



# xShare

Studiengang Kommunikationsinformatik  
an der Dualen Hochschule Baden-Württemberg Stuttgart

von

Team Beta bestehend aus

**Cihan Aktas, Julian Diehlmann, Christine Krüger, Matthias Reckmeyer,  
Steffen Weinstein**

April 2016

**Bearbeitungszeitraum**  
**Kurs**  
**Betreuer**

11 Wochen  
TINF13K  
Miroslav Juric

# Inhaltsverzeichnis

<b>Abbildungsverzeichnis</b>	<b>1</b>
<b>Tabellenverzeichnis</b>	<b>2</b>
<b>1 Einleitung</b>	<b>3</b>
1.1 Was ist xShare? . . . . .	3
1.2 Was unterscheidet xShare von ähnlichen Produkten? . . . . .	3
1.3 Für wen ist xShare? . . . . .	3
<b>2 Brainstorming</b>	<b>4</b>
<b>3 Produkt-Userstory</b>	<b>5</b>
<b>4 Konkurrenzanalyse</b>	<b>6</b>
4.1 Applikation . . . . .	6
4.2 Web-Applikation . . . . .	7
4.3 Vergleich mit unserer Idee . . . . .	7
<b>5 Vorgehensmodel</b>	<b>9</b>
5.1 Steakholder . . . . .	9
5.2 Product Owner . . . . .	9
5.3 Scrum Master . . . . .	10
5.4 Entwicklungsteam . . . . .	10
5.5 Vorgehensweise . . . . .	10
5.6 Software-Lebenszyklus . . . . .	10
<b>6 Features</b>	<b>12</b>
6.1 Produkt verleihen . . . . .	12
6.2 Produkt suchen . . . . .	14
6.3 Informationen angezeigt bekommen . . . . .	15
6.4 Produkt ausleihen . . . . .	16
6.5 Über Veränderungen informiert werden . . . . .	17
6.6 Informationen über Benutzer erhalten . . . . .	18
6.7 In Kontakt treten . . . . .	18
6.8 Benutzer bewerten . . . . .	19
6.9 Regelverstöße melden . . . . .	19
6.10 Account verwalten . . . . .	20
6.11 Benutzer authentifizieren . . . . .	20

6.12	Über Rechte und Pflichten aufklären . . . . .	21
6.13	Benutzer animieren . . . . .	22
6.14	Endanwendung benutzerfreundlich gestalten . . . . .	22
6.15	Geld verdienen . . . . .	23
6.16	Benutzer verwalten . . . . .	24
6.17	Produkt verwalten . . . . .	24
6.18	Endanwendung verwalten . . . . .	25
<b>7</b>	<b>Use-Cases</b>	<b>26</b>
<b>8</b>	<b>Use-Case-Beschreibungen</b>	<b>30</b>
8.1	Produkt suchen . . . . .	31
8.2	Benutzer löschen . . . . .	32
8.3	Benutzer authentifizieren . . . . .	33
8.4	Produkt anbieten . . . . .	34
8.5	Anfrage erstellen . . . . .	35
8.6	Produkt bewerten . . . . .	36
<b>9</b>	<b>Klassendiagramme</b>	<b>37</b>
<b>10</b>	<b>Sequenzdiagramme</b>	<b>40</b>
<b>11</b>	<b>Mock-Ups</b>	<b>43</b>
11.1	Produkt suchen . . . . .	43
11.2	Produktdetails anzeigen . . . . .	44
11.3	Produkt verleihen . . . . .	45
<b>12</b>	<b>Projekt-Resume</b>	<b>47</b>

# Abbildungsverzeichnis

2.1 Brainstorming . . . . .	4
7.1 Use-Cases der Akteure . . . . .	26
7.2 Use-Cases der Benutzerverwaltung . . . . .	27
7.3 Use-Cases der Endanwendung verwalten . . . . .	27
7.4 Use-Cases der Suche Ausleih Bewertung . . . . .	28
7.5 Use-Cases der Verleih . . . . .	29
9.1 Klassendiagramme der Akteure . . . . .	37
9.2 Klassendiagramme der Klassen . . . . .	38
9.3 Klassendiagramme der Controller . . . . .	39
10.1 Sequenzdiagramm der Suche . . . . .	40
10.2 Sequenzdiagramm der Anfrage, negativer Fall . . . . .	41
10.3 Sequenzdiagramm der Anfrage, positiver Fall . . . . .	42
11.1 Mock-up: Suche, Teil 1 . . . . .	43
11.2 Mock-up: Suche, Teil 2 . . . . .	44
11.3 Mock-up: Verleihen - Produktinfos eingeben . . . . .	45
11.4 Mock-up: Verleihen - ohne Kauton eingeben . . . . .	46
11.5 Mock-up: Verleihen - mit Kauton eingeben . . . . .	46

# Tabellenverzeichnis

8.1	Use-Case-Beschreibung: Produkt suchen . . . . .	31
8.2	Use-Case-Beschreibung: Benutzer löschen . . . . .	32
8.3	Use-Case-Beschreibung: Benutzer authentifizieren . . . . .	33
8.4	Use-Case-Beschreibung: Produkt anbieten . . . . .	34
8.5	Use-Case-Beschreibung: Anfrage erstellen . . . . .	35
8.6	Use-Case-Beschreibung: Produkt bewerten . . . . .	36

# 1 Einleitung

Sie kennen das ja von früher. Der Kaffee ist gekocht, aber Ihnen ist die Milch ausgegangen. Zum Glück haben Sie einen netten Nachbarn, von dem Sie sich diese besorgen können. Ihr Nachbar ist der Beste. Alles was Sie brauchen, finden Sie bei ihm. Leider findet man diese Art der Interaktion heutzutage viel zu selten. Jeder ist nur noch mit sich selbst beschäftigt. xShare möchte dieses Gefühl der Brüderlichkeit und Nächstenliebe zurück in den Alltag bringen.

## 1.1 Was ist xShare?

xShare soll eine digitale Plattform werden, auf der jeder Benutzer ungenutzte Dinge zum Verleih anbieten kann. Umgekehrt kann er auch Dinge von anderen Personen ausleihen. Im Fokus stehen hierbei Gegenstände, die man beispielsweise ab und zu im Haushalt gebrauchen kann, die aber nicht jeder automatisch besitzt, da sie zu selten Gebrauch finden. Zum Beispiel eine Bohrmaschine.

## 1.2 Was unterscheidet xShare von ähnlichen Produkten?

Einzigartig an xShare ist, dass Benutzer nicht aus finanziellem Profit handeln sollen, sondern eine Art Gemeinschaft bilden, in der Dinge geteilt werden. Jedem wird die Möglichkeit geboten, selten benutzte Gegenstände anzubieten und auszuleihen. xShare bietet dabei keine Möglichkeit, Geld zu verdienen.

## 1.3 Für wen ist xShare?

xShare ist für alle, die den Community-Gedanken mögen und sich gerne aktiv beteiligen. Vertrauen unter den Mitgliedern ist das oberste Gebot, da eine Plattform dieser Art nicht ohne Vertrauen funktionieren kann. In erster Linie richtet sich xShare an junge Erwachsene, denen die nötigen Mittel zum Beispiel im Haushalt fehlen.

## 2 Brainstorming

Das folgende Kapitel dokumentiert die Ergebnisse des initialen Brainstormings nach Erhalt des Projektauftrages. Ziel war hierbei die Sammlung von Ideen, was die neue App leisten mussten und was nötig ist um die Ziele zu erreichen.

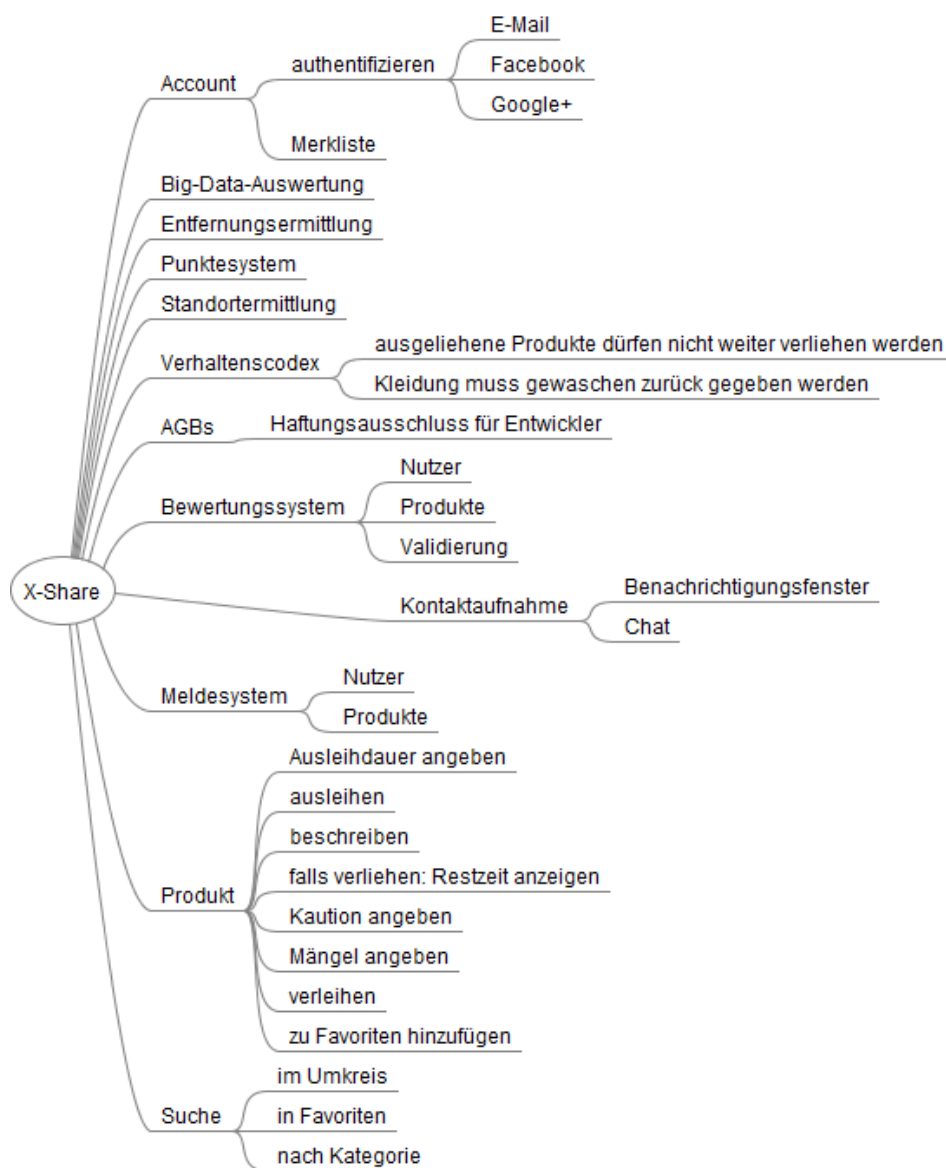


Abbildung 2.1: Brainstorming

### 3 Produkt-Userstory

Aus dem Brainstorming, das in Abbildung 2.1 [auf der vorherigen Seite](#) zu sehen ist, haben wir die Produkt-Userstory abgeleitet. Dazu haben wir die relevanten Punkte abstrahiert und sie aus der Sicht der unterschiedlichen Anwender formuliert. Dadurch haben wir eine Übersicht der Features und den jeweils dazugehörigen Anwender erhalten.

Als Benutzer möchte ich **Produkte verleihen** und **ausleihen** können.

Als Benutzer möchte ich nach **Produkten** suchen und die **Informationen** darüber **angezeigt** bekommen.

Als Benutzer möchte ich, dass ich über **Veränderungen** die mich betreffen **informiert werde**.

Als Benutzer möchte ich **Information über die andere Benutzer erhalten**.

Als Benutzer möchte ich mit anderen Benutzern **in Kontakt treten** können.

Als Benutzer möchte ich andere **Benutzer bewerten** können

Als Benutzer möchte ich **Regelverstöße melden** können.

Als Benutzer möchte ich meinen **Account verwalten**.

Als Betreiber möchte ich, dass der **Benutzer sich authentifizieren**, bevor er die Funktionen der Endanwendung nur nutzen kann.

Als Betreiber möchte ich die Benutzer über ihre **Rechte und Pflichten aufklären**.

Als Betreiber möchte ich die **Benutzer animieren** Produkte zu verleihen.

Als Betreiber möchte ich durch die Endanwendung **Geld verdienen**.

Als Administrator möchte ich **Benutzer und Produkte verwalten** können.

Als Administrator möchte ich die **Endanwendung verwalten** können.



## 4 Konkurrenzanalyse

Nach dem Brainstorming zu unserer Produktidee, schauten wir uns um, ob es bereits andere Anwendungen gibt, die ähnliche Funktionen hatten wie es unser Produkt am Ende haben soll. Dabei gab es ein paar Suchergebnisse, von denen zwei im Nachfolgenden weiter betrachtet und mit unsere Idee verglichen werden sollen.

### 4.1 Applikation

*Letgo* ist eine kostenlose Android und IOS Applikation, mit der es möglich ist gebrauchte Produkte zu kaufen oder verkaufen. Außerdem ist es auch erlaubt ganz neue Produkte zum Verkauf anzubieten. Verleih- oder Ausleihoptionen werden nicht angeboten. Die Produktsuche wird in einer bestimmte Umgebung realisiert. Die Eigentümer dieser Applikation stellen sicher, dass keine Benutzerinformation an dritte weiter gegeben werden.

#### 4.1.1 Vorteile

- Benutzerdatensicherheit
- Systemunabhängigkeit - Man kann wenn man möchte auch die Webseite von *Letgo* nutzen
- Die Ansicht der Webseite ist auch für die Mobilgeräte angepasst

#### 4.1.2 Nachteile

- Hochgeladene Bilder werden nicht automatisch skaliert. Manche Bilder sind zu groß. Man muss je nach Suchergebnis die Seite lang scrollen.

## 4.2 Web-Applikation

*frents.com* ist eine Webplattform, die es ermöglicht gebrauchte Produkte zu verleihen, ausleihen auch zu verkaufen. Die Produktsuche wird bundesweit realisiert. Damit man diese Plattform nutzen kann, muss man nicht unbedingt Gebühren zahlen. Außerdem ist es auch möglich in einer bestimmter Umgebung anzuschauen, was und wie viel aus- oder verliehen wird. Man erstellt auf diese Plattform ein Konto, kann sich aber auch mit seinem Facebook-Konto authentifizieren.

### 4.2.1 Vorteile

- Man muss kein Konto erstellen oder sich authentifizieren, um Produkt zu suchen.
- Man kann auch anschauen, wie aktiv in seiner Umgebung aus- und verliehen wird. Man kann sogar die andere Umgebungen anschauen, um die Aktivität anzuschauen.

### 4.2.2 Nachteile

- Die Mobile Ansicht wurde nicht berücksichtigt. Wenn man die Webseite mit auf seinem Handy anschauen möchte, bekommt man keine optimale Ansicht von der Webseite.

## 4.3 Vergleich mit unserer Idee

### 4.3.1 Vorteile

Was die beiden Konkurrenten im Gegensatz zu und nicht anbieten, ist ein Mechanismus zur Benutzerbewertung. Mit Hilfe dieses Mechanismus kann festgestellt werden, ob der Aus- oder Verleiher eine zuverlässige Person ist.

Ein anderer Mechanismus, der zum Einsatz kommen kann, ist die Produktmeldung. Mit diesem Mechanismus werden illegale Produkte, die angeboten werden, als "kein geeignetes Produkt" gemeldet werden. Wenn ein Produkt gemeldet wird, wird der Administrator informiert. Dieser wird entsprechende Schritte einleiten, damit das gemeldete Produkt aus dem Angebot weggenommen werden kann.

### **4.3.2 Nachteile**

In unseren Applikation muss man sich einloggen und authentifizieren, um die Applikation bedienen zu können. Die Produktsuche kann ohne Authentifizierung nicht erfolgt. Unsere Gesammelten Kundendaten, werden für Big-Data Auswertungen genutzt.

Da unsere Applikation nur für die Apple Mobilgeräte gedacht ist, bieten wir keine Möglichkeit, dass unsere Applikation mit einer Webplattform und einem Android-Gerät bedient werden können.

## 5 Vorgehensmodell

Team Beta hat sich dafür entschieden, dass die Applikation xShare mit dem Vorgehensmodell SCRUM entwickelt wird. Zu dieser Entscheidung kam es, da Scrum die benötigte Flexibilität und Kundennähe bietet, die anderen Modelle nicht haben. Die Flexibilität wird benötigt, da die Applikation ständig in Form von Releases verbessert werden muss. Im ersten Release sind nur die Features implementiert, die für die Applikation wirklich notwendig sind. In jedem weiteren Release werden dann neue Features hinzugefügt und entdeckte Bugs entfernt. So erhält man bereits nach kurzer Entwicklungszeit eine funktionsfähige Applikation, die dann Stück für Stück verbessert wird. Da man nach jedem Sprint ein funktionsfähiges Ergebnis hat, kann man dieses dem Kunden präsentieren. Durch die Überprüfung der Funktionalität durch den Kunden und der daraus resultierenden Rückmeldung, können im folgenden Sprint Anpassungen vorgenommen werden. So ist der Kunde immer auf dem neusten Stand und das Team Beta kann auf die Kundenwünsche eingehen.

### 5.1 Steakholder

**Kunde** Miroslav Juric ist der Kunde von Team Beta. Er bestellte eine Endanwendung, die den Anwendern das Verleihen und Ausleihen verschiedener Produkte ermöglicht.

**Anwender** Als Hauptzielgruppe wurden jüngere Menschen ausgewählt, die sich Produkte, die sie nur selten verwenden, nicht leisten möchten. Vor allem diesen Menschen soll die Applikation ermöglichen, solche Produkte kostenlos für einen bestimmten Zeitraum zu erhalten. Trotzdem kann die Applikation für jeden ab 18 Jahre (Unbeschränkte Geschäftsfähigkeit) verwendet werden.

### 5.2 Product Owner

Als Product-Owner trat Steffen Weinstein in diesem Projekt auf.

## 5.3 Scrum Master

Die Rolle des Scrum-Masters übernahm Christine Krüger.

## 5.4 Entwicklungsteam

Das Entwickler-Team besteht aus Cihan Aktas, Julian Diehlmann und Matthias Reckmeyer.

## 5.5 Vorgehensweise

Es ist festgelegt worden, dass ein Sprint eine Woche lang dauert und er immer Donnerstag Nachmittags, nach dem Meeting mit dem Kunden, beginnt. Vor dem Beginn eines neuen Sprints wird in einem Meeting der kommende Sprint geplant. Aus dem Produktbacklog, das während dem Meeting mit dem Kunden mit neuen Aufgaben gefüllt wurde, werden die Aufgaben, die in diesem Sprint abgearbeitet werden können, ausgewählt. Die ausgewählten Aufgaben werden einem Entwickler zugewiesen und dieser schätzt die benötigte Zeit, die er für die Aufgabe braucht. Nach dem Meeting wird der Sprint gestartet. Dies bedeutet, dass jeder Entwickler seine zugeteilten Aufgaben startet. Der Beginn und die Vollendung einer Aufgabe wird unverzüglich im Scrum eingetragen. So erhält der Scrum-Master ein Burndown-Diagramm, an dem er den Zeitlichen Rahmen übersichtlich im Blick hat. Zusätzlich wird durch regelmäßige Kurzmeetings mit dem Scrum-Master, bei denen jeder kurz vorstellt, wie weit er mit seiner Aufgabe ist und wo es vielleicht zu Problemen kommen könnte, sicher gestellt, dass alle Aufgaben des Sprints pünktlich zum Ende des Sprints vollendet werden. Aufgetretene Probleme wurden in Teamarbeit gelöst.

## 5.6 Software-Lebenszyklus

Softwarelösungen, wie unsere Applikation, folgen einem Lebenszyklus. Bei unserer Applikation xShare benötigen folgende drei Phasen ein erhöhtes Augenmerk.

### 5.6.1 Phase 1: Inbetriebnahme

Der Erfolg von xShare wird über die Größe seiner Community definiert. Erst durch eine große Anzahl an Nutzern kann gewährleistet werden, dass es ein ausreichendes Angebot

von Produkten gibt. Bei der Inbetriebnahme ist aber noch keine Community vorhanden. Da der Aufbau einer solchen Community Zeit benötigt, veröffentlichen wir zuerst eine Beta Version. Diese enthält nur die Features, die die Anwendung wirklich benötigt. So können bereits die ersten Nutzer Erfahrungen mit der App machen. Dadurch ist dann beim ersten Release schon eine Community vorhanden.

### **5.6.2 Phase 2: Betrieb und Wartung**

Um die Applikation möglichst schnell veröffentlichen zu können, werden bei dem ersten Release nicht alle Features implementiert. Die restlichen Features werden stückchenweise in neuen Versionen bereitgestellt. Zusätzlich wird auf die Reaktionen der Anwender reagiert und dementsprechende Verbesserungen und Bug-behebungen vorgenommen. So entsteht eine kundennahe Applikation.

Neben der Applikation muss auch ein Server mit einer Datenbank betrieben werden. Zum einen muss regelmäßig die Kapazität des Servers überprüft werden und diese bei falscher Größe entsprechend angepasst werden. Zum anderen müssen die Daten redundant gespeichert werden, zum Beispiel in Form eines Backups, damit es zu keinem Verlust der Daten kommt.

### **5.6.3 Phase 3: Außerbetriebnahme**

Sollte xShare aus unterschiedlichen Gründen irgendwann außer Betrieb genommen werden, müssen dabei auf einige Dinge geachtet werden: Zum einen muss sichergestellt werden, dass keine Produkte mehr ausgeliehen sind. Dies wird erreicht, indem man den Anwendern eine Nachricht in der Applikation zukommen lässt, die die nächsten Schritte und was es zu beachten gibt erklärt. Außerdem wird eine letzte neue Version veröffentlicht, bei der die Ausleihfunktion deaktiviert wurde. Danach kann die App noch bis zum Ablauf der letzten Ausleihfrist verwendet werden. Dann wird der Server abgestellt und die Datenbank archiviert.

# 6 Features

Basierend auf unserer Produkt-Userstory, wurden verschiedene Features erarbeitet. Die Features wurden in weitere Use-Cases unterteilt, kurz beschrieben und mit einer Priorität versehen.

## 6.1 Produkt verleihen

### 6.1.1 Produkt anbieten

**Priorität** must-be

**Beschreibung** Produkte zum Verleih freigeben

**User-Story** Als Benutzer kann ich Produkte zum Verleih anbieten. Ich kann meine Produkte beschreiben und eine Ausleihdauer bestimmen.

### 6.1.2 Kategorie angeben

**Priorität** expected

**Beschreibung** Produkt in eine Kategorie (z.B. Kleidung) einordnen

**User-Story** Als Benutzer kann ich aus einer vorgegebenen Liste von Kategorien eine Kategorie auswählen, die zu meinem Produkt, das verliehen wird, passt.

### 6.1.3 Mängelbeschreibung eingeben

**Priorität** nice-to-have

**Beschreibung** Mängel, die ein Produkt hat, können in einem extra Feld eingegeben werden.

**User-Story** Als Benutzer, dem das Produkt gehört, kann ich die Mängel meines Produktes eingeben.

## 6.1.4 Beschreibung eingeben

**Priorität** must-be

**Beschreibung** Eine Beschreibung über das Produkt kann eingegeben werden.

**User-Story** Als Benutzer möchte ich eine Beschreibung über das Produkt eingeben und lesen können.

## 6.1.5 Kautio eingeben

**Priorität** expected

**Beschreibung** Eine Kautio, die für ein Produkt verlangt wird, kann in einem extra Feld eingegeben werden.

**User-Story** Als Benutzer kann ich für mein Produkt eine Kautio, in von mir bestimmter Höhe, eingeben.

## 6.1.6 Standort eingeben

**Priorität** must-be

**Beschreibung** Standort, an dem das Produkt abgeholt oder zurück gebracht werden kann, kann eingegeben werden.

**User-Story** Als Benutzer kann ich den Standort eingeben, an dem mein Produkt abgeholt bzw. zurück gebracht werden kann.

## 6.1.7 Bilder hinzufügen

**Priorität** nice-to-have

**Beschreibung** Zu jedem auszuleihenden Produkt ein Bild davon hinzufügen.

**User-Story** Als Benutzer kann ich jedem Produkt, das ich verleihe, ein oder mehrere Bilder hinzufügen, um den anderen Benutzern eine bessere Vorstellung des Produktes zu ermöglichen.



## 6.1.8 Produkt entfernen

**Priorität** expected

**Beschreibung** Produkte die man nicht mehr verleihen möchte entfernen

**User-Story** Als Benutzer möchte ich Produkte, die ich zum Verleih anbiete, auch wieder entfernen können.

## 6.2 Produkt suchen

### 6.2.1 Produkte suchen

**Priorität** must-be

**Beschreibung** Nach einem bestimmten Produkt kann gesucht werden

**User-Story** Als Benutzer kann ich den Name des Produktes, das ich ausleihen möchte, angeben und dann danach suchen. Produkte mit ähnlichen Namen werden darauf hin angezeigt.

### 6.2.2 zur Merkliste hinzufügen

**Priorität** nice-to-have

**Beschreibung** Produkte markieren, um sie später wieder zu finden.

**User-Story** Als Benutzer möchte ich mir in bestimmten Situationen Produkte merken, um sie später einfach wieder zu finden.

### 6.2.3 Suchradius ändern

**Priorität** expected

**Beschreibung** Suche nach Artikeln in einem voreingestelltem oder geänderten Umkreis

**User-Story** Als Benutzer möchte ich bei der Suche nur Produkte in einem bestimmten Radius von meinem Standort aus angezeigt bekommen. Diesen Suchradius möchte ich ändern können.

## 6.2.4 Standort bestimmen

**Priorität** nice-to-have

**Beschreibung** Der Standort des Ausleiher wird durch Geo Location lokalisiert.

**User-Story** Als ausleihender Benutzer möchte ich, dass mein Standort über Geo Location bestimmt wird, um Produkte in der Nähe meines Standortes zu finden

## 6.3 Informationen angezeigt bekommen

### 6.3.1 Kurzinformationen anzeigen

**Priorität** must-be

**Beschreibung** Die Kurzinformationen der Produkte anzeigen.

**User-Story** Als Benutzer möchte ich eine Übersicht über die Kurzinformationen der Produkte angezeigt bekommen.

### 6.3.2 Produktdetails anzeigen

**Priorität** must-be

**Beschreibung** Alle Details des Produktes anzeigen

**User-Story** Als Benutzer möchte ich alle Details eines Produktes angezeigt bekommen.

### 6.3.3 Status anzeigen

**Priorität** expected

**Beschreibung** Anzeige ob das gewünschte Produkt verfügbar ist oder nicht

**User-Story** Als ausleihender Benutzer möchte ich eine Statusanzeige, um zu erkennen, ob das gewünschte Produkt verfügbar ist oder nicht. Der Status wird automatisch beim Verleih und nach dem Ablauf der Ausleihdauer aktualisiert.

### 6.3.4 Restzeit anzeigen

**Priorität** nice-to-have

**Beschreibung** Anzeige der Restzeit bis das Produkt wieder verfügbar ist

**User-Story** Als ausleihender Benutzer möchte ich eine Restzeitanzeige um zu wissen wann das Produkt wieder zum Verleih verfügbar ist.

## 6.4 Produkt ausleihen

### 6.4.1 Anfrage erstellen

**Priorität** must-be

**Beschreibung** Zu verfügbaren Produkten, die ich ausleihen möchte kann eine Anfrage erstellt werden.

**User-Story** Als authentifizierter Benutzer Anfrage erstellen, um es auszuleihen, nachdem ich es in der Suche gefunden habe.

### 6.4.2 Produkt bewerten

**Priorität** must-be

**Beschreibung** Ein ausgeliehens Produkt kann vom Ausleiher bewertet werden.

**User-Story** Als Ausleiher möchte ich das ausgeliehene Produkt bewerten können, sowohl mit einem Punktesystem, als auch mit einer Beschreibung, wenn ich das möchte.

## 6.5 Über Veränderungen informiert werden

### 6.5.1 Benachrichtigung über Statusänderungen von Produkten

**Priorität** expected

**Beschreibung** Der Benutzer bekommt Benachrichtigungen über die Statusänderung von Produkten auf seiner Merkliste.

**User-Story** Als Benutzer möchte ich benachrichtigt werden, wenn die Produkte, die ich auf meiner Merkliste habe, wieder verfügbar sind.

### 6.5.2 Anfragebestätigung

**Priorität** must-be

**Beschreibung** Der ausleihende Benutzer wird über das Ergebnis seiner Ausleihanfrage informiert

**User-Story** Als ausleihender Benutzer möchte ich benachrichtigt werden, wenn ein verleihender Benutzer meine Ausleihanfrage annimmt oder ablehnt. Wird die Anfrage angenommen kommt ein Vertrag zustande.

### 6.5.3 Ermahnungsbenachrichtigung

**Priorität** expected

**Beschreibung** Bei Fehlverhalten von Benutzern wird dieser Benutzer ermahnt und darüber informiert.

**User-Story** Als Administrator möchte ich Benutzer über ihre Ermahnung bei einem Fehlverhalten benachrichtigen.

### 6.5.4 Aktionspunkte anzeigen

**Priorität** expected

**Beschreibung** Aktionspunkte sind für den Benutzer sichtbar

**User-Story** Als Benutzer möchte ich meine Aktionspunkte angezeigt bekommen.

## 6.6 Informationen über Benutzer erhalten

### 6.6.1 Benutzer suchen

**Priorität** nice-to-have

**Beschreibung** Nach einem Benutzer kann gesucht werden

**User-Story** Als Benutzer kann ich den Namen eines anderen Benutzers angeben und anschließend danach suchen. Benutzer mit ähnlichen Namen werden angezeigt. Bei Auswahl eines Benutzer, wird das Profil angezeigt.

### 6.6.2 Benutzerprofil anzeigen

**Priorität** expected

**Beschreibung** Benutzerprofil eines Benutzer anzeigen

**User-Story** Als Benutzer möchte ich Informationen über einen anderen Benutzer angezeigt bekommen. Dafür kann ich mir das Benutzerprofil des Benutzers anschauen.

## 6.7 In Kontakt treten

### 6.7.1 Benachrichtigungen über erhaltene Nachrichten

**Priorität** expected

**Beschreibung** Bei Erhalt einer Nachricht wird der Benutzer benachrichtigt.

**User-Story** Als Benutzer möchte ich Benachrichtigung werden, wenn ein Benutzer mir eine neue Nachricht geschrieben hat.

### 6.7.2 Chatsystem

**Priorität** expected

**Beschreibung** Nachrichtenaustausch zwischen Verleihenden und Ausleihenden

**User-Story** Als verleihender und ausleihender Benutzer möchten wir Nachrichten untereinander austauschen um uns absprechen zu können.

## 6.8 Benutzer bewerten

### 6.8.1 authentifizierten Benutzer bewerten

**Priorität** expected

**Beschreibung** Bewerten von Nutzern im Hinblick auf Verlässlichkeit und Ablauf einer Transaktion

**User-Story** Als Benutzer möchte ich andere Benutzer bezüglich der Verlässlichkeit und dem Ablauf der Transaktion bewerten können.

## 6.9 Regelverstöße melden

### 6.9.1 Benutzer melden

**Priorität** must-be

**Beschreibung** Benutzer melden die die Verhaltensrichtlinien nicht eingehalten zu haben

**User-Story** Als Benutzer möchte ich Benutzer melden können, um im Falle eines Verstoßes gegen den Verhaltenskodex diesen Nutzer bestrafen zu lassen, damit alle ein positives Erlebnis mit der Anwendung haben

### 6.9.2 Produkt melden

**Priorität** must-be

**Beschreibung** Unangemessene Produkte an die Administratoren zur Überprüfung melden

**User-Story** Als Benutzer möchte ich Produkte melden können, die illegal oder sinnlos sind.

## 6.10 Account verwalten

### 6.10.1 Profil verwalten

**Priorität** expected

**Beschreibung** Daten des Benutzers können verwaltet werden

**User-Story** Als Benutzer möchte ich meine Daten in meinem Profil verwalten können.

### 6.10.2 allgemeine Einstellungen

**Priorität** expected

**Beschreibung** Allgemeine Einstellungen der Endanwendung können vorgenommen werden

**User-Story** Als Benutzer möchte ich Einstellungen an der Endanwendung vornehmen können.

## 6.11 Benutzer authentifizieren

### 6.11.1 Erstellen eines Accounts

**Priorität** must-be

**Beschreibung** Persönliche Informationen eines Benutzers angeben

**User-Story** Als Betreiber will ich, dass die Benutzer einen Account erstellen und dort ihre persönlichen Daten angeben, die für die Endanwendung notwendig sind.

### 6.11.2 Authentifizierung über Facebook

**Priorität** nice-to-have

**Beschreibung** Die Endanwendung ermöglicht die Authentifizierung mit einem Facebook-Account

**User-Story** Als Betreiber möchte ich die Möglichkeit schaffen, dass sich die Benutzer über ihren Facebook Account authentifizieren können.

### 6.11.3 Authentifizierung über Google+

**Priorität** expected

**Beschreibung** Die Endanwendung ermöglicht die Authentifizierung mit einem Google-Account

**User-Story** Als Betreiber möchte ich die Möglichkeit schaffen, dass sich die Benutzer über seinen Google+ Account authentifizieren können.

### 6.11.4 Authentifizierung über Email

**Priorität** must-be

**Beschreibung** Die Endanwendung ermöglicht eine Authentifizierung über die E-Mail Adresse

**User-Story** Als Betreiber möchte ich die Möglichkeit schaffen, dass sich die Benutzer über ihre E-Mail Adresse authentifizieren können.

## 6.12 Über Rechte und Pflichten aufklären

### 6.12.1 AGB

**Priorität** must-be

**Beschreibung** Zeigt die AGB

**User-Story** Als Betreiber möchte ich, dass die Benutzer die AGBs einsehen können.

### 6.12.2 Haftungsausschluss

**Priorität** must-be

**Beschreibung** Rechtliche Absicherung des Entwicklerteams und der Betreiber im Haftungsfall

**User-Story** Als Betreiber möchte ich einen Haftungsausschluss, um mich gegen rechtliche Klagen von Benutzern im Bezug auf die Anwendung abzusichern.



### 6.12.3 Verhaltenscodex

**Priorität** must-be

**Beschreibung** Grundlegende Richtlinien, an die sich die Nutzer zu halten haben um für alle Mitglieder ein positives Nutzungserlebnis zu garantieren

**User-Story** Als Betreiber möchte ich einen Verhaltenscodex für die Benutzer, um diesen eine Positive Benutzungserfahrung mit der Anwendung zu bieten.

## 6.13 Benutzer animieren

### 6.13.1 Ausleihverhältnis

**Priorität** must-be

**Beschreibung** Das Ausleihverhältnis soll gewährleisten, dass jeder etwas der Community zum ausleihen anbietet, und nicht nur von andere etwas ausleiht.

**User-Story** Als Benutzer möchte ich, dass alle etwas ausleihen und verleihen, damit jeder Benutzer davon profitieren kann.

## 6.14 Endanwendung benutzerfreundlich gestalten

### 6.14.1 Mehrsprachige Oberfläche

**Priorität** expected

**Beschreibung** Dem Benutzer soll es ermöglicht werden, die Oberfläche der Anwendung in unterschiedlichen Sprachen, je nach Vorliebe an zu sehen.

**User-Story** Als Benutzer möchte ich, dass ich die Sprache in der die Oberfläche angezeigt wird frei wählbar ist.

## 6.15 Geld verdienen

### 6.15.1 Premiumaccount

**Priorität** must-be

**Beschreibung**

**User-Story** Als Benutzer möchte ich, dass wenn ich ein Premiumaccount habe, keine Werbung mehr sehe und von allen Features der Endanwendung profitieren kann, ohne Einschränkungen.

### 6.15.2 Big-Data-Auswertung

**Priorität** nice-to-have

**Beschreibung**

**User-Story** Als Betreiber möchte ich Big-Data an andere Firmen verkaufen, um so Geld zu verdienen.

## 6.16 Benutzer verwalten

### 6.16.1 Account sperren

**Priorität** expected

**Beschreibung** Nach einer bestimmten Anzahl an Beschwerden gegenüber einem Benutzer, kann dieser für einen bestimmten Zeitraum gesperrt werden

**User-Story** Als Administrator möchte ich bei häufigen Fehlverhalten eines Benutzers, dessen Account temporär sperren.

### 6.16.2 Account löschen

**Priorität** nice-to-have

**Beschreibung** Bei zu häufigem Regelverstoß kann ein Benutzer gelöscht werden.

**User-Story** Als Administrator möchte ich einen Account eines Benutzers, der sich trotz mehrerer Sperren sein Verhalten nicht bessert, löschen können.

## 6.17 Produkt verwalten

### 6.17.1 Produkt sperren

**Priorität** must-be

**Beschreibung** Nach einer bestimmten Anzahl an Beschwerden gegenüber einem Produkt, kann dieses Produkt gesperrt werden.

**User-Story** Nicht legitime Produkte kann ein Administrator entfernen.

## 6.17.2 Produkt entfernen

**Priorität** must-be

**Beschreibung** Als Administrator möchte ich nicht legitime Produkte, die von Benutzern gemeldet wurden, sperren. Der Benutzer der das gesperrte Produkt verleiht, wird aufgefordert das Produkt zu löschen.

**User-Story** Sollte ein Benutzer trotz Aufforderung das Produkt nicht löschen, kann der Administrator das Produkt entfernen.

## 6.18 Endanwendung verwalten

### 6.18.1 Updates einspielen

**Priorität** must-be

**Beschreibung** Updates können eingespielt werden.

**User-Story** Als Betreiber möchte ich die Endanwendung in Form von Updates weiterentwickeln können.

## 7 Use-Cases

In diesem Kapitel werden alle Funktionen der Applikation xShare vorgestellt. Damit die Darstellung effizient und einfach zu verstehen ist, werden Use-Case Diagramme verwendet. Mit Hilfe dieser Diagrammen werden zuerst die unterschiedlichen Benutzergruppen und deren Abhängigkeiten angezeigt. Danach werden alle nötigen Use-Cases veranschaulicht.

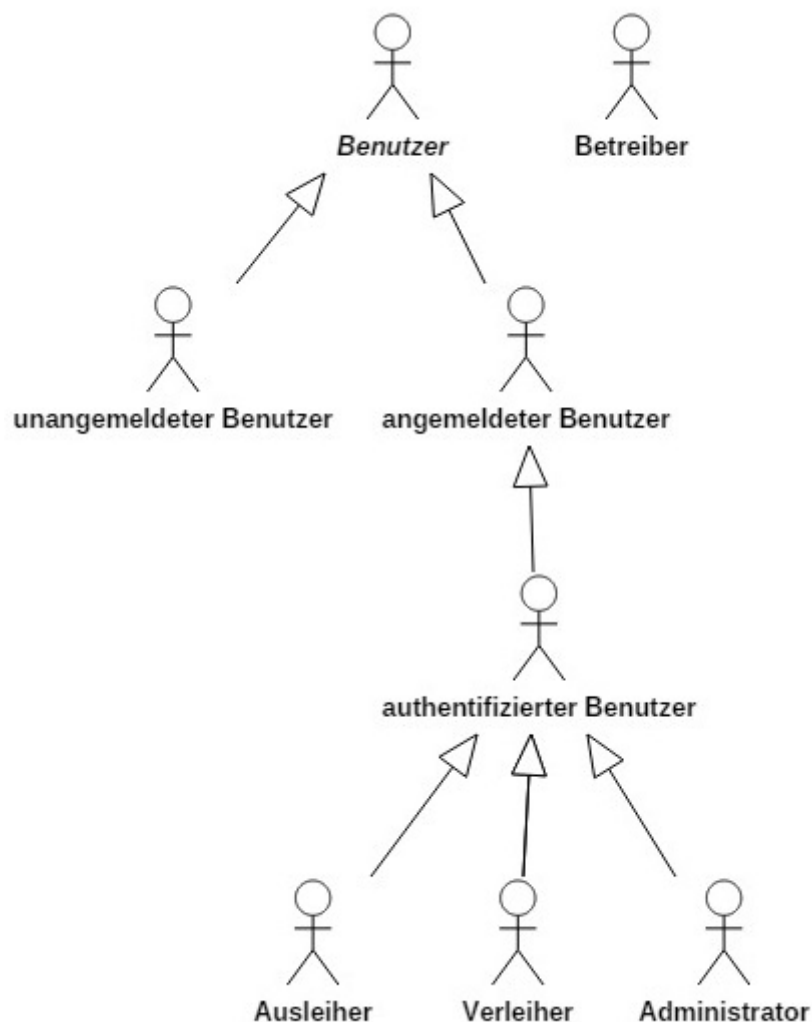


Abbildung 7.1: Use-Cases der Akteure

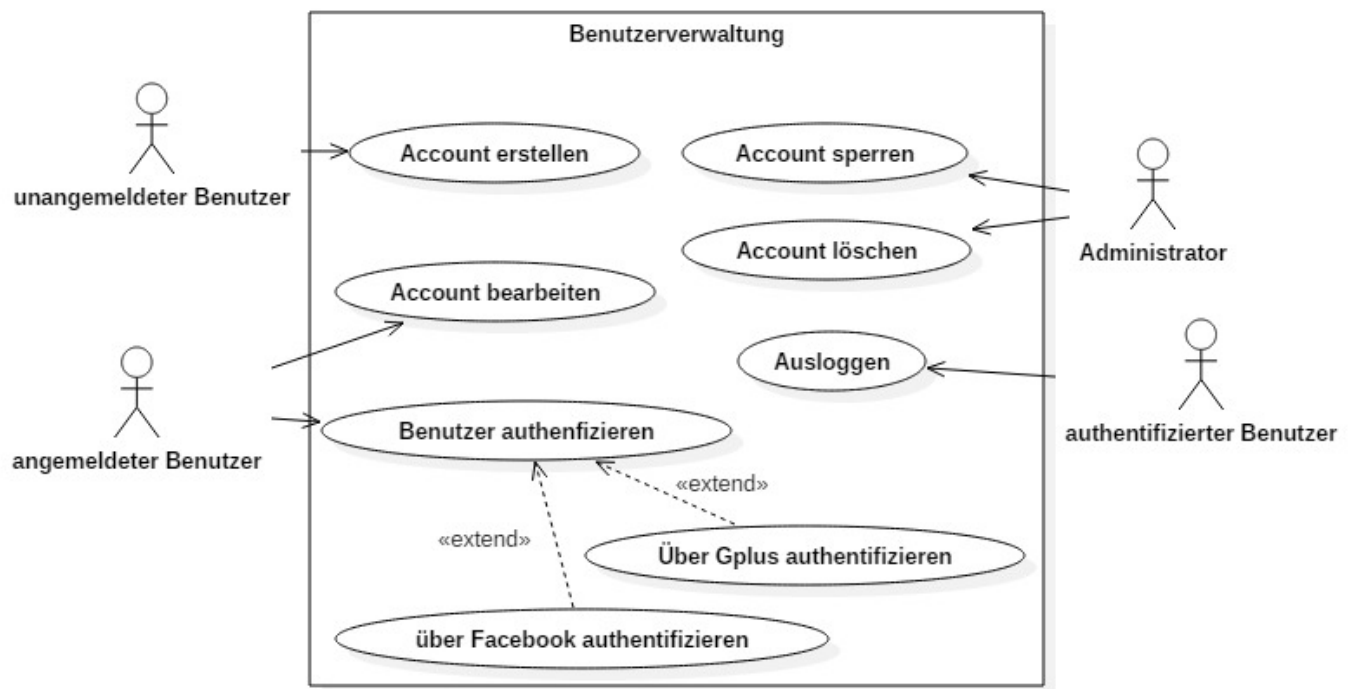


Abbildung 7.2: Use-Cases der Benutzerverwaltung

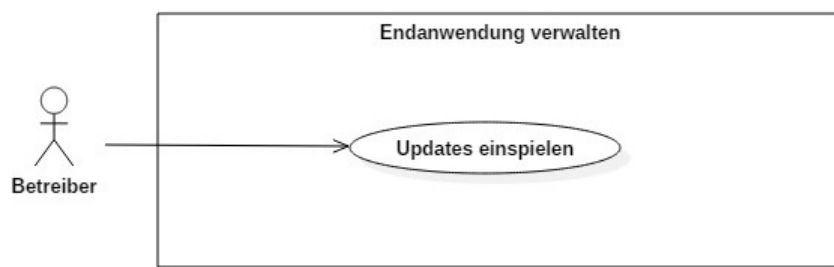


Abbildung 7.3: Use-Cases der Endanwendung verwalten

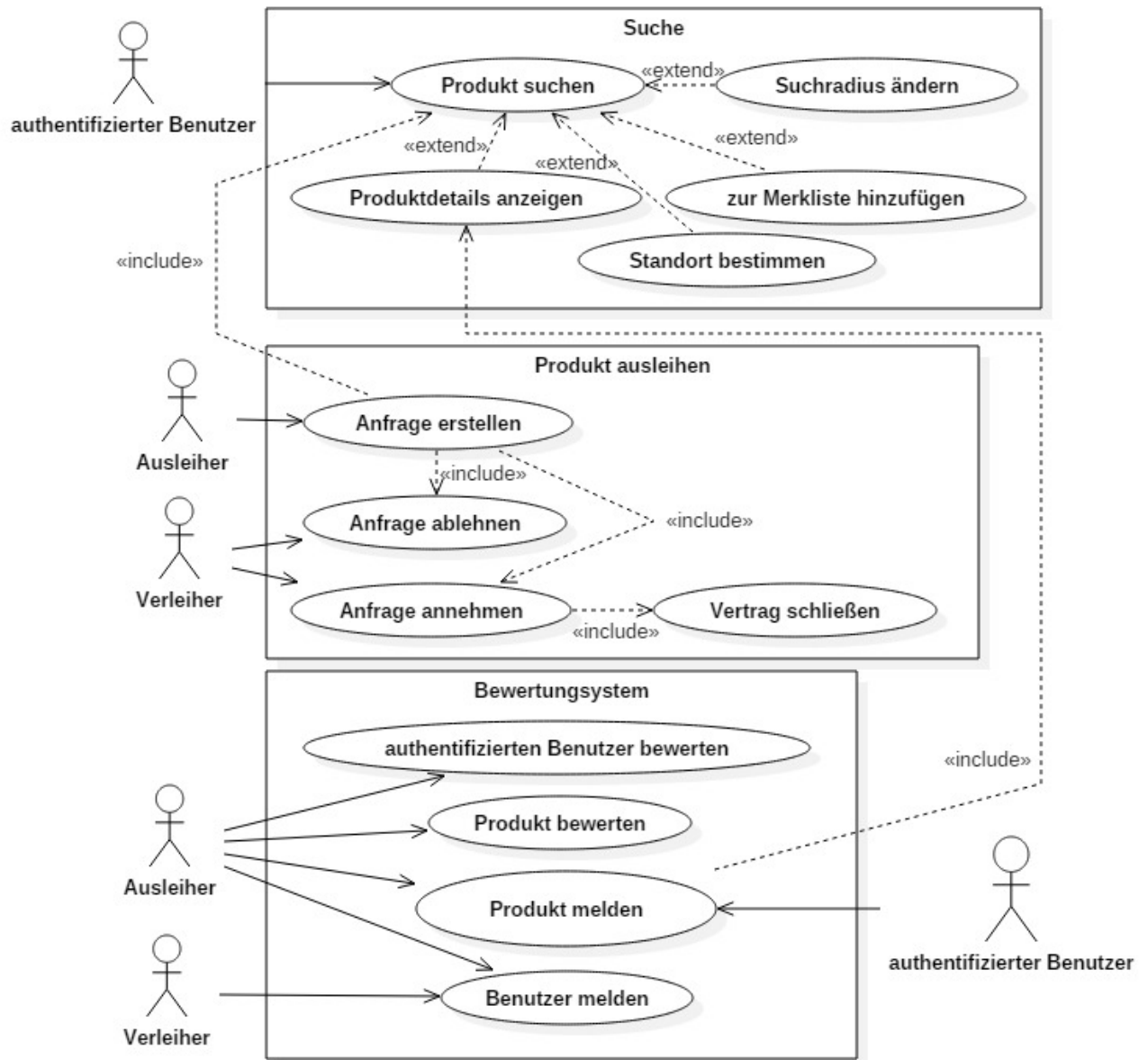


Abbildung 7.4: Use-Cases der Suche Ausleih Bewertung

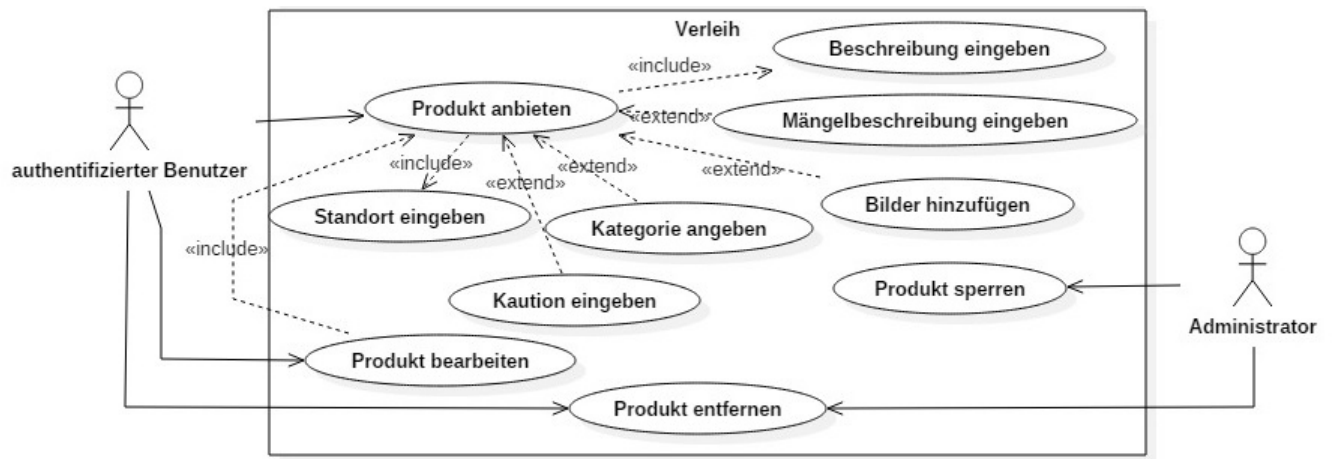


Abbildung 7.5: Use-Cases der Verleih



## 8 Use-Case-Beschreibungen

Im folgendem Kapitel werden einige wichtige, im vorangegangenen Kapitel beschriebenen Use-Cases, in Tabellenform detaillierter beschrieben. Hierbei liegt der Fokus auf den einzelnen Ablaufschritten, sowie Vor- und Nachbedingungen eines Use-Cases.

## 8.1 Produkt suchen

<b>UseCase</b> Produkt suchen		<b>UC ID</b> 01
<b>Akteure</b>	Authentifizierter Benutzer	
<b>Kurzbeschreibung</b>	Der Benutzer sucht ein Produkt.	
<b>Vorbedingungen</b>	Der Standort von dem Benutzer wird ermittelt, oder vom Benutzer angegeben.	
<b>Nachbedingung bei Erfolg</b>		<b>Nachbedingung bei Misserfolg</b>
Die Suchergebnisse werden dem Benutzer angezeigt.		Es werden keine Suchergebnisse angezeigt. Der Benutzer wird darüber informiert und darum gebeten Suchparameter anzupassen.
<b>Ablaufschritte Standard:</b>		<b>Ablaufschritte Alternativ:</b>
Der Benutzer gibt einen Suchbegriff ein. Im eingegebenen Suchradius wird nach den Produkten gesucht. Die gefundenen Produkte werden angezeigt.		keine
<b>Priorität</b>	Must-be	
<b>Spezialanforderung</b>	keine	
<b>Bemerkung</b>	keine	

Tabelle 8.1: Use-Case-Beschreibung: Produkt suchen

## 8.2 Benutzer löschen

<b>UseCase</b> Benutzer löschen		<b>UC ID</b> 02
<b>Akteure</b>	Administrator	
<b>Kurzbeschreibung</b>	Der Administrator kann einen Benutzer löschen.	
<b>Vorbedingungen</b>	<p>Der Benutzer wurde aufgrund von Fehlverhalten zuvor gesperrt.</p> <p>Der Benutzer hat sein Verhalten trotz Ermahnung nicht geändert.</p> <p>Sicherungszustand wurde erstellt.</p> <p>Es wurde sichergestellt, dass der Benutzer keine Produkte mehr ausgeliehen hat und ausleihen darf.</p>	
<b>Nachbedingung bei Erfolg</b>		<b>Nachbedingung bei Misserfolg</b>
<p>Account existiert nicht mehr.</p> <p>Alle von dem Benutzer erstellten Produkte existieren nicht mehr.</p> <p>Die verwendete Email-Adresse kann nicht erneut zur Erstellung eines Accounts verwendet werden.</p>		<p>Der Sicherungszustand wird wiederhergestellt.</p>
<b>Ablaufschritte Standard:</b>		<b>Ablaufschritte Alternativ:</b>
<p>Der Account wird ausgewählt</p> <p>Alle zu dem Account gehörigen Objekte werden gelöscht.</p> <p>Der Account wird gelöscht.</p>		keine
<b>Priorität</b>	Must-be	
<b>Spezialanforderung</b>	Der Administrator besitzt alle nötigen Rechte.	
<b>Bemerkung</b>	keine	

Tabelle 8.2: Use-Case-Beschreibung: Benutzer löschen

## 8.3 Benutzer authentifizieren

<b>UseCase</b> Benutzer authentifizieren		<b>UC ID</b> 03
<b>Akteure</b>	Administrator	
<b>Kurzbeschreibung</b>	Der Benutzer authentifiziert sich mit seinen Zugangsdaten.	
<b>Vorbedingungen</b>	Der Benutzer hat bereits einen Account erstellt.	
<b>Nachbedingung bei Erfolg</b>		<b>Nachbedingung bei Misserfolg</b>
Der Benutzer hat sich authentifiziert und kann die Endanwendung und alle Funktionen nutzen.		Bei der Eingabe falscher Authentifizierungsdaten, wird der Benutzer darüber informiert und darum gebeten diese erneut einzugeben.  Der Benutzer wird nicht authentifiziert.
<b>Ablaufschritte Standard:</b>		<b>Ablaufschritte Alternativ:</b>
Der Benutzer gibt seine Authentifizierungsdaten ein.  Diese werden geprüft.  Bei Übereinstimmung wird der Benutzer authentifiziert		Nach der ersten Anwendung, wird dem Benutzer die Wahl geboten, dass seine Anmeldung gespeichert wird, und er somit bei erneutem Aufruf der Endanwendung automatisch authentifiziert wird.  Sollte der Benutzer seine Zugangsdaten vergessen haben, kann er eine Accountwiederherstellung vornehmen lassen.
<b>Priorität</b>	Must-be	
<b>Spezialanforderung</b>	keine	
<b>Bemerkung</b>	keine	

Tabelle 8.3: Use-Case-Beschreibung: Benutzer authentifizieren

## 8.4 Produkt anbieten

<b>UseCase</b> Produkt anbieten		<b>UC ID</b> 04
<b>Akteure</b>	Authentifizierter Benutzer	
<b>Kurzbeschreibung</b>	Der Benutzer bietet ein Produkt an.  Der Benutzer gibt die nötigen Daten an.	
<b>Vorbedingungen</b>	Der Benutzer ist authentifiziert.	
<b>Nachbedingung bei Erfolg</b>		<b>Nachbedingung bei Misserfolg</b>
Das Produkt kann von anderen Benutzern gefunden werden.  Es können Anfragen zu dem Produkt gestellt werden.		Es wird kein Eintrag zu dem Produkt erstellt.  Der Benutzer wird darüber informiert und darum gebeten die Daten erneut einzugeben.
<b>Ablaufschritte Standard:</b>		<b>Ablaufschritte Alternativ:</b>
Der Benutzer gibt Produktname, Beschreibung, Höhe der Kautions, null oder mehrere Bilder, Standort und eine Kategorie an.		keine
<b>Priorität</b>	Must-be	
<b>Spezialanforderung</b>	keine	
<b>Bemerkung</b>	keine	

Tabelle 8.4: Use-Case-Beschreibung: Produkt anbieten

## 8.5 Anfrage erstellen

<b>UseCase</b> Anfrage erstellen		<b>UC ID</b> 05
<b>Akteure</b>	Ausleiher	
<b>Kurzbeschreibung</b>	Der Benutzer erstellt eine Anfrage zu einem Produkt, das er ausleihen möchte.	
<b>Vorbedingungen</b>	Er hat das Produkt bei der Suche gefunden und ausgewählt. Das Produkt ist verfügbar.	
<b>Nachbedingung bei Erfolg</b>		<b>Nachbedingung bei Misserfolg</b>
Es wird eine Anfrage an den Verleiher gesendet.		Es wird keine Anfrage an den Verleiher gesendet.  Der Benutzer wird aufgefordert die Anfrage erneut abzusenden.
<b>Ablaufschritte Standard:</b>		<b>Ablaufschritte Alternativ:</b>
Der Benutzer sendet eine Anfrage zu dem ausgewählten Produkte ab.  Der Verleiher wird über die Anfrage informiert.		keine
<b>Priorität</b>	Must-be	
<b>Spezialanforderung</b>	keine	
<b>Bemerkung</b>	keine	

Tabelle 8.5: Use-Case-Beschreibung: Anfrage erstellen

## 8.6 Produkt bewerten

<b>UseCase</b> Produkt bewerten		<b>UC ID</b> 06
<b>Akteure</b>	Ausleiher	
<b>Kurzbeschreibung</b>	Der Benutzer kann nachdem er das Produkt ausgeliehen hat eine Bewertung über diese abgeben.	
<b>Vorbedingungen</b>	Er muss das Produkt ausgeliehen haben.	
<b>Nachbedingung bei Erfolg</b>		<b>Nachbedingung bei Misserfolg</b>
Die Bewertung wird bei der Produktbeschreibung angezeigt.		Es wird kein Bewertungs-Eintrag erstellt.  Der Benutzer wird darüber informiert und aufgefordert die Bewertung erneut einzugeben.
<b>Ablaufschritte Standard:</b>		<b>Ablaufschritte Alternativ:</b>
Der Benutzer wählt das Produkt, welches bewerten möchte aus.  Er gibt als Bewertung eine Punktzahl an.  Wenn er möchte kann er auch eine Bemerkung angeben und speichert alles.		keine
<b>Priorität</b>	Must-be	
<b>Spezialanforderung</b>	keine	
<b>Bemerkung</b>	keine	

Tabelle 8.6: Use-Case-Beschreibung: Produkt bewerten

## 9 Klassendiagramme

Mit Hilfe der Klassen-Diagramme werden in diesem Kapitel, für die zukünftige Implementierung, die benötigten Klassen vorgestellt. Dabei werden die wichtige Informationen z.B. die Attributen und die Methoden der Klassen angezeigt. Außerdem werden die Abhängigkeiten zwischen der vorgestellten Klassen verdeutlicht.

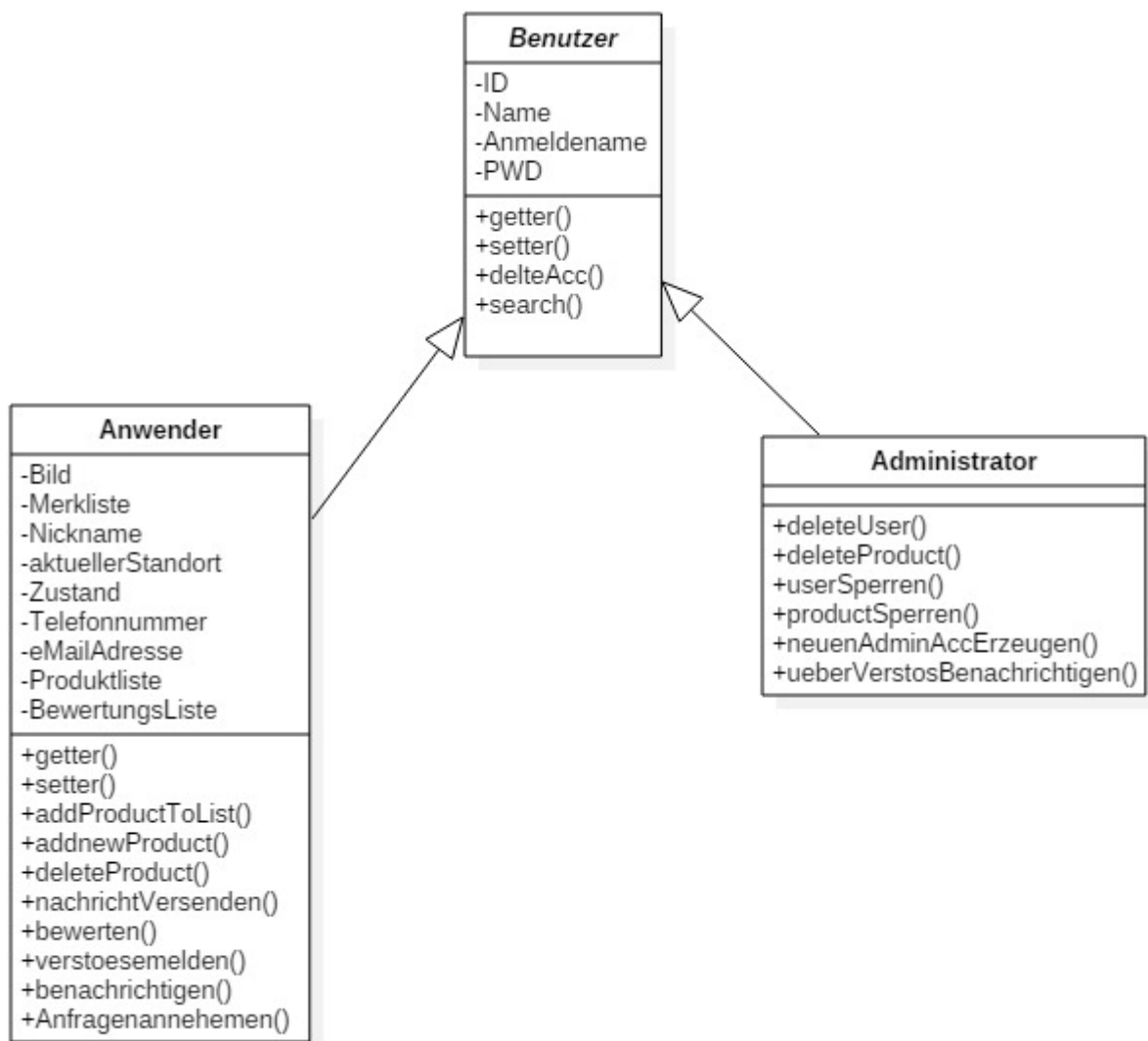


Abbildung 9.1: Klassendiagramme der Akteure



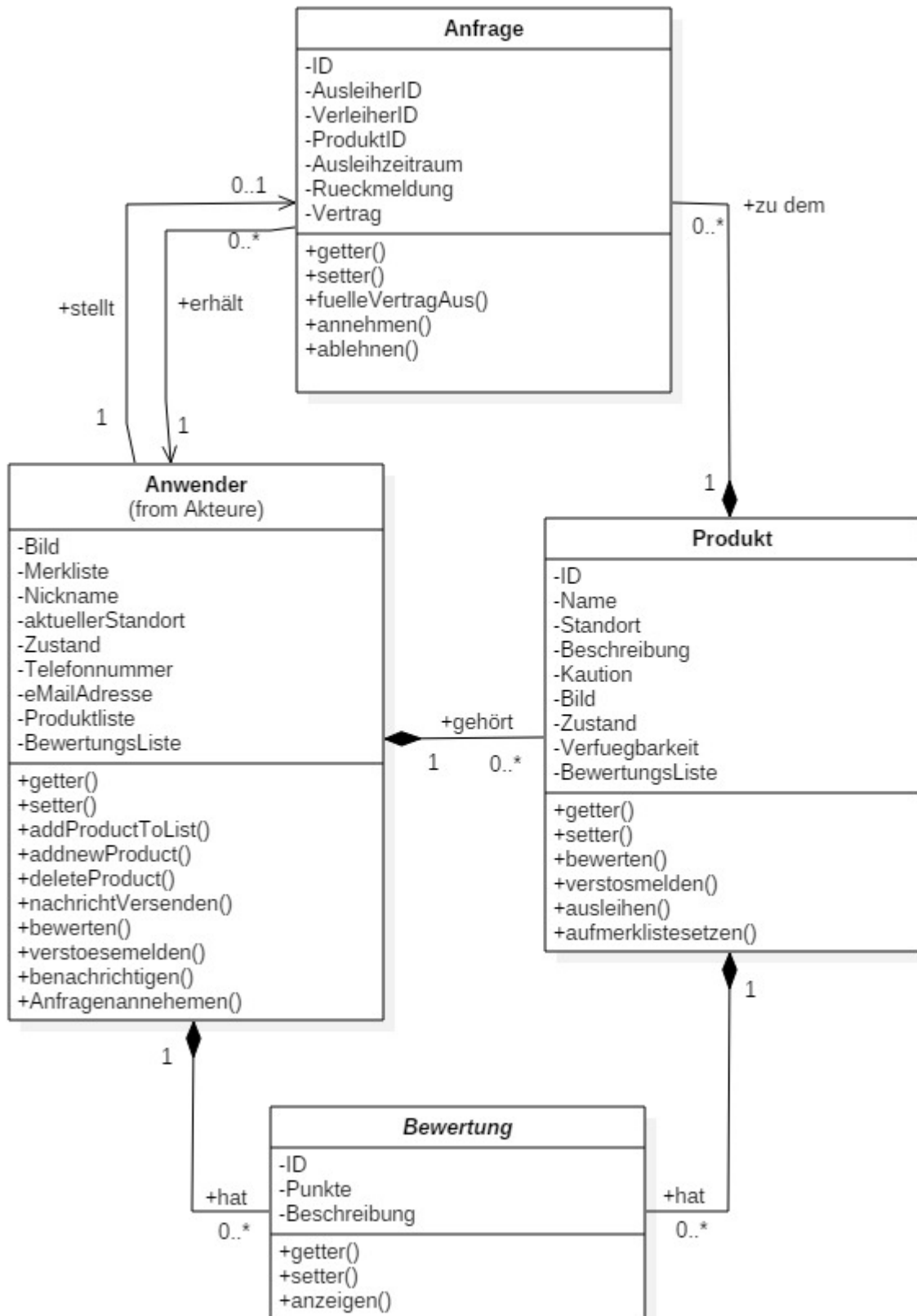


Abbildung 9.2: Klassendiagramme der Klassen

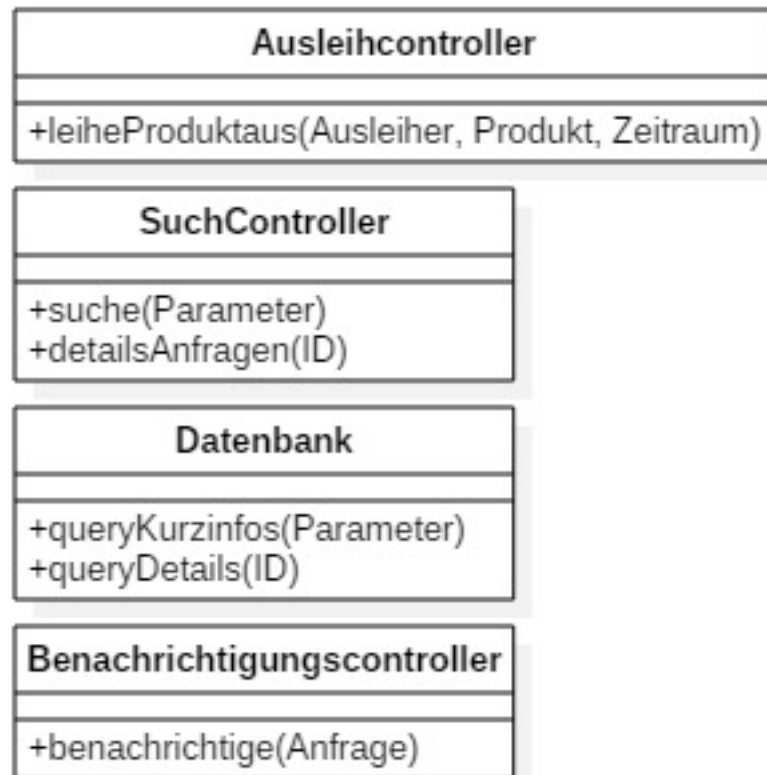


Abbildung 9.3: Klassendiagramme der Controller

# 10 Sequenzdiagramme

Im Folgenden Kapitel sind die Sequenzdiagramme der Suche und der Ausleihe zu sehen. Sie stellen die Interaktionen der verschiedenen Klassen grafisch dar.

Da ein Sequenzdiagramm immer nur einen Weg des Entscheidungsbaumes darstellt, wurden für die Ausleihe zwei Diagramme erstellt. Zum einen für den Fall, dass ein Benutzer die Abfrage annimmt und zum anderen, dass er sie ablehnt.

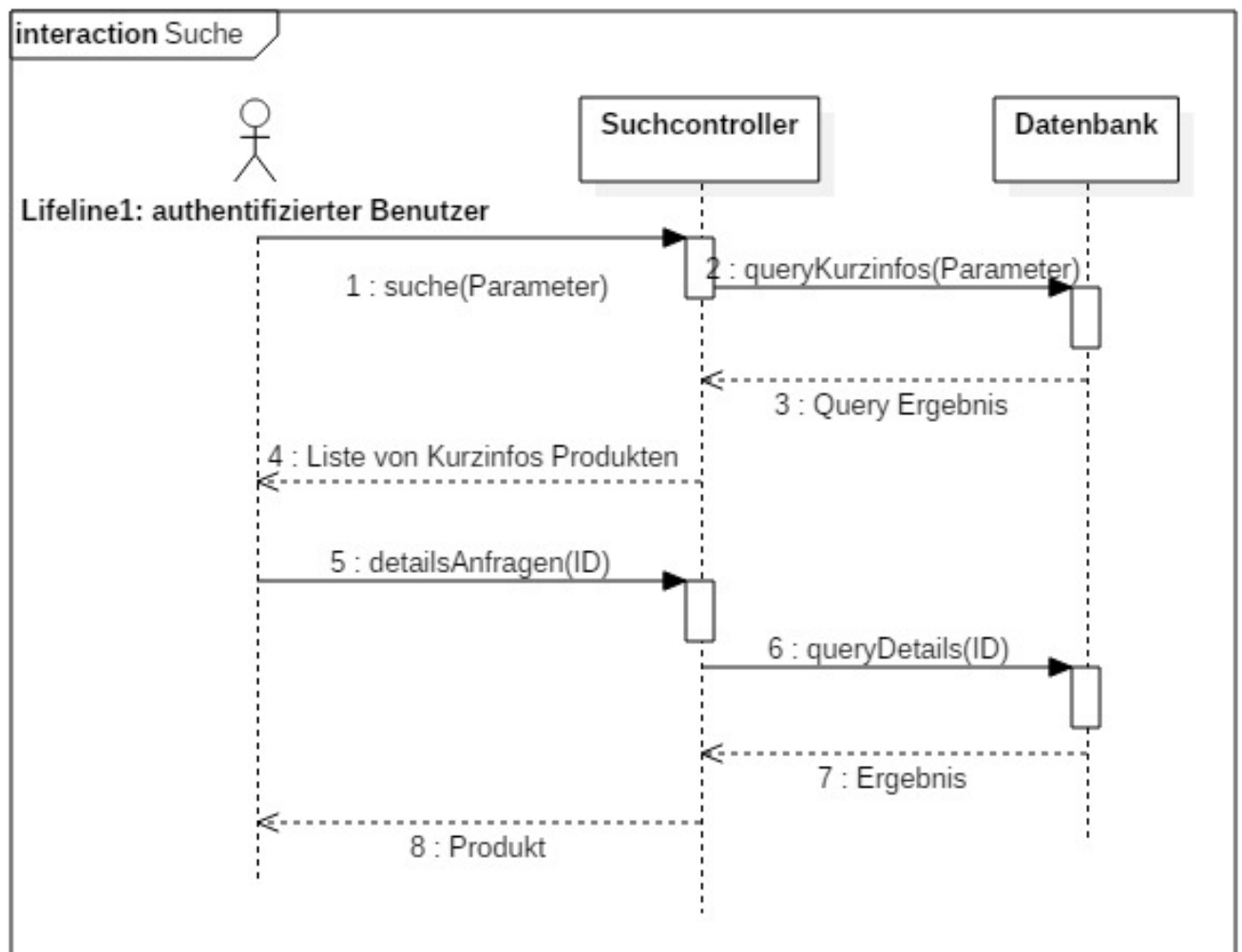


Abbildung 10.1: Sequenzdiagramm der Suche

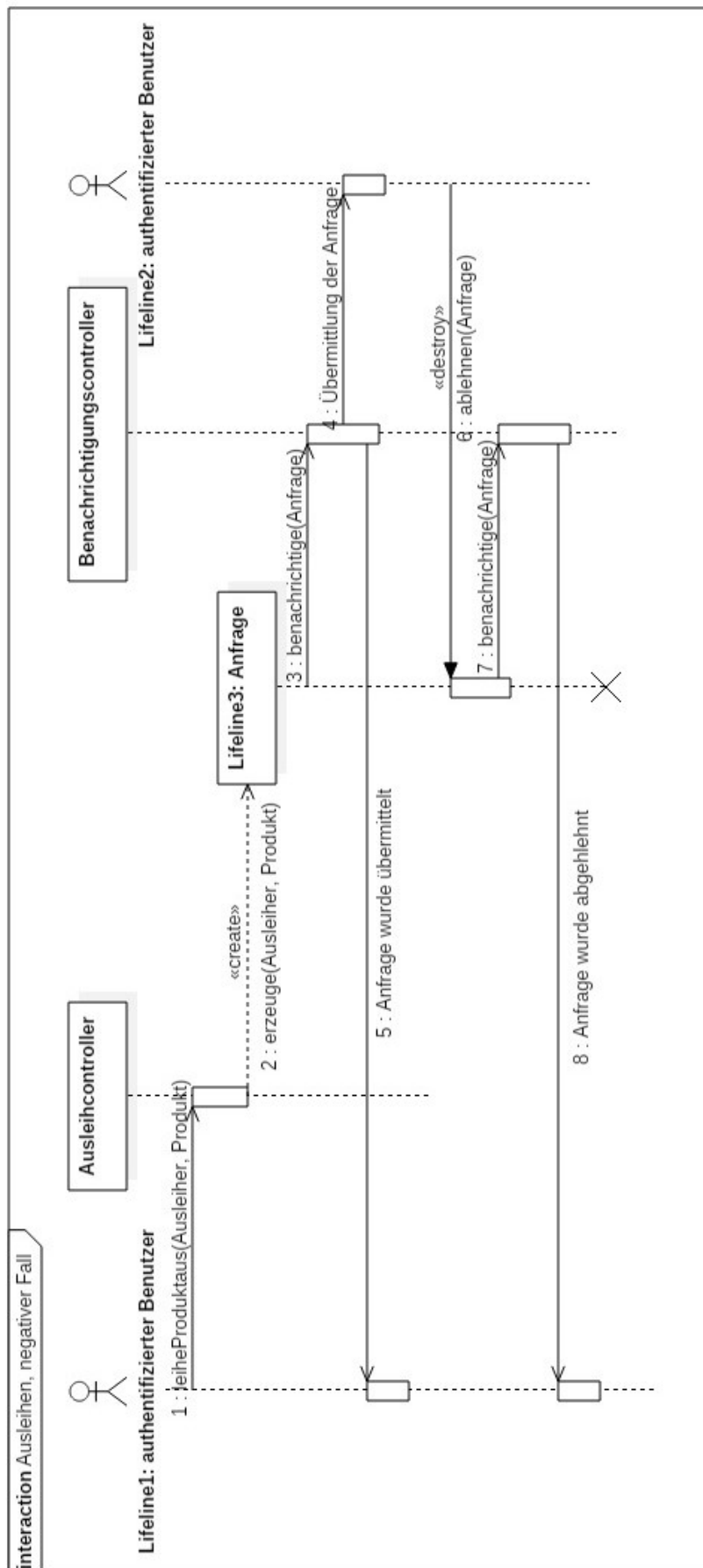


Abbildung 10.2: Sequenzdiagramm der Anfrage, negativer Fall

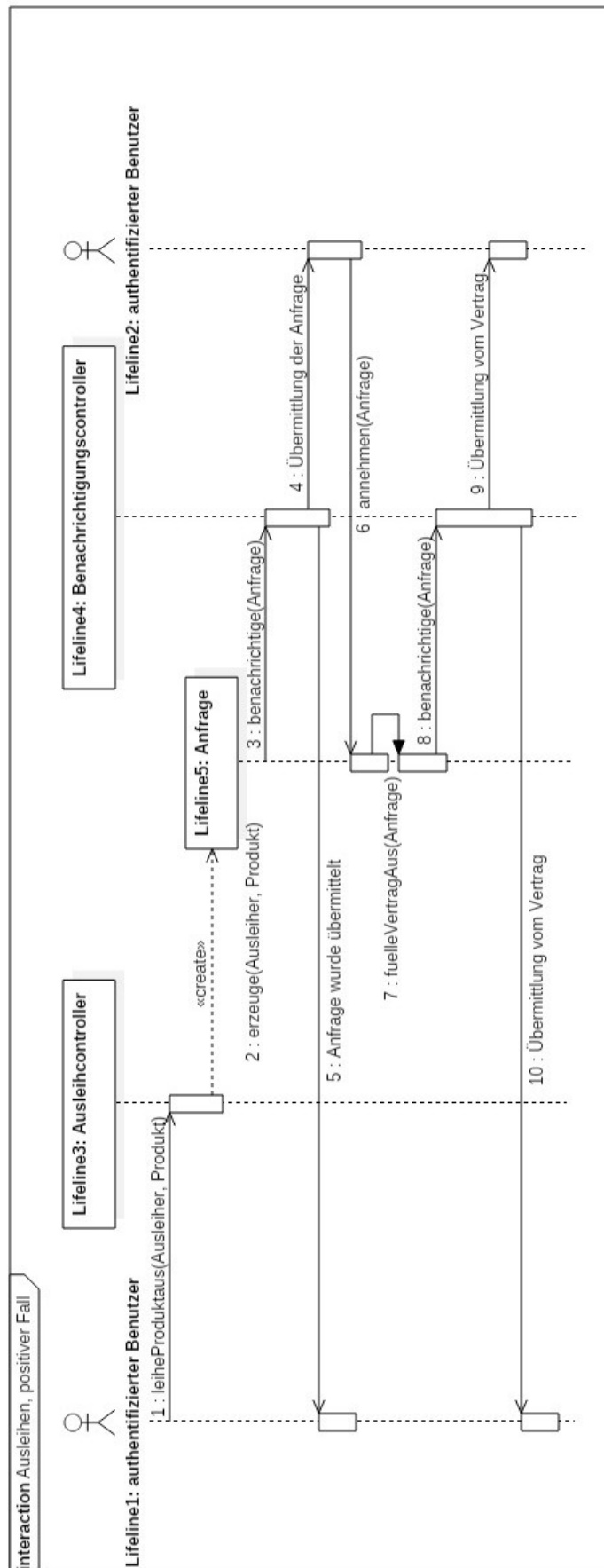


Abbildung 10.3: Sequenzdiagramm der Anfrage, positiver Fall

# 11 Mock-Ups

Im nachfolgenden Kapitel werden die für die wichtigsten Funktionen entworfenen Mock-ups vorgestellt. Dabei werden auch verschiedene Eingabevarianten beim Verleihen beachtet und näher erläutert.

## 11.1 Produkt suchen

Im dem Dialog "Produkte suchen" kann der Benutzer im oberen Textfeld seinen Suchbegriff eingeben. Daraufhin werden ihm beim Tippen dynamisch die passenden Ergebnisse unterhalb des Suchfeldes in Listenform präsentiert. Hierbei werden ein verkleinertes Produktbild, der Produktname sowie die Entfernung zum Ort der Abholung angezeigt. Mit einem Klick auf ein Produkt wird der Benutzer zur Detailansicht desselben weitergeleitet.

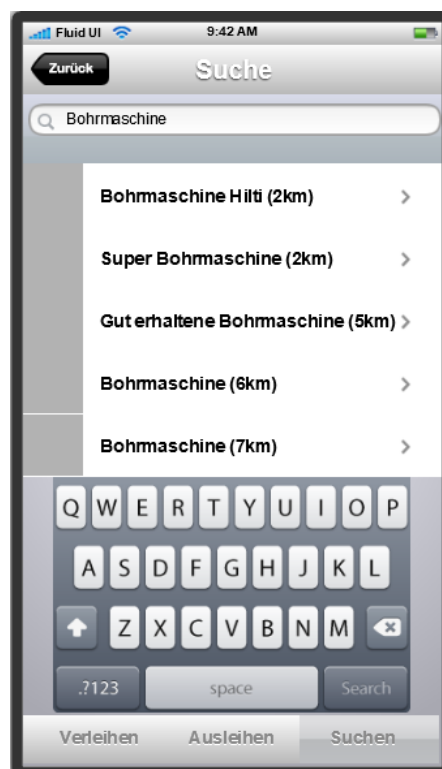


Abbildung 11.1: Mock-up: Teil 1

## 11.2 Produktdetails anzeigen

In der Detailansicht werden dem Benutzer in der ersten Zeile ein verkleinertes Profilbild des Verleihers, dessen Benutzername und daneben die Entfernung zum Ort der Abholung angezeigt. Darunter befinden sich der Produktname, ein Produktbild und die detaillierte Produktbeschreibung. Bei einem Klick auf das Produktbild öffnet sich eine Galerie, die Zugriff auf alle Bilder des Produktes ermöglicht. Am unteren Ende des Fensters befindet sich der Button zum Versenden einer Anfrage für das betreffende Produkt. Der Benutzer wird dabei auf den Dialog zum Erstellen der Anfrage weitergeleitet.



Abbildung 11.2: Mock-up: Suche, Teil 2

## 11.3 Produkt verleihen

Der Dialog zum Hinzufügen eines neuen Produktes besteht aus zwei Fenstern. Im ersten Fenster werden die allgemeinen Produktinformationen eingegeben. Darunter der Produktname, die Produktbilder und die Ausleihdauer. Die zugehörigen Elemente sind untereinander angeordnet. Neben dem Text "Produktbilder hinzufügen" befindet sich ein Kamerasymbol, welches bei einem Klick darauf, den Standarddialog des Betriebssystems zur Auswahl / Aufnahme eines Bildes anzeigt. Der Zeitraum kann mittels eines Sliders festgelegt werden. In der Titelleiste befindet sich das Navigationssymbol "Weiter" um den Benutzer zur zweiten Seite weiterzuleiten. Im zweiten Fenster werden unter der Überschrift ein verkleinertes Produktbild, das Textfeld für die Detailbeschreibung, das Textfeld für den Ort der Abholung und das Auswahlfeld für die Kautionshöhe angezeigt. Wird das Feld für die Kautionshöhe aktiviert, dann wird darunter ein Textfeld generiert, in dem die Höhe eingetragen werden kann. Am unteren Fensterrand befindet sich der Button "Verleihen", welcher das Produkt den anderen Benutzern zum Verleih anbietet und den Vorgang abschließt.



Abbildung 11.3: Mock-up: Verleihen - Produktinfos eingeben





Abbildung 11.4: Mock-up: Verleihen - ohne Kaution eingeben



Abbildung 11.5: Mock-up: Verleihen - mit Kaution eingeben

# 12 Projekt-Resume

Zusammenfassend wird in diesem Kapitel das Projekt "xShare" unter dem Aspekt reflektiert, was als der Vorlesung "Advanced Softwareengineering" gelernt, im Projekt umgesetzt werden konnte und wie es das Team weitergebracht hat.

Die Vorlesung führte anhand des Software-Lebenszyklus durch die Entwicklung eines neuen Produktes. Die Wichtigkeit von konsistenter Beschreibung und einem hohen Abstraktionsgrad der einzelnen Elemente wurde beim durchlaufen der Phasen Anforderungen, Spezifikation und Design überaus deutlich und führt damit zu guter Verständlichkeit der Ergebnisse. Desweiteren war es dem Team möglich, durch das in der Vorlesung besprochene und im Projekt angewendete Vorgehensmodell Scrum, ein modernes und agiles Konzept in der Praxis anzuwenden.

Besonderen Nutzen konnte das Team aus der Vertiefung der Kenntnisse in UML, bei der Erstellung von Klassen- und Sequenzdiagrammen für die weitere Entwicklung schöpfen. Hierbei nicht zu vergessen ist die Einführung in das UML Tool "Star UML".

Abschließend hat die Vorlesung und die Durchführung Projekts bei allen Teammitgliedern das Wissen und die Fähigkeiten um gute Software zu entwickeln wesentlich verbessert. Außerdem ist im Team der Entschluss gereift, die entworfene Anwendung, im Hinblick auf die gute Produktidee, nach Abschluss des Studiums zu implementieren und zu betreiben.