

|  |
| --- |
| **Projektbericht – Team 14** |
| Entwicklung einer Java EE Web-Shop Anwendung (SWE3)  Hochschule Bremerhaven |Wintersemester 2019-2020| Marcel Malitz |
| Name = Ben Inza Coulibaly  Email = [bcoulibaly@hs-bremerhaven.de](mailto:bcoulibaly@hs-bremerhaven.de)  Matikel Nr. = 33460 |
|  |





Inhalt

[1- Teamvorstellung 1](#_Toc35059346)

[2- Semesteraufgabe 1](#_Toc35059347)

**Es konnten keine Einträge für ein Abbildungsverzeichnis gefunden werden.**

# Teamvorstellung

Das Team bestand aus drei Personen. Jedoch im Laufe des Projekts, hatten Meinungsverschiedenheit und unterschiedlichen Ziele. Aus diesem Grund habe ich das Projekt allein zu machen. Ich, Ben Inza Coulibaly, bin gleichzeitig der Teamleiter und Entwickler dieses Projekt. Die Arbeit eines Teamleiters besteht darin, das Projekt in mehreren Aufgaben zu teilen und dies dann dementsprechend weiter an den anderen Teammitgliedern, zuzuweisen. Außerdem sorgt er dafür, dass das Projekt Fehlerfrei läuft und das Liefer-Datum des Endprodukts eingehalten wird. Meine Arbeit als Entwickler beide Entwickler war die Aufgaben zu bearbeiten und zu dokumentieren.

 **Teamleiter und Entwickler**

Ben Inza Coulibaly

# Semesteraufgabe

Als Semesteraufgabe müssen die Studierenden der Hochschule Bremerhaven eine funktionsfähige Web-Applikation entwickeln. Die Art der Web-Applikation ist uns überlassen, jedoch musste sie folgende Bedingungen erfüllen:

1. JEE Web Profile Anwendung: Dynamisches Web-Projekt unter Verwendung folgender Werkzeuge und Techniken
2. Java Enterprise 7
3. Java 8
4. Projekt muss in Eclipse IDE 2019-09 Importierbar sein
5. Maven 3.6.x
6. Wildfly 10.1.0.Final
7. JavaServer Faces, CDI, Persistence API, Enterprise JavaBeans

Den 14.03.2020 ist als Abgabe-Termin festgesetzt. Ich habe mich für einen Web-Shop namens BELISSO SHOP entschieden, auf dem wir, als Betreiber des Shops, verschiedene elektronische Produkte anbieten.

Folgende Technologien wurden zusätzlich während der Entwicklung angewendet:

1. Primefaces 7.0 und Bootfaces (für die Verschönerung der Seiten)
2. Die I18n (für die Sprache der Web-Shop (englisch und deutsch)
3. XML-Dateien für Konfiguration des Applikations (web.xml)
4. H2-Console und JPA-Persistence (Für die Datenbank-Aktionen)

# Beschreibung des Gesamtsystems von Belisso Shop

Der Shop kann auf zwei Ebene genutzt werden. Auf den ersten Ebene, anders genannt Kunde-Ebene, hat man die Möglichkeit eine Kunde-Konto mit seinen persönliche, Daten zu erstellen, seine persönlichen Daten zu bearbeiten, Artikel in seinem Warenkorb hinzuzufügen bzw. zu löschen, seinem Warenkorb zu leeren, unsere Artikel-Portfolio zu erkunden, vorausgesetzt mit einer Kreditkarte einzukaufen.

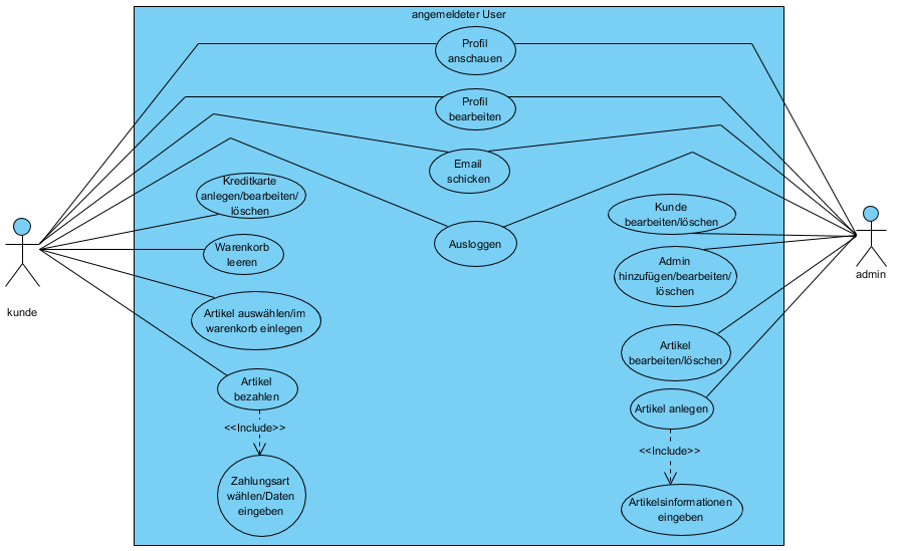
Auf den zweiten Ebene, auch Administratoren-Ebene genannt, hat man die Möglichkeit Kunde und Artikels zu verwalten. Anders gesagt, Kunde-Daten zu bearbeiten bzw. löschen. Dann neue Administratoren hinzufügen bzw. löschen. Schließlich Produkte-Daten bei Bedarf zu aktualisieren oder zu löschen.

Die Startseite des Shops dient nur den Zweck ein Einblick in dem Produkt-Portfolio zu werfen. Erst wenn das Interesse besteht, einen Einkauf durchzuführen, muss man sich anhand seiner persönlichen Daten (Kreditkarte, Name, Geburtsdatum usw.…) einloggen bzw. registrieren. Erst nach der Registrierung besitzt der Kunde einen Warenkorb, um dort seine ausgewählten Artikel hinzulegen, für einen späteren Einkauf. In der Abschlussphase des Einkaufs, hat den Kunde nochmal die Möglichkeit eine Ansicht seines Warenkorbs mit Artikel-Details zu bekommen und seine Kreditkarte-Daten zu überprüfen bzw. zu korrigieren. Dann nach einer Auftragsgabe wird der Warenkorb des Kunden automatisch geleert. Außerdem manche Seiten bedürfen eine Anmeldung, sonst wird man auf die Anmelde-Seite geleitet.

Bei Probleme, Bemerkungen oder rechtlichen Gebrauch haben wir unserem Impressum und Kontakt niedergeschrieben.

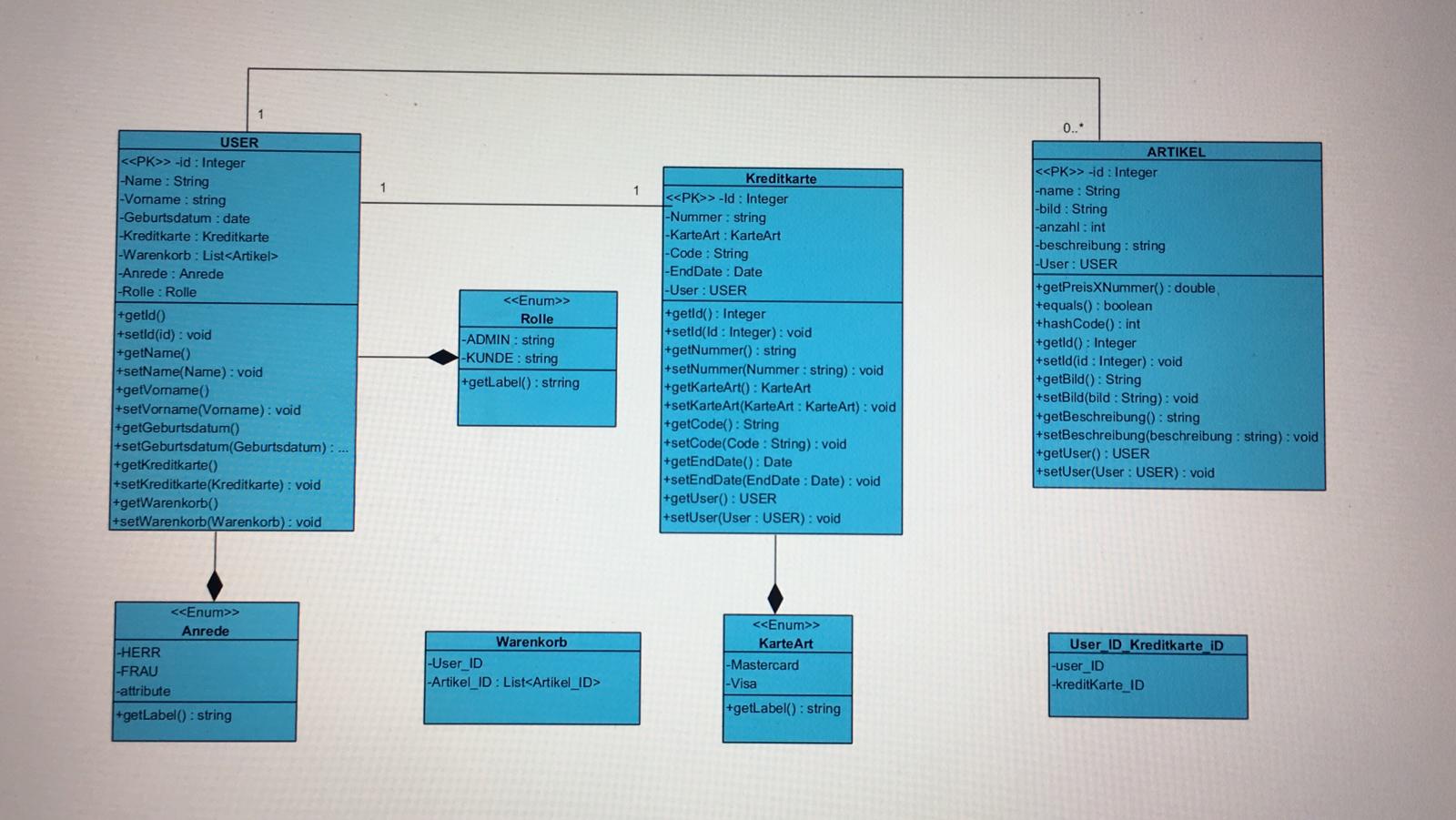
# Anforderungsdiagramm

# 4.1- Uses-Case Diagramm



Die obenstehende Graphik weist drauf ein, welche Aktionen der jeweiligen Benutzer durchführen können. Zum Beispiel können sich beide ausloggen, jedoch kann der Admin nicht einkaufen (Artikel in dem Warenkorb einlegen, dann bezahlen) und der Kunde kann auch keine Admin-Konto löschen oder welches anlegen.

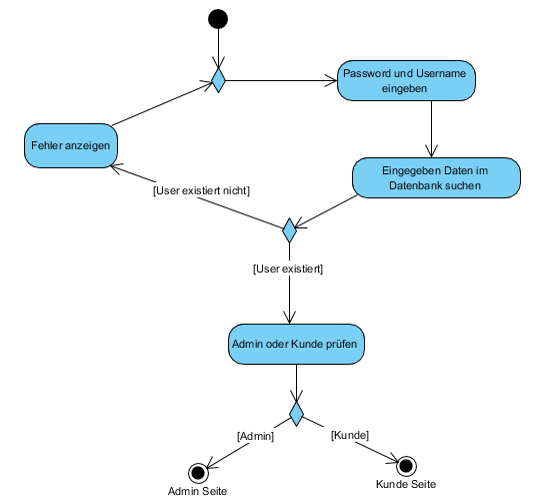
# 4.2- Klassendiagramm



Das obige Klassendiagramm beschreibt den die Datenbank unseres Shops.  
Die User-Klasse hat eine „EinsZuEins-Beziehung“ zu der Kreditkarte-Klasse. Das bedeutet das eine Kunde kann nur eine Kreditkarte besitzen und wiederrum. Aber ein Kunde kann mehrere Artikel kaufen und ein Artikel kann an verschiedene Kunde verkauft werden. Diese Beziehung besteht zwischen die Klassen User und Artikels. Solche Verbindungen werden zusätzlich in einer externe Table wie „Warenkorb“ und „UserId\_Kreditkarte\_Id“ gespeichert.

Die Enums-Klassen wie „Rolle, Anrede“ werden von der Klasse User benutzt, um den Status des Users zu bestimmen. Genauso wie die Klasse „KarteArt“ bestimmt den Typ der Kreditkarte.

# 4.3- Aktivitätsdiagramm des Einloggens



Das Einloggen-Aktivitätsdiagramm zeigt den Verlauf eines Users, der sich anmelden möchte.

Zuerst wird er aufgefordert seine Einloggen-Daten einzugeben. Dann werden die eingegebenen Daten mit denen unsere Datenbank verglichen und wenn Sie zutreffend sind, dann wird der User auf die entsprechende Seite weitergeleitet. Ansonsten bekommt er eine Fehlermeldung zu sehen.

# Aufwandschätzung

Für die Kalkulation der Aufwandschäzung habe ich Openprojekt benutzt. Dann stellte sich heraus dass, die Zeit für die Vorgesehene Aufgabe nicht eingehalten könnte. Solche Fehler tretten auf, wenn man sich selbst überschätzt oder