

# **Pflichtenheft**

<b>Version</b>	<b>Status</b>	<b>Bearbeitungsdatum</b>	<b>Autoren</b>	<b>Vermerk</b>
V0.1	In Arbeit	15.10.2017	Gruppe 17	
V0.2	In Arbeit	25.10.17	Gruppe 17	
V1.0	Final	29.10.17	Gruppe 17	

# Inhaltsverzeichnis

- 1. Zusammenfassung**
- 2. Aufgabenstellung und Zielsetzung**
- 3. Produktnutzung**
- 4. Interessengruppen (Stakeholders)**
- 5. Systemgrenze und Top-Level-Architektur**
  - 5.1 Kontextdiagramm**
  - 5.2 Top-Level-Architektur**
- 6. Anwendungsfälle**
  - 6.1 Akteure**
  - 6.2 Überblick Anwendungsfalldiagramm**
  - 6.3 Anwendungsfallbeschreibungen**
- 7. Funktionale Anforderungen**
  - 7.1 Muss-Kriterien**
  - 7.2 Kann-Kriterien**
- 8. Nicht-Funktionale Anforderungen**
  - 8.1 Qualitätsziele**
  - 8.2 Konkrete Nicht-funktionale Anforderungen**
- 9. GUI-Prototyp**
  - 9.1 Überblick Dialoglandkarte**
  - 9.2 Dialogbeschreibung**
- 10. Datenmodell**
  - 10.1 Überblick: Klassendiagramm**
  - 10.2 Klassen und Enumerationen**
- 11. Akzeptanzfälle**
- 12. Glossar**

## **13. Offene Punkte**

### **Zusammenfassung**

Es soll eine Terminverwaltungssoftware für einen Artistenverein erstellt werden, in welcher die Artisten einen Überblick über ihre Termine haben und Kunden Shows und Workshops buchen können. Außerdem umfasst die Anwendung einen kleinen Online-Shop, in welchem Kunden Utensilien für angebotene Workshops besorgen können.

### **Aufgabenstellung und Zielsetzung**

Der Artistenverein „Funkenschlag“ besteht aus mehreren Feuerkünstlern und Jongleuren verschiedenster Art. Diese bieten sowohl Workshops als auch Shows für Veranstaltungen an und benötigen hierzu eine Verwaltung ihrer Termine.

Es soll dabei eine Datenbank aufgebaut werden, in der die einzelnen Artisten und Artistengruppen mit ihren Angeboten gespeichert werden können. Aus dieser Datenbank sollen dann über das Internet Shows und Workshops buchbar sein. Wichtig ist, dass wenn ein Artist verbucht ist, dass dann auch Gruppenshows in denen er sonst teilnimmt nicht mehr buchbar sind. Da viele die Shows nur als zusätzliche Einnahmequelle neben ihrer normalen Beschäftigung nutzen, muss die Möglichkeit bestehen Sperrzeiten zu vermerken.

Für die Verwaltung der Showbuchungen, Kunden und Artistendatenbank muss ein einfaches System implementiert werden, auf welches nur den Artisten zugänglich ist. Hierbei sieht jeder Artist nach dem Einloggen nur seine eigenen Kunden, Shows, etc. Dem Vorstand des Vereins soll sich hingegen eine Zusammenfassung über alle Aktivitäten präsentieren.

Workshops können in der vereinseigenen Halle abgehalten werden oder außerhalb. Bei Workshops außerhalb (auch mehrtägig) müssen die Kosten für Anfahrt, Verpflegung und Unterbringung vom Kunden übernommen werden.

Zusätzlich zur Verwaltung der Shows und Workshops soll noch ein kleiner(!) Shop unterhalten werden. Hier werden einige Utensilien zum Jonglieren verkauft. Unter anderem Feuerstäbe, Pois, Jonglierbälle und Diabolos. Workshopteilnehmern soll ein Rabatt auf alle Shopartikel eingeräumt werden.

### **Produktnutzung**

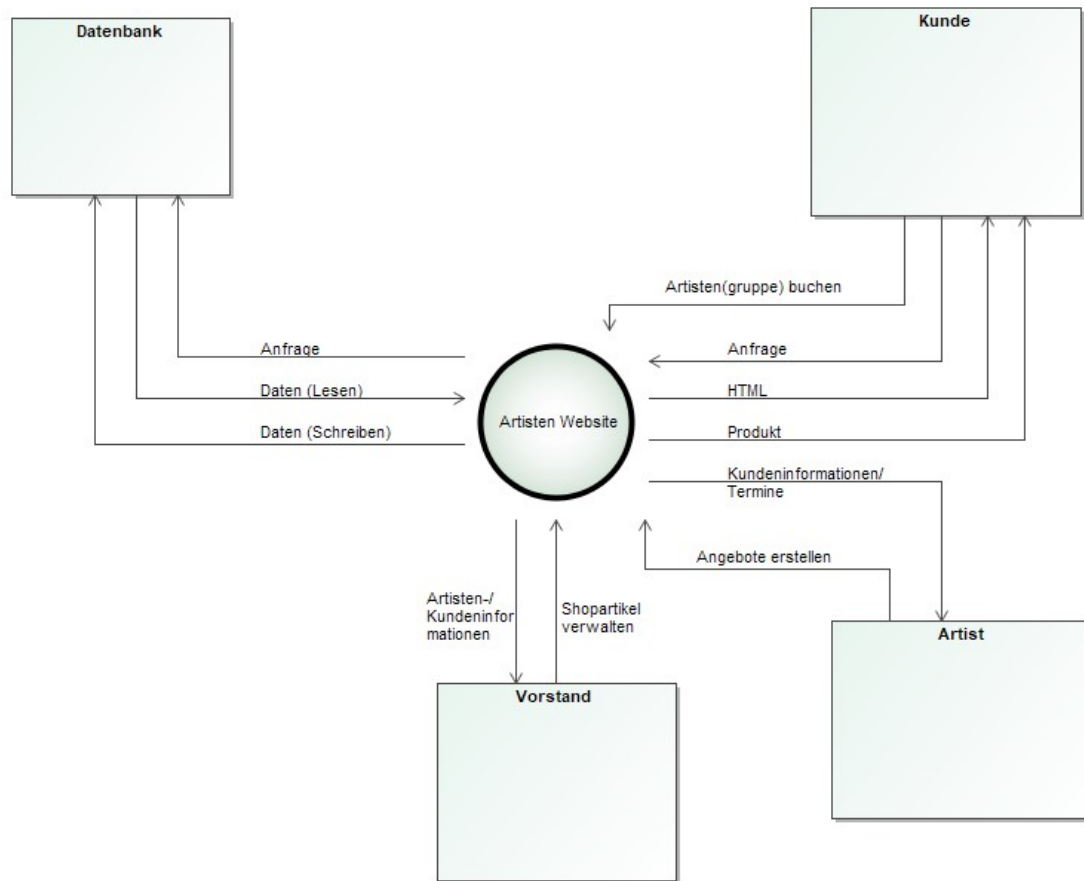
Die Software soll über einen Webbrowser jedem zur Verfügung stehen. Sie soll Kunden zur Bestellung von Shopartikeln und zur Buchung von Workshops und Shows dienen. Auf der anderen Seite soll sie den Artisten einen Überblick über deren Termine verschaffen und ihnen die Möglichkeit geben eigene Shows und Workshops anzubieten. Der Vorstand des Vereins kann administrativ in alle Bereiche der Software eingreifen. Nichtregistrierte Besucher können die Seite nur betrachten, jedoch nichts buchen oder bestellen.

## Interessensgruppen (Stakeholders)

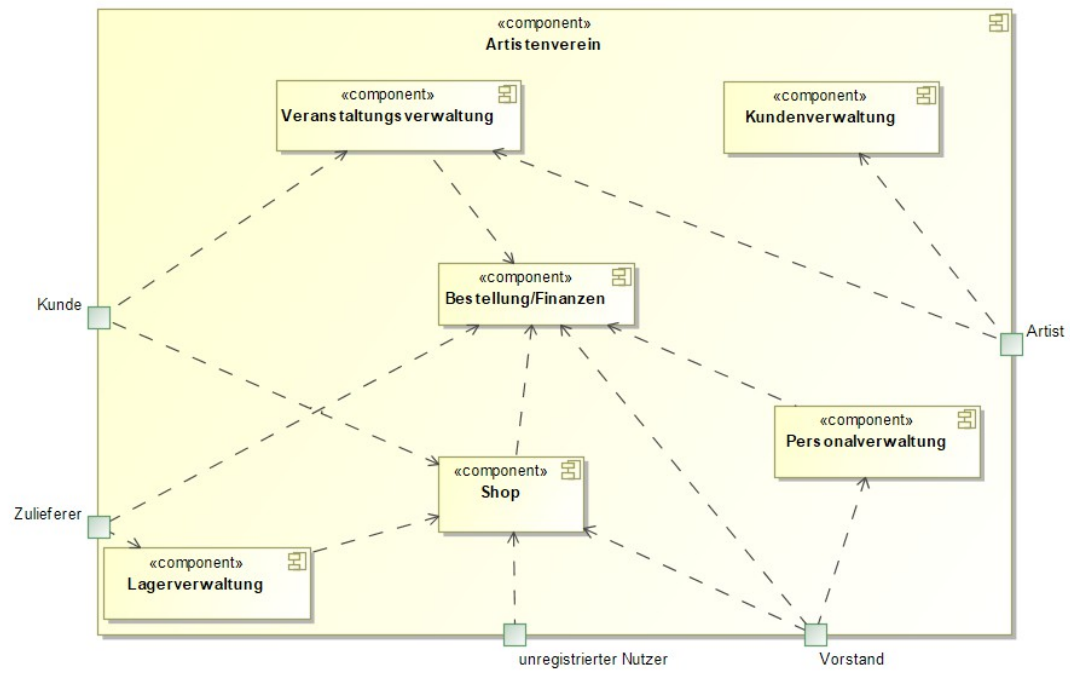
Name	Beschreibung	Ziele
Artistenverein „Funkenschlag“	Der Kunde dieses Projekts	<ul style="list-style-type: none"><li>• Mehr Artikel verkaufen</li><li>• Veranstaltungen anbieten</li><li>• Webpräsenz</li><li>• Gute Nutzererfahrung</li></ul>
Kunden	Kunden des Vereins	<ul style="list-style-type: none"><li>• Kaufen von Artikeln im Shop des Vereins</li><li>• Einfaches Buchen von Veranstaltungen</li></ul>
Vorstand	Der Vorstand des Vereins, der gleichzeitig die Website verwaltet	<ul style="list-style-type: none"><li>• Übersicht über alle Aktivitäten erhalten</li><li>• Website verwalten</li></ul>
Artisten	Mitglieder des Vereins, die Veranstaltungen und Workshops gestalten.	<ul style="list-style-type: none"><li>• Sperrzeiten eintragen</li><li>• Überblick über ihre Termine</li><li>• Einfach erweiterbare Anwendung</li></ul>
Entwickler	Entwickler, die die Anwendung implementieren oder später warten.	<ul style="list-style-type: none"><li>• Wenig Wartungsaufwand</li><li>• Einfache Debugging-Mechanismen</li></ul>

# Systemgrenze und Top-Level-Architektur

## Kontextdiagramm



## Top-Level-Architektur



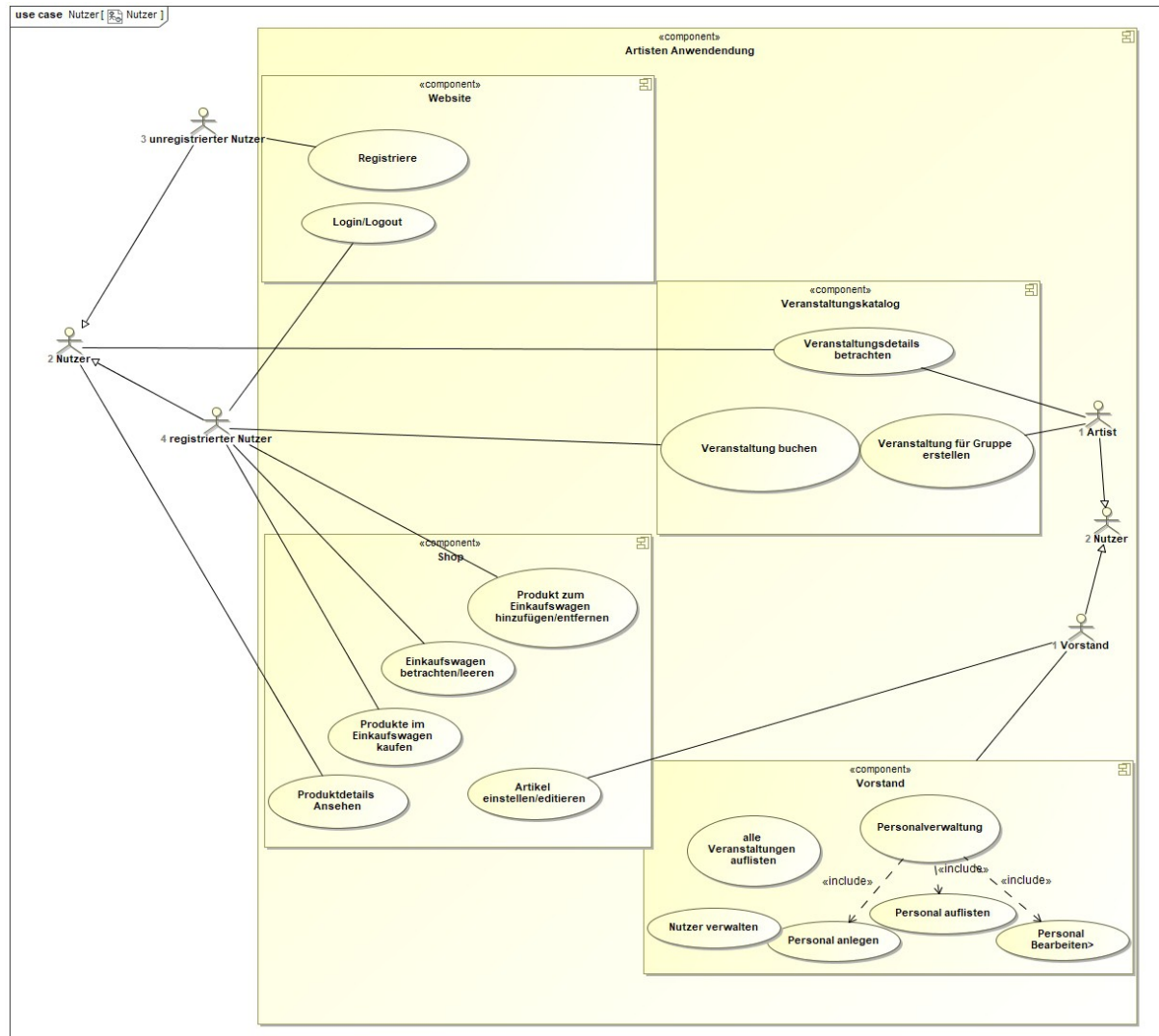
# Anwendungsfälle

## Akteure

Akteure sind die Benutzer des Software-Systems oder Nachbarsysteme, welche darauf zugreifen. Dokumentieren Sie die Akteure in einer Tabelle. Diese Tabelle gibt einen Überblick über die Akteure und beschreibt sie kurz.

Name	Description
Nutzer	Repräsentiert alle Personen die die Website besuchen und nutzen.
Kunde	Repräsentiert einen angemeldeten Nutzer vom Typ „Kunde“. Dieser kann Artikel kaufen und Veranstaltungen buchen.
Vorstand	Mitglied des Vorstands des Vereins und Benutzer vom Typ „Vorstand“. Administriert die Website.
Artist	Mitglieder des Vereins und Nutzer vom Typ „Artist“. Diese können Sperrzeiten festlegen und Termine einsehen.

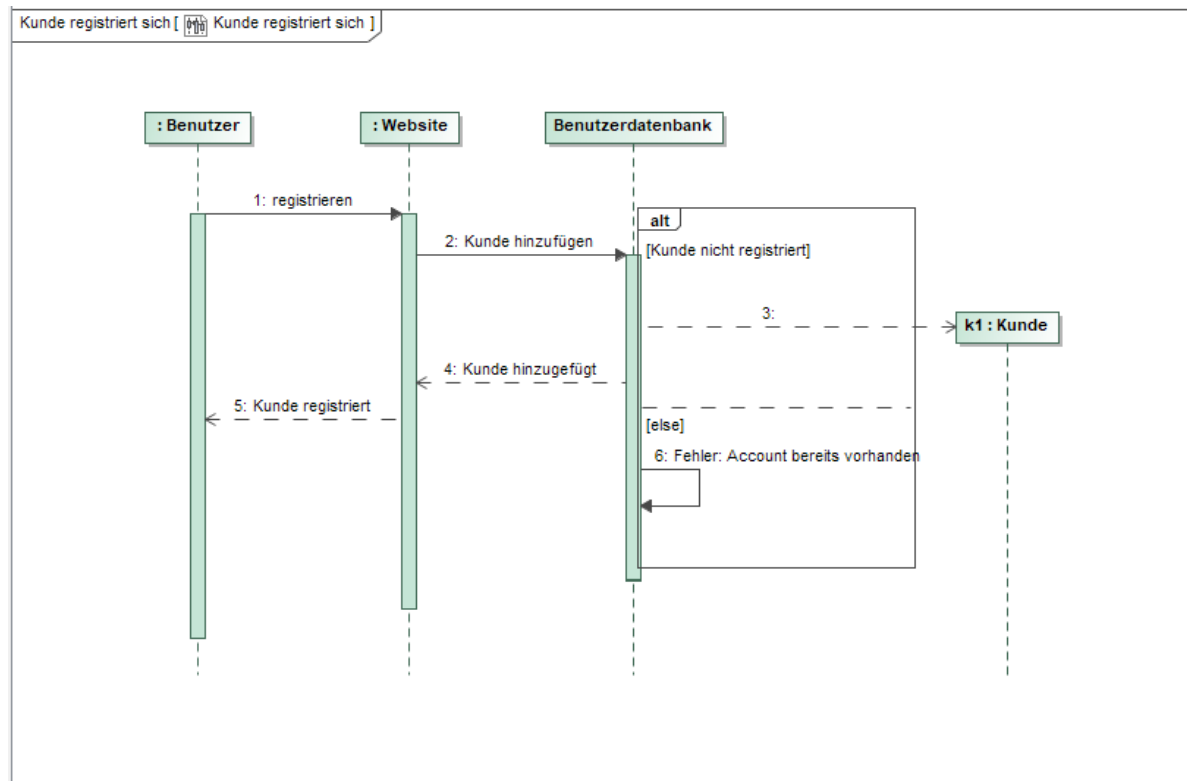
# Überblick Anwendungsfalldiagramm





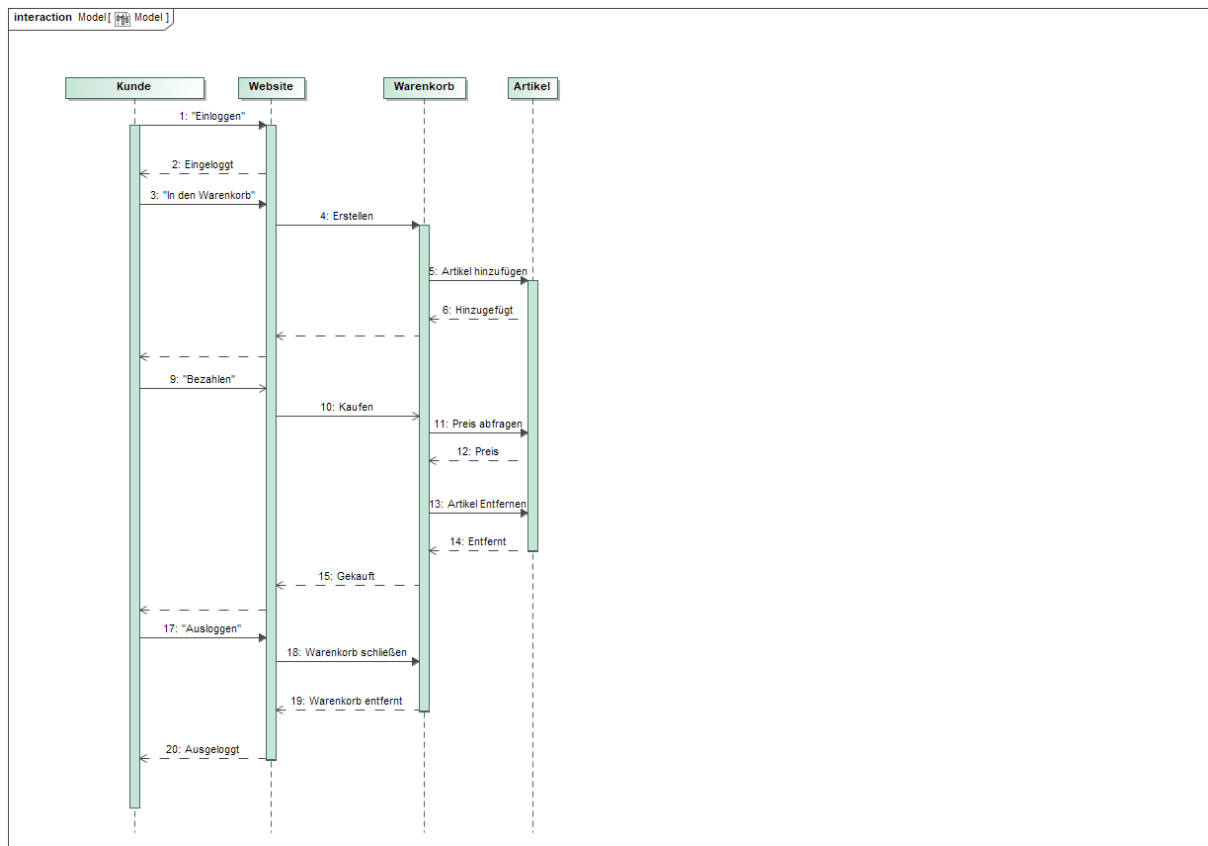
## Anwendungsfallbeschreibungen

### Kunde wird registriert:



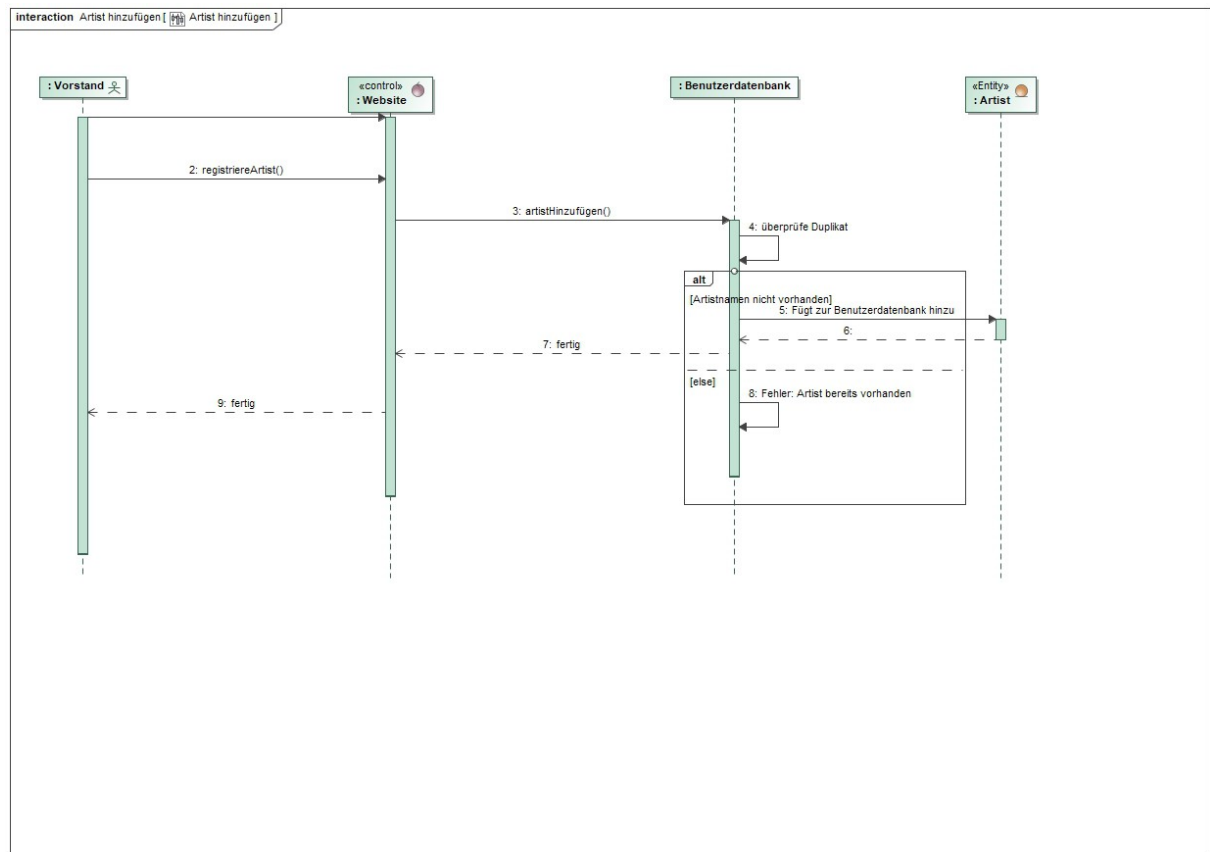
Wenn sich ein Kunde registrieren möchte, wird über die Website eine Nachricht an die Benutzerdatenbank gesendet, welche den Kunden hinzufügt, wenn noch kein Konto mit dem gleichen Namen vorhanden ist.

## Kunde kauft einen Artikel:



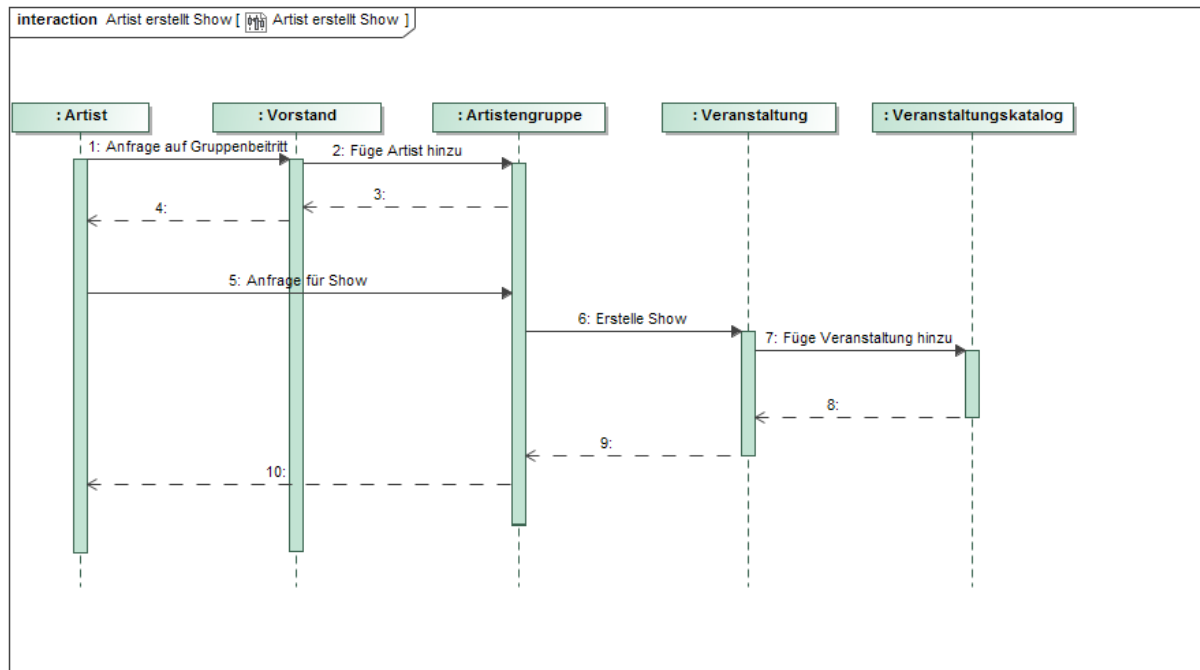
Der Kunde befindet sich im Shop. Er wählt einen Artikel aus und drückt auf „in den Warenkorb“ Knopf. Der Warenkorb erstellt einen Posten mit dem Artikel. Wenn der Kunde bezahlen drückt, erfragt der Warenkorb die Preise aller Artikel, zählt diese zusammen. Anschließend wird der Artikel aus dem Warenkorb entfernt und der Kunde hat somit eingekauft.

## Artist wird hinzugefügt:



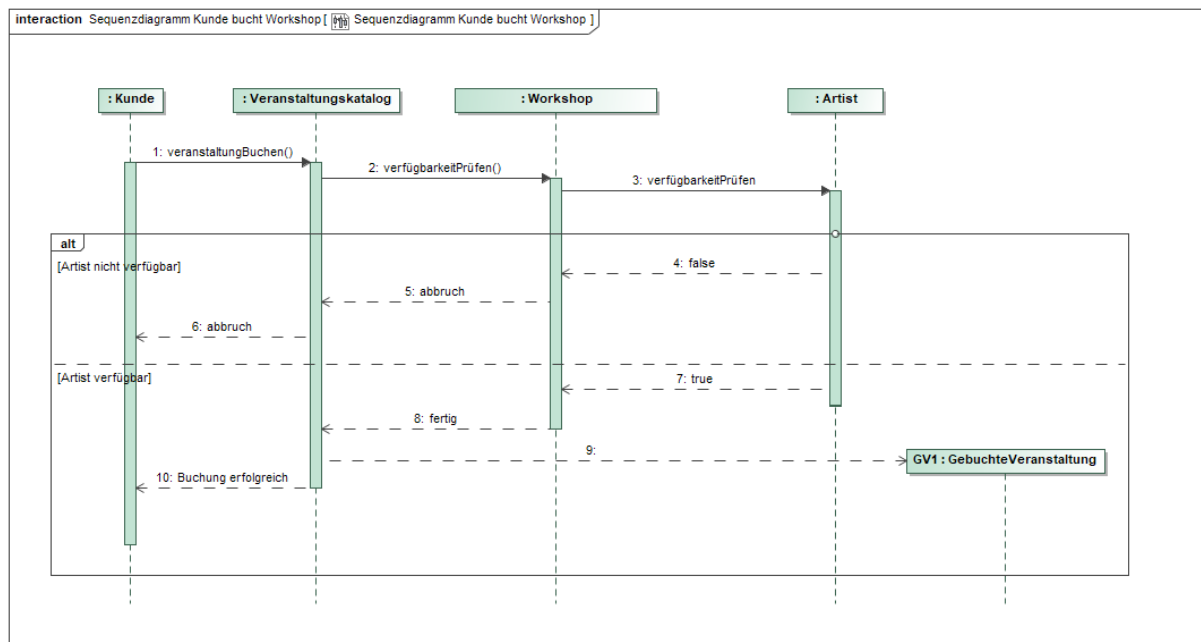
Der Vorstand kann über die Webseite das Personalverwalten. Wenn ein Artist zur Datenbank hinzugefügt werden soll, erfolgt erst eine Überprüfung der eingetragenen Daten und ob der Artist bereits in der Datenbank vorhanden ist. Falls die Überprüfung erfolgreich ist, wird der Artist als Datenbankeintrag angelegt.

## Artist erstellt Show:



Wenn ein Artist eine Show erstellen möchte, muss er erst der Artistengruppe beitreten, mit welcher er diese Veranstaltung geplant hat. Nur der Vorstand kann ihn in diese Gruppe einfügen, weswegen er zunächst eine Anfrage an ihn senden muss. Er wird nun der Gruppe hinzugefügt und kann eine Anfrage für eine Show stellen. Die Anfrage wird nun schließlich gespeichert und im Veranstaltungskatalog veröffentlicht, wo nun Kunden diese einsehen und buchen können.

## Kunde bucht Workshop:



Möchte eine Kunde einen Workshop buchen so fragt er dies beim Veranstaltungskatalog an. Dieser lässt die Verfügbarkeit des ausgewählten Workshops prüfen der dazu wiederum den veranstaltenden Artist aufruft und dessen Verfügbarkeit anfragt. Ist die Antwort negativ, so gibt der Workshop dies an den Veranstaltungskatalog weiter der wiederum dem Kunden eine Fehlermeldung ausgibt. Ist die Antwort des Artisten hingegen positiv so erstellt der Veranstaltungskatalog eine Gebuchte Veranstaltung und gibt dem Kunden ein positives Feedback.

# Funktionale Anforderungen

## Muss-Kriterien:

### Artisten:

- Jeder Artist kann für sich Workshops erstellen, bearbeiten und entfernen.
- Artisten in Artistengruppen sollen Shows erstellen können.
- Die Buchungen sollen im Terminkalender (in Kalenderansicht) der Artisten festgehalten werden.
- Wenn ein Artist verbucht ist, sind auch die Gruppenshows von diesem nicht mehr buchbar.
- Artisten sollen Sperrzeiten in ihrem Profil anlegen können.
- Artisten sollen eine Übersicht über ihre Termine, ihre Kunden, ihre Shows, Workshops und ihre Artistengruppen haben.
- Anfahrtszeiten sollen nicht mit beachtet werden (im Terminkalender).
- Artisten soll es möglich sein Termine(Shows, Workshops) abzusagen. In diesem Fall bekommt der Kunde sein Geld zurück.

### Kunden:

- User können den Shop und Veranstaltungsangebote (Workshops, Shows) ansehen. Bei Registrierung werden sie zu Kunden und können diese buchen/ bestellen.
- User sollen Konten anlegen können (Name, Email, Telefonnummer, Passwort, Nutzernamen(Überprüfung auf Einmaligkeit)).
- Kunden sollen ihre Konten auch bearbeiten können.
- Kunden können selbst den Zeitpunkt des Workshops/Show(nur im angegebenen Zeitrahmen) und den Ort auswählen und buchen.

- Die Buchungen des Kunden werden in einer Kalenderansicht dargestellt.
- Für Orte außerhalb müssen die Kunden die Kosten übernehmen (Anfahrt, Verpflegung, Unterbringung).
- Der Kunde soll eine Übersicht über seine Workshops, Shows, bestellte Artikel, angeheuerte Artistengruppen und bereits bezahlte Kosten haben.
- Kunden sollen Veranstaltungen bewerten und kommentieren können. Zudem soll ihnen eine Liste der Veranstaltungen mit den besten Bewertungen empfohlen werden.
- Dem Kunden ist es möglich Termine (Shows oder Workshops) abzusagen. Er bekommt sein Geld jedoch nicht zurück.

#### Vorstand des Vereins:

- Das Konto des Vorstands soll Administratorrechte bekommen. (Wird vom Programmiererteam erstellt, nur Passwort noch selbst wählbar)
- Er soll Einsicht auf alle Kunden, Artisten und Shopartikel haben. Hierbei kann er sich eine Sortierung aussuchen (z.B. nach Datum oder nach Alphabet). Zudem soll er eine Bestellübersicht und eine Finanzansicht haben, zu denen er sich jeweils Details anzeigen lassen kann.
- Der Vorstand kann auf einen Zeitplan des zur Verfügung stehenden Raums haben (er kann nun in Kalenderansicht sehen, wann der Raum bereits verbucht ist).
- Er soll Artistenkonten und Shopartikel anlegen, bearbeiten und löschen können.
- Artistengruppen werden von ihm angelegt.
- Der Vorstand kann allgemeine Sperrzeiten (in denen weder Workshops noch Shows stattfinden können) bestimmen.
- Er legt den Rabatt für die Kunden, welche an Workshops teilnehmen, auf den Shop fest.
- Der Vorstand kann einen Mindestbestand von bestimmten Artikeln im Shop angeben. Diese werden dann automatisch bestellt, wenn diese Zahl unterschritten wurde.

#### Workshops/Shows:

- Workshops sollen in der eigenen Halle gebucht werden können oder an einem beliebigen Ort außerhalb.
- Bei Erstellung soll Name der Show/Workshop, Zeitraum (in dem dieser angeboten werden soll), Dauer, Kosten und Beschreibung gespeichert werden.

- Es soll möglich sein einem Workshop oder einer Show Artikel aus dem Shop zuzuordnen (mit denen z.B. im Workshop gearbeitet wird)
- Die Shows und Workshop sollen in einer Liste dargestellt werden. Bei Klick auf die entsprechende Veranstaltung soll eine Detailansicht erscheinen.

#### Shop:

- Die Artikel werden sofort beim Kauf bezahlt, sodass es keine offenen Rechnungen gibt.
- Die Artikel im Shop sollen Bild, Name, Beschreibung und Preis erhalten.
- Es soll eine Suchfunktion geben.
- Rabatte sind nur in der Zeitdauer bis zum Workshop möglich.
- Es gibt einen Warenkorb in den man Artikel hinzufügen, einzelne Artikel wieder löschen, die Anzahl der Artikel ändern aber auch den ganzen Warenkorb leeren kann.
- Im Lager werden ausverkaufte Artikel automatisch nachbestellt.

#### **Kann-Kriterien:**

##### Kunden:

- Kunden können einen Newsletter abonnieren und erhalten Nachrichten zu neuen Produkten/ Workshops und Shows.

##### Vorstand:

- Kann Konten sperren (unzuverlässige Kundenkontos).

##### Shop:

- Artikel können durch Kunden bewertet werden.(Am besten nur durch Kunden, welche dieses Produkt auch bestellt haben)
- Produkte, die sich nicht verkaufen (und schlecht bewertet werden), werden (vom Vorstand) aus dem Sortiment entfernt.
- Für die Artikel gibt es verschiedene Sortierungen (z.B. nach Bewertung, nach Preis(niedrig/hoch), usw)
- Im Shop kann nach bestimmten Artikeln gesucht werden.



### Sonstiges:

- Man soll im Browser vor- und rückwärts navigieren können.

## Nicht-Funktionale Anforderungen

### Qualitätsziele

1- sehr wichtig, 5- nicht wichtig

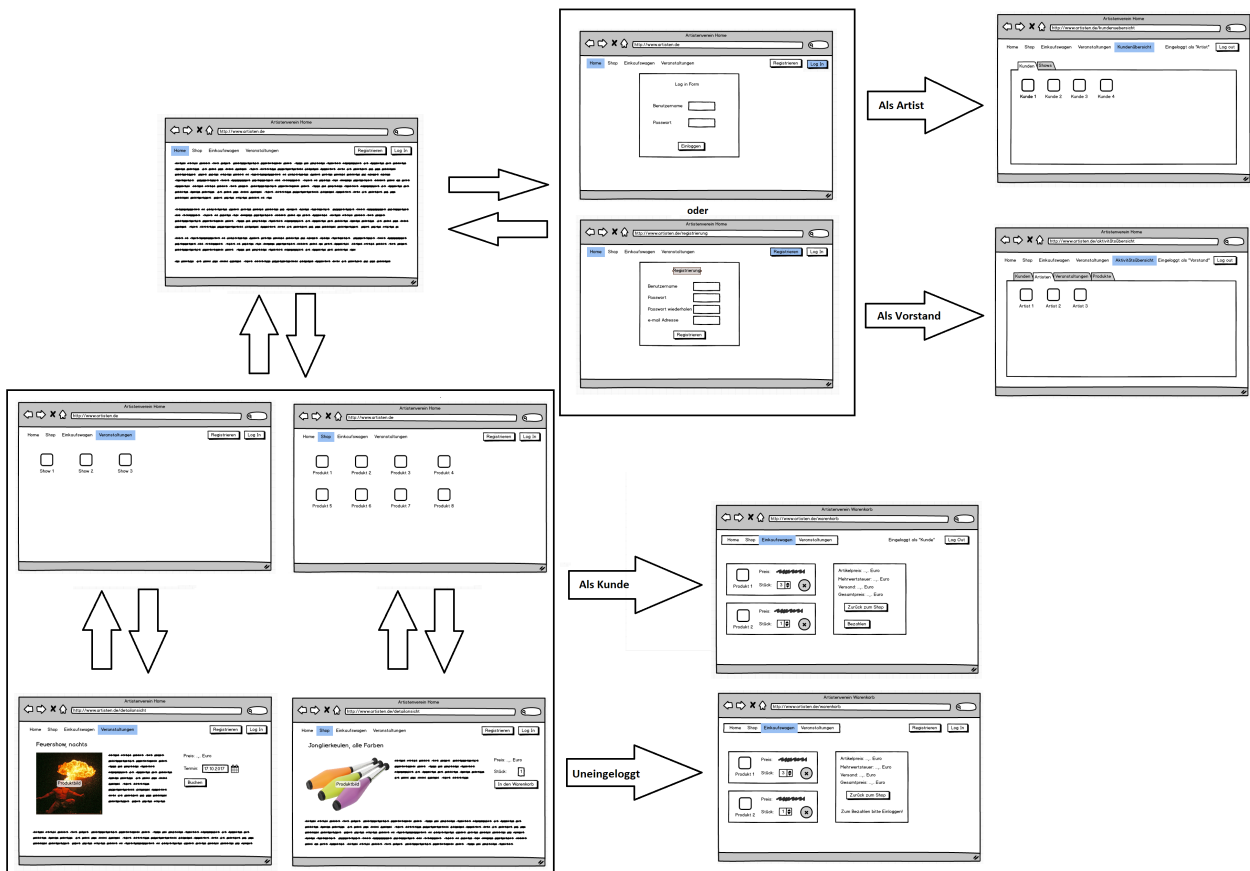
Produktivität	1	2	3	4	5
Funktionalität		+			
Zuverlässigkeit	+				
Effizienz				+	
Benutzbarkeit			+		
Änderbarkeit			+		
Validierung				+	

### Konkrete Nicht-Funktionale Anforderungen

Passwort- Sicherheit: Die Passwörter der Kunden sollen nur in Hash-Werten gespeichert werden.

# GUI Prototyp

## Überblick: Dialoglandkarte

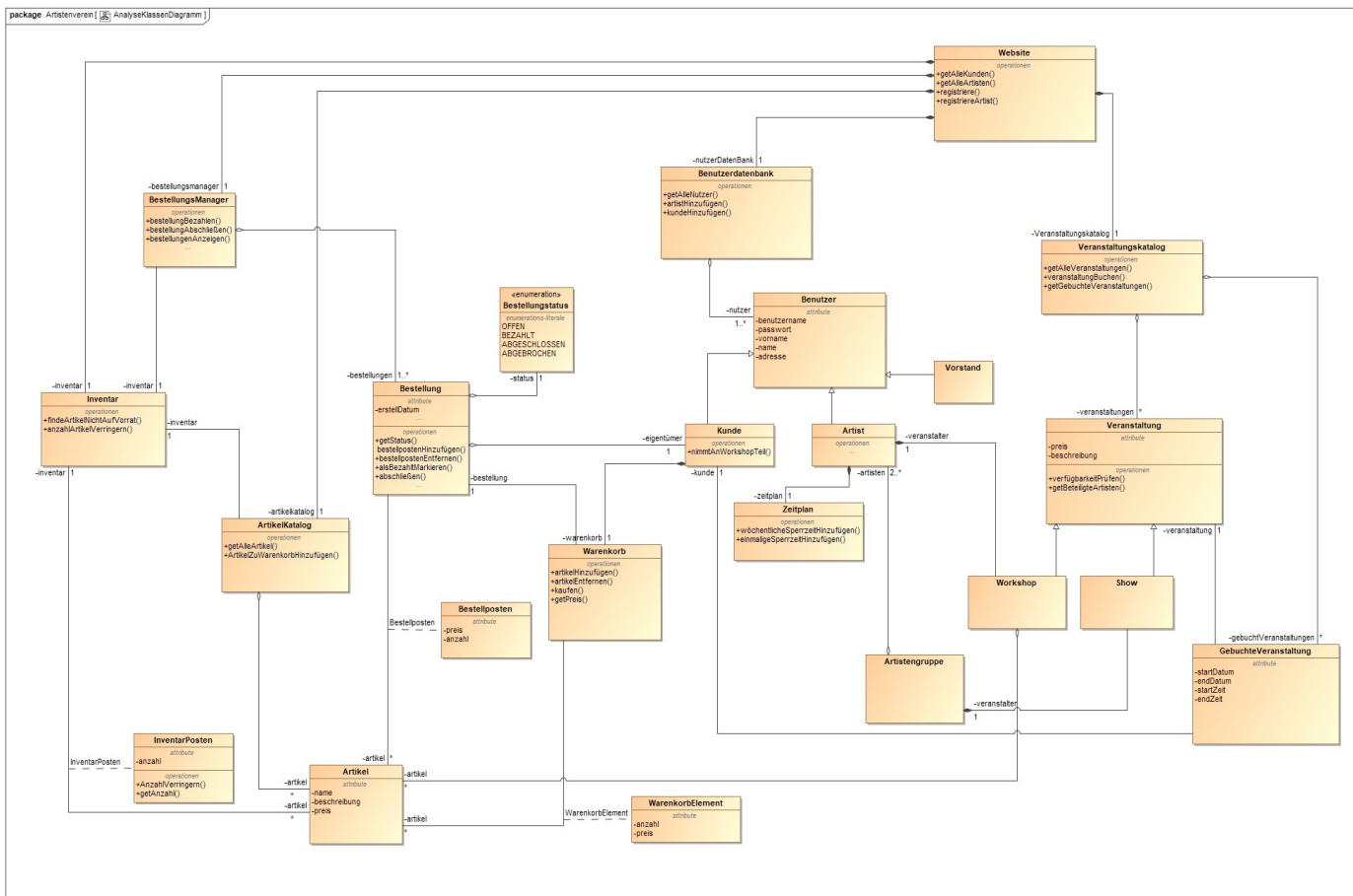


## Dialogbeschreibung

Element	Beschreibung
Tabs	von jeder Seite erreichbar mit jeweiligen Rechten

Buttons	nur von auf bestimmten Seiten aus erreichbar
Homescreen -> Registrierungsscreen	Als Benutzer der Seite kann man auf dem Homescreen auf den Registrieren-Button klicken, woraufhin die Oberfläche sich zum Registrierenscreen wechselt. Nach der Registrierung landet man wieder auf dem Homescreen.
Homescreen -> Loginscreen	Wenn man ein Kunde, Artist oder der Admin ist, kann man über den Homescreen auf einen Loginbutton drücken.  Nachdem man sich angemeldet hat, kommt man auf den Homescreen zurück.
"Beliebiger" - Screen -> Homescreen	Man kann über einen Tab von jeder Seite aus auf den Homescreen gehen.
"Beliebiger" - Screen -> Veranstaltungsscreen	Man kann von jeder Seite aus, egal ob oder als was man angemeldet ist, über Tabs zum Veranstaltungsscreen.  Dort werden listenartig alle Veranstaltungen angezeigt, mit der jeweiligen Artistengruppe, einer Beschreibung,..
"Beliebiger" - Screen -> Shopscreen	Ähnlich wie oben, nur dass alle Artikel angezeigt werden, die man kaufen kann.
Shopscreen -> Einkaufswagen -> Bezahlenscreen	Man kann im Shop auf einen Button drücken, um in den Einkaufswagen zu kommen. Von dort kann man über einen weiteren Button bezahlen oder Produkte entfernen.
Kundenübersicht - Screen	Als Artist hat man einen zusätzlichen Tab, worüber man seine Kunden, Shows etc. einsehen kann.
Aktivitätsübersicht - Screen	Dieser Tab ist nur für den Vorstand zugänglich. Dort hat er eine Übersicht über alle Kunden, Artisten und Veranstaltungen.

## Überblick: Klassendiagramm



## Klassen und Enumerationen

Klasse/Enumeration	Beschreibung
<b>Artikel</b>	Stellt einen abstrakten im Shop angebotenen Artikel dar.
<b>ArtikelKatalog</b>	Verwaltet alle im Shop angebotenen abstrakten Artikel.
<b>Artist</b>	Stellt einen einzelnen Artisten dar.
<b>Artistengruppe</b>	Stellt eine Gruppe von Artisten dar, die zusammen Veranstaltungen anbieten/durchführen. Besteht aus mehrern Benutzern des Typs Artist.
<b>Benutzer</b>	Stellt einen einzelnen Nutzer dar. Wird verfeinert durch die Klassen „Kunde“, „Artist“ und „Vorstand“.
<b>Benutzerdatenbank</b>	Verwaltet sämtliche Nutzer.
<b>Bestellposten</b>	Stellt einen Einzelnen Posten in einer Bestellung dar.
<b>Bestellung</b>	Repräsentiert eine Bestellung eines Kunden, ermöglicht die Abwicklung des Bezahlvorgangs.
<b>BestellungsManager</b>	Verwaltet sämtliche Bestellungen.
<b>Bestellungsstatus</b>	Bezeichnet den Status eine Bestellung. OFFEN: Bestellung wurde erstellt, aber noch nicht bearbeitet. BEZAHLT: Bestellung wurde vom Kunden bezahlt, Ware muss noch aus Lager entfernt werden. ABGESCHLOSSEN: Bestellung wurde bezahlt und versandt. ABGEBROCHEN: Bestellung wurde durch einen Fehler oder ein anders Problem abgebrochen.
<b>GebuchteVeranstaltung</b>	Eine gebuchte Veranstaltung stellt eine konkrete Buchung mit Zeit, Typ und buchendem Kunden dar.

Klasse/Enumeration	Beschreibung
<b>Inventar</b>	Verwaltet alle realen Artikel im Lager.
<b>InventarPosten</b>	Repräsentiert einen Typ von Artikeln im Lager sowie dessen Vorrat.
<b>Kunde</b>	Stellt einen einzelnen Kunden dar.
<b>Show</b>	Die Show ist ein Typ einer buchbaren Veranstaltung.
<b>Veranstaltung</b>	Repräsentiert eine buchbare Veranstaltung, siehe auch „Workshop“ und „Show“.
<b>Veranstaltungskatalog</b>	Verwaltet sämtliche buchbaren und gebuchten Veranstaltungen.
<b>Vorstand</b>	Stellt ein Mitglied des Vorstands dar.
<b>Warenkorb</b>	Verwaltet die Artikel die der Kunde kaufen möchte.
<b>WarenkorbElement</b>	Repräsentiert einen einzelnen Posten im Warenkorb des Kunden.
<b>Website</b>	Die zentrale Klasse, repräsentiert die Website an sich.
<b>Workshop</b>	Der Workshop ist eine Typ einer buchbaren Veranstaltung.
<b>Zeitplan</b>	Verwaltet die Sperrzeiten eines Artisten. Bietet Möglichkeiten, wöchentliche oder einmalige Sperrzeiten hinzuzufügen.

# Akzeptanztestfälle

<b><u>Testfall: Produkt Details im Shop anschauen</u></b>	
Vorbedingungen	Im Shop existieren bereits Produkte
Event	Sowohl registrierter, als auch nicht registrierter Benutzer können auf ein Produkt klicken
Erwartetes Ergebnis	Zu dem angeklickten Produkt werden weitere Details angezeigt(z.b. Wie viele davon noch vorhanden sind, Produktbeschreibung,..)

<b><u>Testfall: Übersicht über Einkäufe (Vorstand)</u></b>	
Vorbedingungen	Vorstand ist eingeloggt
Event	Der Vorstand kann eine Übersicht über alle bereits gekauften Produkte öffnen
Erwartetes Ergebnis	Es öffnet sich eine Liste, in der alle bereits gekauften Produkte (sortiert nach Kaufdatum, das letzte zuerst) angezeigt werden. Zusätzlich wird zu jedem Produkt der eingennommene Preis angezeigt. Falls noch keine Produkte gekauft wurden erscheint der Text: „Es wurden noch keine Produkte gekauft“.

<b><u>Testfall: Bewertung und Kommentar zur Veranstaltung erstellen</u></b>	
Vorbedingungen	Benutzer ist registriert und eingeloggt
Event	Benutzer kann zu jeder Veranstaltung auf einen Button „Kommentar hinzufügen“ oder „bewerten“ klicken.
Erwartetes Ergebnis	Kommentar hinzufügen: Nutzer kann einen Kommentar eintippen, dann auf „Senden“ klicken. Dieser erscheint nun unter der dazugehörigen Veranstaltung unter der Überschrift „Kommentare“. Bewerten: Nutzer kann eine Bewertung in Form von Schulnoten abgeben. Aus allen Bewertungen derselben Veranstaltung wird der Mittelwert gebildet und dieser wird unter dieser Veranstaltung unter der Überschrift „Bewertung“ angezeigt.

### **Testfall: Nutzung der Suchfunktion im Shop**

Vorbedingungen	keine
Event	Der Nutzer kann im Feld „Suche:“ einen Begriff eingeben, nachdem er den Shop durchsuchen will und dann auf den Button „Suchen!“ klicken.
Erwartetes Ergebnis	Bei erfolgreicher Suche werden die gefundenen Produkte , welche den Suchbegriff enthalten aufgelistet. Bei fehlgeschlagener Suche wird der Text:„Es gab keine Übereinstimmungen.“ angezeigt.

### **Testfall: Registrieren**

Vorbedingungen	Der User ist auf der Homepage.
Event	Der User drückt auf einen Registrieren-Button.
Erwartetes Ergebnis	Er kommt auf den Registrieren-Screen. Wo er Name, Passwort und Email angeben kann.
Event	User gibt seine Daten an und drückt auf einen Registrieren-Button.
Erwartetes Ergebnis	Er kommt auf die Home. Nun hat er die Rechte eines Kunden.

### **Testfall: Produkte im Einkaufswagen kaufen**

Vorbedingungen	Mindestens ein Produkt im Einkaufswagen
Event	Ein User ist im Einkaufswagen und drückt auf den Kaufen-Button.
Erwartetes Ergebnis	Es erscheint ein Nachricht auf dem gleichen Screen, dass er die Waren gekauft hat, wenn er als Kunde eingekauft hat. Ansonsten wird er auf den Login-Screen gewiesen.



**Testfall: „Kein“ Produkt im Einkaufswagen kaufen**

Vorbedingungen	Kein Produkt im Einkaufswagen
Event	Ein User ist im Einkaufswagen und drückt auf den Kaufen-Button.
Erwartetes Ergebnis	Es erscheint eine Nachricht auf dem gleichen Screen, dass mindestens ein Produkt im Einkaufswagen sein muss.

**Testfall: Kunde bucht Veranstaltung, woraufhin eine andere Veranstaltung nicht mehr buchbar ist**

Vorbedingungen	Zwei Veranstaltungen mit einem Artisten der in beiden auftreten würde, überlappen sich in der Zeit.
Event	Kunde bucht eine dieser Veranstaltungen.
Erwartetes Ergebnis	Sobald die Veranstaltung gebucht wurde, verschwindet kann die andere Veranstaltung zu dieser Zeit nicht mehr gebucht werden.

**Testfall: Automatische Nachbestellung von Artikeln**

Vorbedingungen	Im Inventar existieren Produkteinträge, eine Mindestvorratsmenge ist festgelegt.
Event	Vorrätige Menge eines Artikels im Inventar fällt unter die Mindestmenge
Erwartetes Ergebnis	Der Vorrat jedes Artikels im Inventar wird überprüft. Ist ein Artikel nicht in ausreichender Stückzahl vorrätig wird er nachbestellt bis dies der Fall ist.

### Testfall: Veranstaltungskatalog betrachten

Vorbedingungen	Der User ist auf der Homepage
Event	Der User navigiert auf der Homepage zum Katalog.
Erwartetes Ergebnis	Der User landet auf einer Übersicht mit allen verfügbaren bzw buchbaren Workshops und Shows.

### Testfall: Kundenliste Anschauen

Vorbedingungen	Der User ist als Vorstand registriert.
Event	Der User navigiert auf der Homepage zur Nutzerverwaltung. In der Nutzerverwaltung wählt er die Kundenübersicht aus.
Erwartetes Ergebnis	Alle registrierten Kunden werden aufgelistet

### Testfall: Eintragen von Sperrzeiten

Vorbedingungen	Der User ist als Artist registriert.
Event	User navigiert zu seinem Profil. In seinem Profil wählt er die Sperrzeitverwaltung aus. User wählt Zeitraum aus in dem er eine Sperrzeit hinzufügen, bearbeiten oder löschen möchte.
Erwartetes Ergebnis	Sperrzeit wird geupdatet. Bestehende Veranstaltungen werden entsprechend der Sperrzeit des Users verfügbar oder nicht buchbar.

**Testfall: Produkt zum Einkaufswagen hinzufügen**

Vorbedingungen	Im Shop existieren bereits Produkte
Event	Sowohl registrierter, als auch nicht registrierter Benutzer können ein Produkt dem Warenkorb hinzufügen
Erwartetes Ergebnis	Das Ausgewählte Produkt wird dem Warenkorb des Kunden hinzugefügt. Ist der Kunde nicht angemeldet passiert dies nur temporär und lokal. Ist der Kunde angemeldet wird die Änderung des Warenkorbs serverseitig übernommen.

**Testfall: Preisbestimmung mit Rabatt / eingeloggt / Kunde nimmt an Workshop teil**

Vorbedingungen	Kunde ist eingeloggt, Kunde hat einen Workshop gebucht
Event	Ein Preis wird abgefragt (im Warenkorb oder auf der Produktseite)
Erwartetes Ergebnis	Der Normalpreis wird erst bestimmt und dann mit dem festgelegten Rabatt verrechnet. Dieser neue Preis wird dem Kunden angezeigt.

**Testfall: Preisbestimmung mit Rabatt / eingeloggt / Kunde nimmt nicht an Workshop teil**

Vorbedingungen	Kunde ist eingeloggt, Kunde hat keinen Workshop gebucht
Event	Ein Preis wird abgefragt (im Warenkorb oder auf der Produktseite)
Erwartetes Ergebnis	Der Normalpreis wird bestimmt und unverändert angezeigt.

### **Testfall: Preisbestimmung mit Rabatt / nicht eingeloggt**

Vorbedingungen	keine
Event	Ein Preis wird abgefragt (im Warenkorb oder auf der Produktseite)
Erwartetes Ergebnis	Der Normalpreis wird bestimmt und unverändert angezeigt.

### **Testfall: Artist erstellt Show**

Vorbedingungen	Artist ist angemeldet und Teil mindestens einer Artistengruppe.
Event	Ein Artist erstellt eine neue Show für eine seiner Gruppen.
Erwartetes Ergebnis	Der Artist drückt den Button Show erstellen auf seiner Übersichtsseite. Auf der nächsten Seite kann er Beschreibung, Bilder und Preis eintragen. Die Show wird anschließend auf der Website sicht- und buchbar.

### **Testfall: Login**

Vorbedingungen	Es gibt mindestens einen Benutzer im System
Event	Ein nicht authentifizierter Benutzer der Seite klickt auf „Einloggen“, gibt die Benutzerdaten (Name = hans und Passwort = 123) eines existierenden Benutzers ein und klickt auf anmelden.
Erwartetes Ergebnis	Der Benutzer ist eingeloggt und wird zu einem Willkommens Bildschirm weitergeleitet. Der Nutzer hat jetzt Zugang zu allen Funktionen, die für die Rolle „Kunde“ zugänglich sind.

### **Testfall: Logout**

Vorbedingungen	Ein Benutzer ist angemeldet
Event	Ein angemeldeter Benutzer der Seite klickt auf „Ausloggen“.
Erwartetes Ergebnis	Der Benutzer wird ausgeloggt, auf eine Startseite weitergeleitet und hat keinen Zugriff mehr auf Funktionen einer Rolle.

### **Testfall: Einkaufswagen überprüfen**

Vorbedingungen	Ein Kunde ist angemeldet
Event	Der Kunde klickt auf den Reiter „Warenkorb“
Erwartetes Ergebnis	Der Kunde sieht alle Produkte im Warenkorb mit Anzahl und Preis. Außerdem wird ein Gesamtpreis gezeigt und der Kunde hat die Möglichkeit die Produkte im Warenkorb zu bestellen oder einzelne Produkte aus dem Warenkorb zu entfernen.

### **Testfall: Teilnahme an einem Workshop**

Vorbedingungen	Ein Kunde ist angemeldet und befindet sich im Reiter Workshops
Event	Der Kunde klickt auf einen Workshop, kommt in eine Detailansicht, gibt ein Datum und Uhrzeit ein und klickt auf buchen.
Erwartetes Ergebnis	Der Workshop ist gebucht und ist für den Kunden einsehbar.

### **Testfall: Sperrung eines Kontos (Kann-Kriterium)**

Vorbedingungen	Ein Vorstand ist angemeldet und befindet sich in der Kundenübersicht.
Event	Der Vorstand klickt bei einem Kundeneintrag auf „Sperrern“ und bestätigt den Vorgang.
Erwartetes Ergebnis	Der Kunde wird gesperrt. Wenn er versucht, sich einzuloggen bekommt er eine entsprechende Fehlermeldung.

## **Glossar**

**Kunde/User:** Ein User ist ein nichtregistrierter Benutzer der Website, wohingegen ein Kunde einen registrierter Benutzer darstellt.

## **Offene Punkte**