# 1. Use-Case „Einkaufen im Shop“

|  |  |
| --- | --- |
| Titel | Einkaufen im Shop |
| Akteur | Kunde |
| Ziel | Bestellen von einem oder mehreren Produkten |
| Auslöser | Kunde entscheidet sich zum Kauf von Produkten |
| Vorbedingung | Die Produkte stehen in der gewünschten Quantität bereit |
| Nachbedingung | - Der Bestand der Produkte wurde entsprechend der Bestellmenge reduziert  - Eine Bestellung mit den betreffenden Produkten wurde aufgegeben und persistiert  - Warenkorb ist nun wieder leer |
| Erfolgsszenario | 1. Kunde ruft unseren Shop auf  2. Kunde fügt beliebig viele Produkte seinem Warenkorb hinzu  3. Kunde kauft die im Warenkorb befindlichen Produkte  4. System benachrichtigt über den erfolgreichen Kauf  5. System persistiert die Änderungen (Siehe Nachbedingungen) |

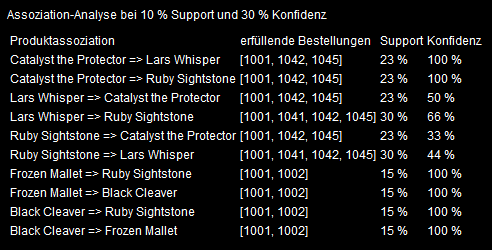
# 2. Implementation der Assoziationsanalyse

Implementation für die Assoziationsanalyse:

Für die Assoziationsanalyse haben wir unserem Projekt eine eigene Klasse angelegt, welche ausschließlich das Analysieren der vorhandenen Daten übernimmt, welche mit unserem zuvor geschriebenen BestellungLoader bereitgestellt werden, und die verdauten Daten anschließend über eine Schnittstelle anbietet. Die Analyse ist mittels minSupport und minKonfidenz Angaben zu parametrisieren. Die Ergebnisse der Analyse wurden in die Admin-Ansicht eingebunden:



Upps, falsches Bild. Hier ist die Ergebnissausgabe:



# 3. Review zur ISO Norm 9241 Teil 110 von ???

# 4. Web Analyse Komponente

Wir haben uns dazu entschieden eine eigene grundlegende Implementation einer Webanalyse einzusetzen.

Zu diesem Zwecke haben wir eine eigene Datenbanktabelle mit folgendem erweiterbarem Format:

**CREATE** **TABLE** Analytics(

website **VARCHAR**(50) **NOT** **NULL**,

aufrufe **INTEGER** **NOT** **NULL** **CHECK**(aufrufe >= 0)

);

Diese erfüllt unsere aktuellen Anforderungen und ließe sich beliebig erweitern. Die Ausgabe der Webanalyse lässt sich im Admin-Bereich unserer Seite beobachten:

