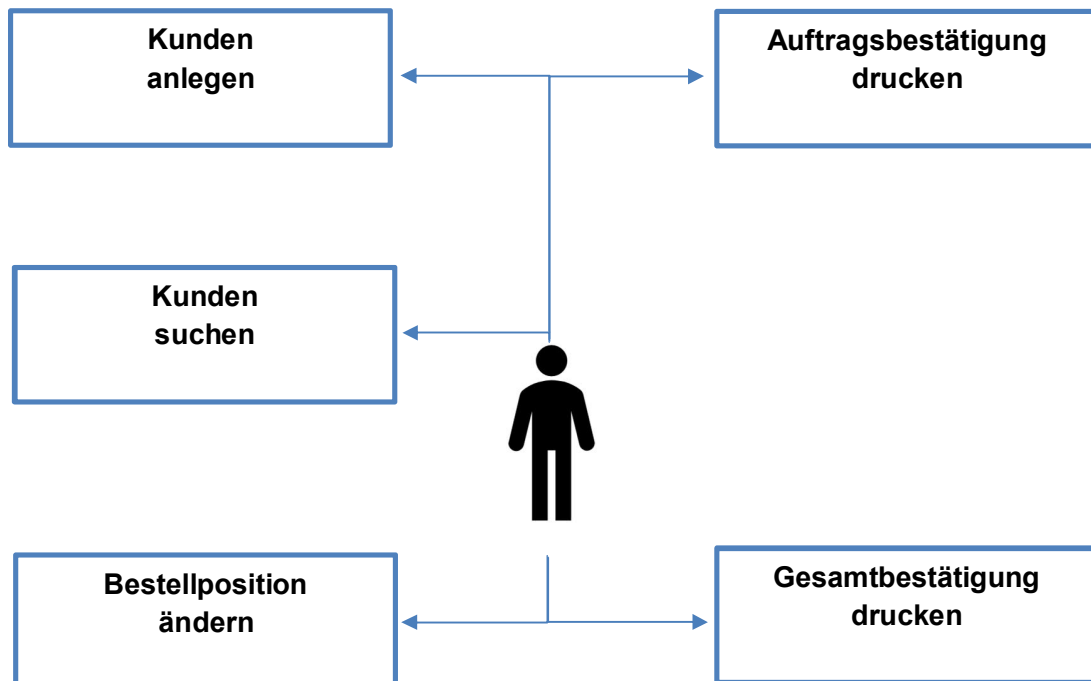
Gruppe 10 – Kalender und sonstiges

Anzahl	Nr.	Titel	Autor	Verlag	Preis	Rab.Gr.
.....	1001	Alle Jahre wieder		Infomedia	10.--	0
.....	1002	Mein Taschenkalender		EveryYear	17.--	0

3 Benutzerdiagramm für den Administrator



4 Benutzerdiagramm für den Erfasser



5 Anwendungsfälle

5.1 Erfasser

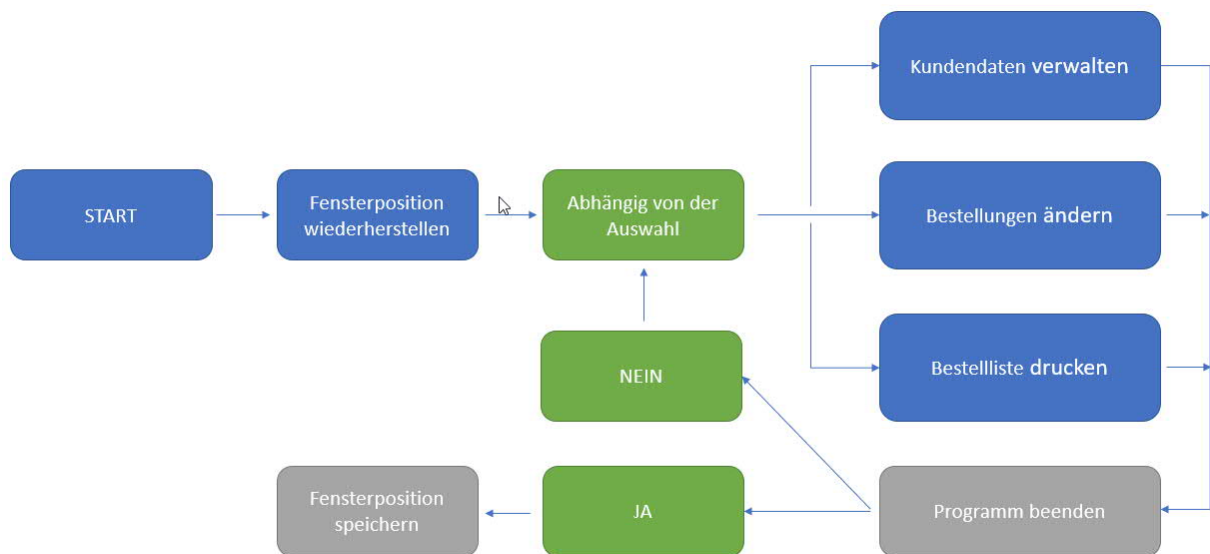
Anwendungsfall	Akteur / Erfasser
Kunde anlegen	<ul style="list-style-type: none"> • Kundendatenfelder befüllen • Kundendaten in der Datenbank speichern
Kunden suchen	<ul style="list-style-type: none"> • Suchen des Kunden mittels dessen Nachnamens. Bei der Eingabe wird die Datenbank durchsucht • Die Namensvorschläge werden in die Combobox gefüllt.
Kunden auswählen	<ul style="list-style-type: none"> • Kunde aus Combobox wählen • Nach der Kundenwahl, werden die Kundendatenfelder automatisch befüllt und eventuell vorhandene Bestellungen angezeigt
Neue Bestellung erstellen	<ul style="list-style-type: none"> • Nachdem ein existenter Kunde ausgewählt wurde, oder ein neuer Kunde angelegt wurde, kann auch eine neue Bestellung erstellt werden.
Der neuen Bestellung ein Buch hinzufügen	<ul style="list-style-type: none"> • Die Buchnummer eingeben • Anzahl eingeben • Button hinzufügen drücken • Nach erfolgreichem Speichern der Bestellung kann eine Bestellbestätigung ausgedruckt werden.
Bestellung ändern	<ul style="list-style-type: none"> • Bestellung im Kundenfenster auswählen • Die zu ändernde Bestellposition auswählen • Die Anzahl der Bücher eingeben • den Button ändern drücken
Gesamtbestellliste drucken	<ul style="list-style-type: none"> • Den Zeitraum der Bestellung auswählen • Das Startdatum muss vor oder gleich dem Enddatum sein • Die bestellten Bücher werden im Listenfeld angezeigt • Button drucken drücken

5.2 Administrator

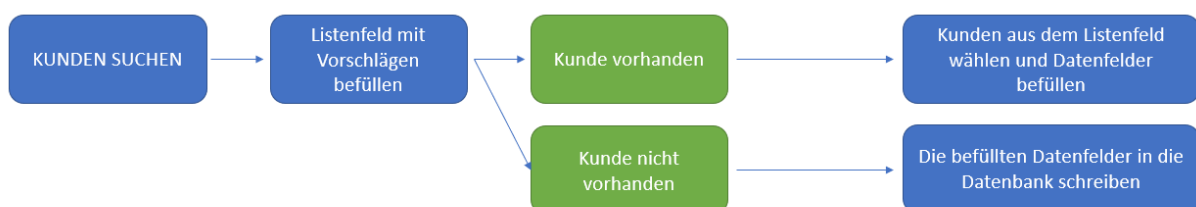
Anwendungsfall	Akteur / Erfasser
Veranstaltungsort eintragen	<ul style="list-style-type: none">• Den Ort der Veranstaltung vor dem Druck der Gesamtbestellung eintragen
Bücher in der Datenbank anlegen	<ul style="list-style-type: none">• Im Bestellfenster das neue Buch eintragen und mit dem Button Buch hinzufügen bestätigen

6 Sequenzdiagramme

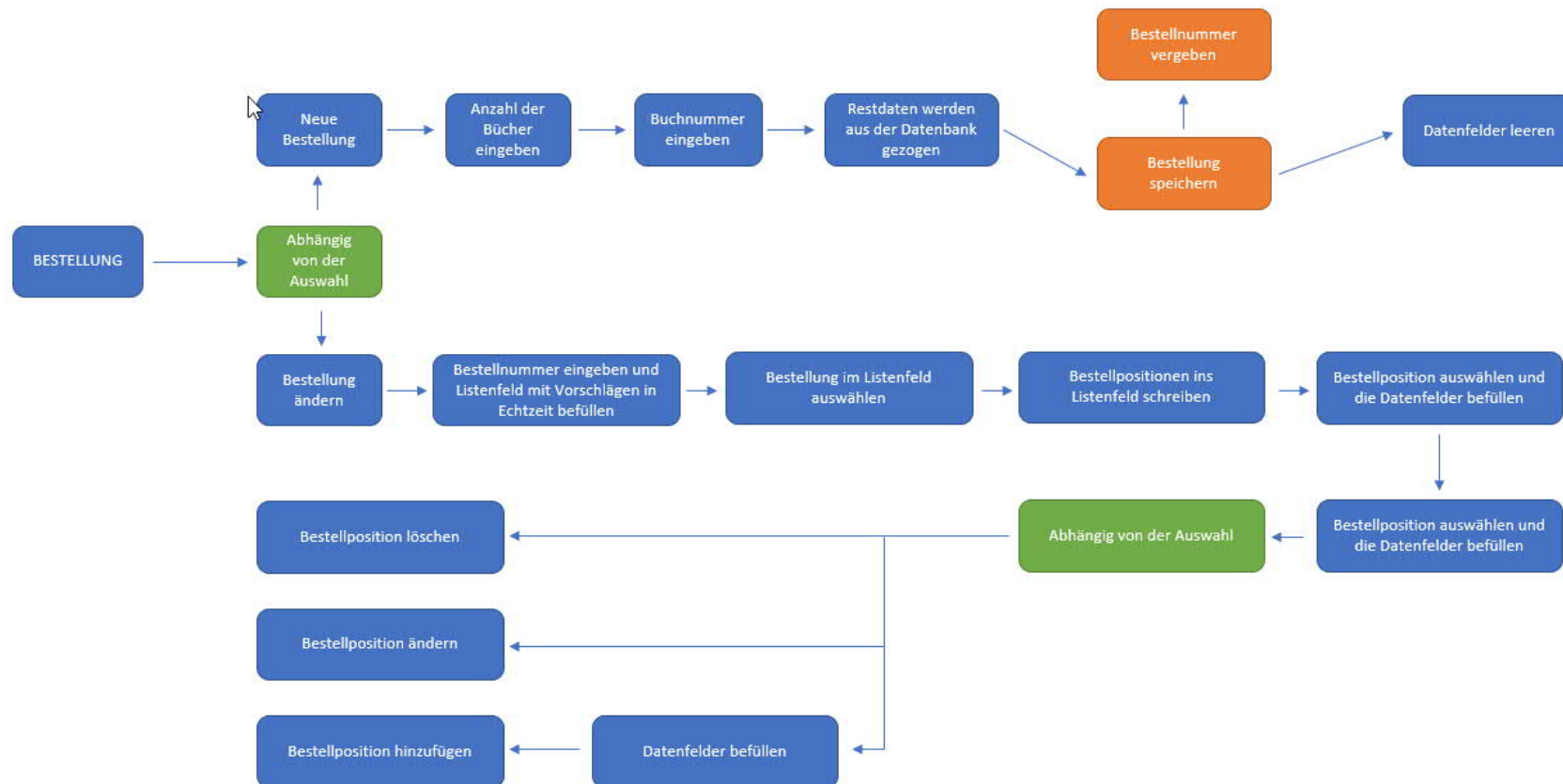
6.1 Allgemein



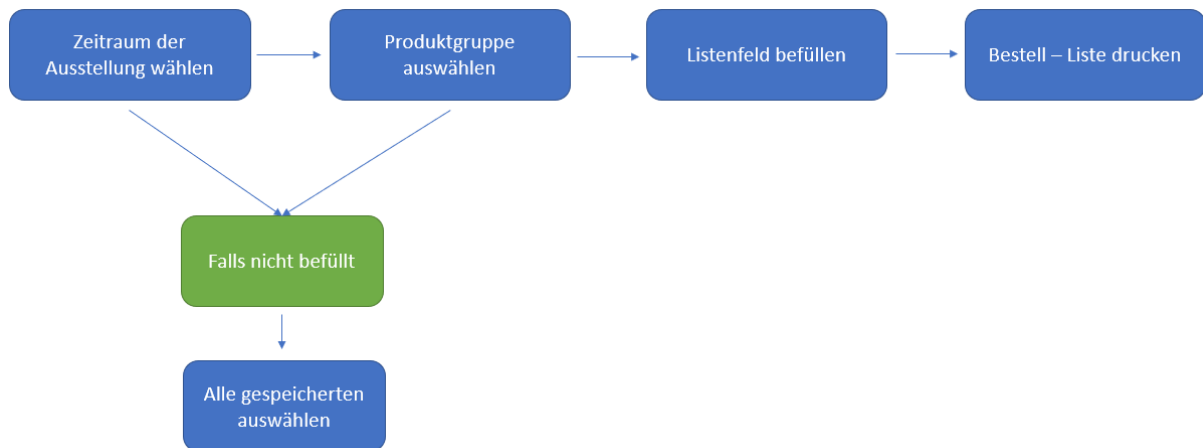
6.2 Kunde



6.3 Kunde

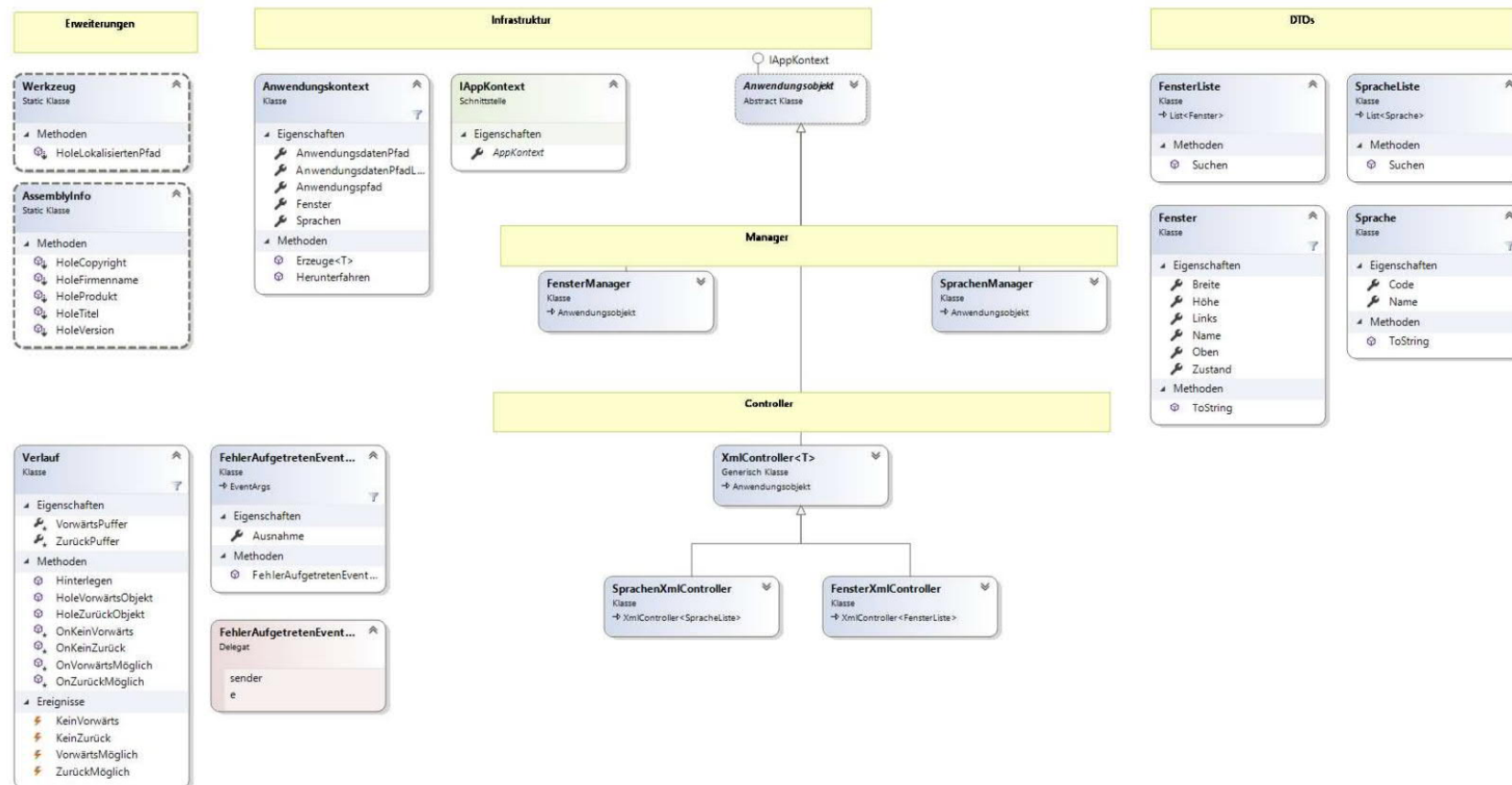


6.4 Bestellungen drucken

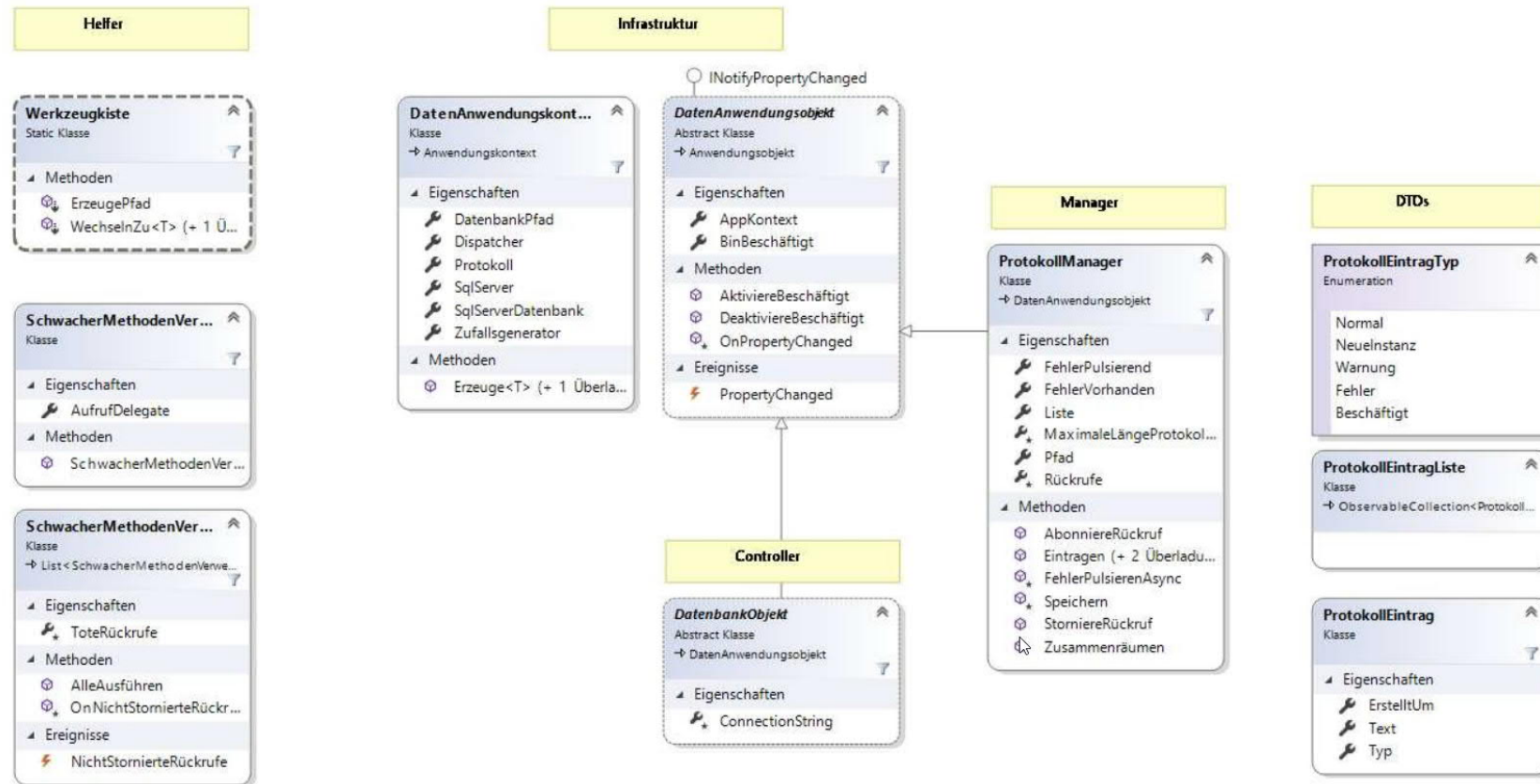


7 Klassen

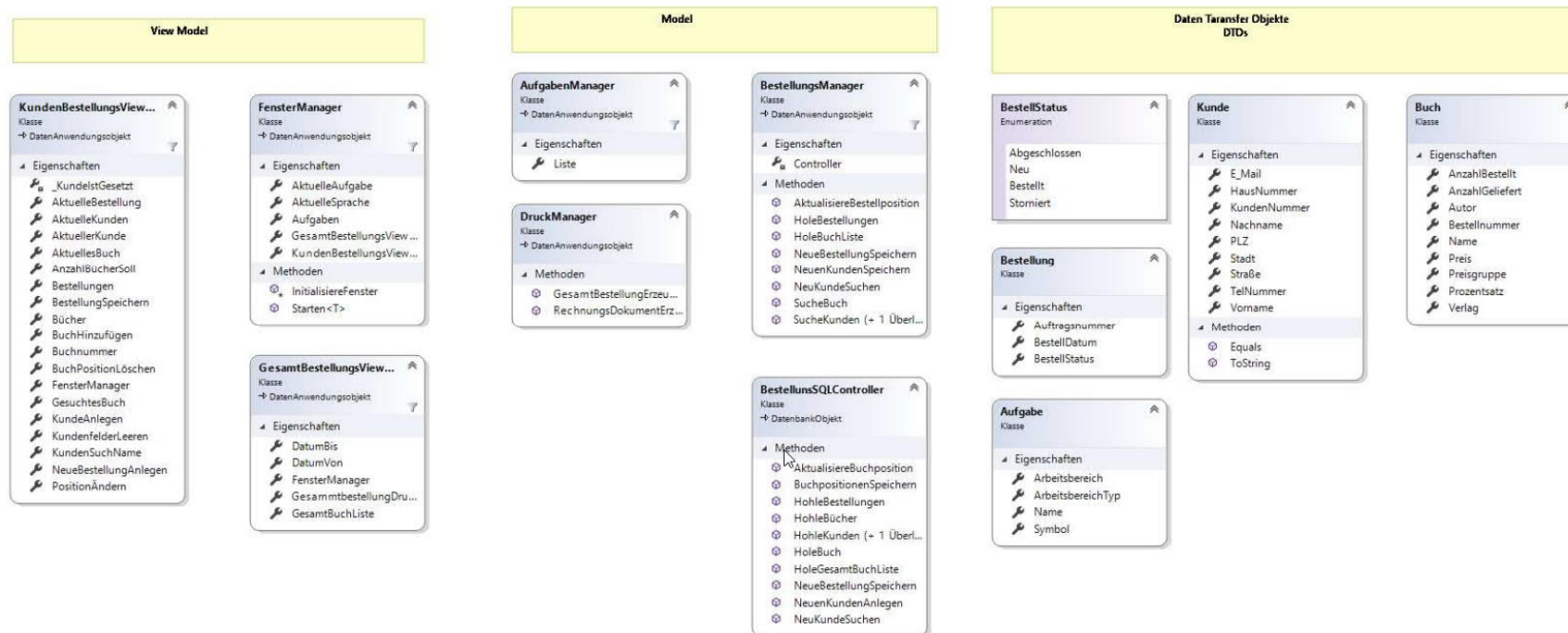
7.1 Klassendiagramm Basisframework



7.2 Klassendiagramm Datenframework



7.3 Ausstellungshelfer Diagramm

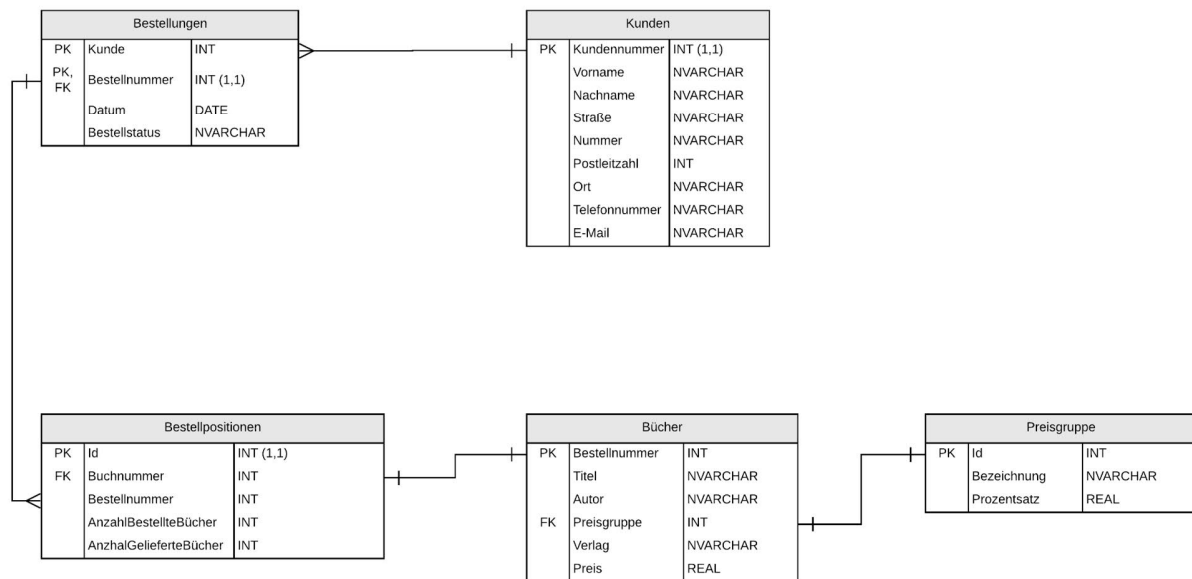


7.4 Ausstellungshelfer Model Diagramm



7.5 ER – Diagramm

Buchausstellungshelfer ER-Diagramm



8 Klassen

8.1 Viewmodel

8.1.1 BestellungenViewModel

Stellt den Dienst zum Verwalten der Bestellungsview zur Verfügung.

▸ Vererbungshierarchie













Namensraum: `NiedSchwa.Ausstellungshelfer.ViewModels`

Assembly: `NiedSchwa.Ausstellungshelfer` (in `NiedSchwa.Ausstellungshelfer.exe`) Version: 1.0.0.0 (1.0.0.0)

▸ Syntax

Der `BestellungenViewModel` Typ exponiert die folgenden Member.

▲ Eigenschaften

	Name	Beschreibung
	<code>AktuelleBestellung</code>	Ruft die Aktuelle Bestellung für den Aktuellen kunden ab oder legt diese fest.
	<code>AktuellesBuch</code>	Ruft das Aktuell Ausgewählte Buch der Aktuellen Buchliste ab oder legt dieses Fest.
	<code>AnzahlBücherSoll</code>	Ruft die Aktuell zu bestellende Anzahl an büchern ab oder legt diese fest.
	<code>Bestellungen</code>	Ruft die für den Aktiven Kunden vorhandenen Bestellungen ab.
	<code>BestellungSpeichern</code>	Ruft den Befehl ab, mit dem eine Bestellung in der Datenbank gespeichert wird.
	<code>Bücher</code>	ruft die Aktuelle Bücher ab
	<code>BuchHinzufügen</code>	Ruft den Befehl ab, mit dem ein Buch der Neuen Bestellung hinzugefügt wird.
	<code>Buchnummer</code>	Legt die zu suchende Buchnummer fest oder gibt dieses zurück.
	<code>BuchPositionLöschen</code>	Ruft den Befehl ab, mit dem die noch unbestellte Buchposition gelöscht werden kann.
	<code>GesuchtesBuch</code>	Ruft das Aktuell Gesuchte Buch der Aktuellen Buchliste ab oder legt dieses Fest.
	<code>NeueBestellungAnlegen</code>	Ruft den Befehl ab, mit dem eine neue Bestellung angelegt werden kann.
	<code>PositionÄndern</code>	Ruft den Befehl ab, mit dem die noch unbestellte Buchposition gelöscht werden kann.

8.1.2 AdministratorViewModel

Stellt Dienste zum Verwalten der Administrator View zur Verfügung.

Vererbungshierarchie

```
System.Object
NiedSchwa.Anwendung.Anwendungsobjekt
NiedSchwa.Anwendung.Daten.DatenAnwendungsobjekt
NiedSchwa.Ausstellungshelfer.ViewModels.BasisViewModel
NiedSchwa.Ausstellungshelfer.ViewModels.AdministratorViewModel
```

Namensraum: NiedSchwa.Ausstellungshelfer.ViewModels

Assembly: NiedSchwa.Ausstellungshelfer (in NiedSchwa.Ausstellungshelfer.exe) Version: 1.0.0.0 (1.0.0.0)

Syntax

```
C#
public class AdministratorViewModel : BasisViewModel
```

Der `AdministratorViewModel` Typ exponiert die folgenden Member.

Eigenschaften

	Name	Beschreibung
	AktuellePreisgruppe	Ruft die Aktuelle Preisgruppe ab oder legt diese fest.
	AlleBücherAbrufen	Ruft den Befehl ab, mit dem alle Vorhandenen Bücher abgerufen werden.
	Buch	Ruft die Daten des Buches ab oder legt diese Fest.
	BuchDatenSindVorhanden	Ruft Den Zustand ob Die Buchdaten Valid sind ab.
	Bücher	Ruft eine Buchliste Ab.
	BuchSpeichern	Ruft den Befehl ab, mit dem Die Buchdaten in die Datenbank geschrieben werden.
	Preisgruppen	Ruft die Aktuell Verfügbaren Preisgruppen ab.

8.1.3 KundenViewModel

Stellt den Dienst zum Verwalten der Kundenview zur Verfügung

Vererbungshierarchie

```
System.Object
  NiedSchwa.Anwendung.Anwendungsobjekt
    NiedSchwa.Anwendung.Daten.DatenAnwendungsobjekt
      NiedSchwa.Ausstellungshelfer.ViewModels.BasisViewModel
        NiedSchwa.Ausstellungshelfer.ViewModels.KundenViewModel
```

Namensraum: NiedSchwa.Ausstellungshelfer.ViewModels

Assembly: NiedSchwa.Ausstellungshelfer (in NiedSchwa.Ausstellungshelfer.exe) Version: 1.0.0.0 (1.0.0.0)

Syntax

```
C#
public class KundenViewModel : BasisViewModel
```

Copy

Der KundenViewModel Typ exponiert die folgenden Member.

Eigenschaften

	Name	Beschreibung
	AktuelleKunden	Ruft die Aktuell verfügbare Kundenliste ab
	AktuellerKunde	Ruft den derzeit aktiven Kunden ab oder legt diesen fest.
	KundeAnlegen	Ruft den Befehl ab, mit dem ein neuer Kunde angelegt werden kann.
	KundelstGesetzt	Prüft ob die Aktuellen Eingaben der Kundenfelder Valid sind.
	KundenfelderLeeren	Ruft den Befehl ab, mit dem ein neuer Kunde angelegt werden kann.
	KundenSuchName	Legt den Aktuellen kunden Suchnamen fest.

8.2 Models

8.2.1 KundenManager

Stellt den Dienst zum Verwalten der Kundendaten zur Verfügung.

Vererbungshierarchie

System.Object
NiedSchwa.Anwendung.Anwendungsobjekt
NiedSchwa.Anwendung.Daten.DatenAnwendungsobjekt
NiedSchwa.Ausstellungshelfer.Models.Manager.BasisDatenManger
NiedSchwa.Ausstellungshelfer.Models.Manager.KundenManager

Namensraum: NiedSchwa.Ausstellungshelfer.Models.Manager

Assembly: NiedSchwa.Ausstellungshelfer (in NiedSchwa.Ausstellungshelfer.exe) Version: 1.0.0.0 (1.0.0.0)

Syntax

C#

```
public class KundenManager : BasisDatenManger
```

Der KundenManager Typ exponiert die folgenden Member.

Methoden

	Name	Beschreibung
	NeuenKundenSpeichern	Speichert einen neuen Kunden in die Datenbank.
	NeuKundeSuchen	Erstellt eine Liste gespeicherten Kunden
	SucheKunden(Int32)	Erstellt eine Liste gespeicherten Kunden
	SucheKunden(String, String)	Erstellt eine Liste gespeicherten Kunden

8.2.2 Bestellungs Manager

Stellt einen Dienst zum Verwalten der Kundenbestellungen zur Verfügung

Vererbungshierarchie

```
System.Object
NiedSchwa.Anwendung.Anwendungsobjekt
NiedSchwa.Anwendung.Daten.DatenAnwendungsobjekt
NiedSchwa.Ausstellungshelfer.Models.Manager.BasisDatenManager
NiedSchwa.Ausstellungshelfer.Models.Manager.BestellungsManager
```

Namensraum: NiedSchwa.Ausstellungshelfer.Models.Manager

Assembly: NiedSchwa.Ausstellungshelfer (in NiedSchwa.Ausstellungshelfer.exe) Version: 1.0.0.0 (1.0.0.0)

Syntax

Der `BestellungsManager` Typ exponiert die folgenden Member.

Methoden

	Name	Beschreibung
	<code>AktualisiereBestellposition</code>	Aktualisiert die Anzahl des Gelieferten Buches einer Bestellung
	<code>BuchSpeichern</code>	Speichert die Buchdaten
	<code>HoleBestellungen</code>	Ruft die Aktuelle Bestellliste ab
	<code>HoleBuchListe</code>	Ruft die Bestellung eines Kunden ab
	<code>HoleGesamtBuchListe</code>	Ruft alle Bestellungen während eines bestimmten Zeitraumes ab
	<code>NeueBestellungSpeichern</code>	Speichert eine Neue Bestellung und speichert diese in der SQL Datenbank ab.
	<code>SucheBuch</code>	Holt ein Buch und gibt dieses zurück
	<code>SucheBücher</code>	Holt eine Liste mit gespeicherten Büchern
	<code>SuchePreisgruppen</code>	Holt eine Liste mit gespeicherten Büchern

8.2.3 Dokument Manager

Stellt Hilfs Methoden zum Erzeugen von Dokumenten bereit

▲ Vererbungshierarchie

System.Object
NiedSchwa.Anwendung.Anwendungsobjekt
NiedSchwa.Anwendung.Daten.DatenAnwendungsobjekt
NiedSchwa.Ausstellungshelfer.Models.Manager.DokumentManager

Namensraum: NiedSchwa.Ausstellungshelfer.Models.Manager

Assembly: NiedSchwa.Ausstellungshelfer (in NiedSchwa.Ausstellungshelfer.exe) Version: 1.0.0.0 (1.0.0.0)

▲ Syntax

```
C#  
public class DokumentManager : DatenAnwendungsobjekt
```

Der DokumentManager Typ exponiert die folgenden Member.

▲ Methoden

	Name	Beschreibung
🔗	GesamtBestellungErzeugen(Bücher, Gesamtbestllung, Size)	Erstellt ein Bestelldokument
🔗	GesamtBestellungErzeugen(Bücher, DateTime, DateTime, Size, String)	Erstellt ein Bestelldokument
🔗	RechnungsDokumentErzeugen	Erstellt ein Rechnungsdokument.

9 Zusatzinformationen

9.1 Mögliche zukünftige Erweiterungen

- Das Administratorfenster soll Passwort geschützt werden
- Auslagern des Models im Namespace:
NiedSchwa.Ausstellungshelfer.Models auf eine eigene Assembly mit dem Namen:
NiedSchwa.Ausstellungshelfer.Daten
- Der Veranstaltungsort soll ebenfalls mit in die Datenbank aufgenommen werden.