

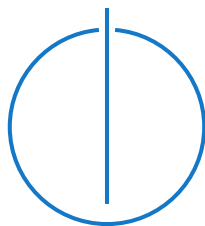


FAKULTÄT FÜR INFORMATIK  
DER TECHNISCHEN UNIVERSITÄT MÜNCHEN

Bachelorarbeit in Informatik

HYBRID WIKI ALS ALTERNATIVE  
ZU EINER ERP-BASIERTEN LÖSUNG  
FÜR EINEN STUDENTISCHEN  
GETRÄNKEMARKT

Raffael Comi





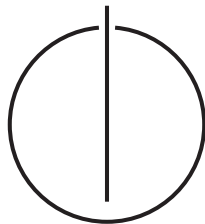
FAKULTÄT FÜR INFORMATIK  
DER TECHNISCHEN UNIVERSITÄT MÜNCHEN

Bachelorarbeit in Informatik

HYBRID WIKI ALS ALTERNATIVE ZU  
EINER ERP-BASIERTEN LÖSUNG FÜR  
EINEN STUDENTISCHEN GETRÄNKEMARKT

HYBRID WIKI AS AN ALTERNATIVE TO AN  
ERP-BASED SOLUTION FOR A COLLEGIATE  
BEVERAGE STORE

Bearbeiter: Raffael Comi  
Aufgabensteller: Prof. Dr. rer. nat. Florian Matthes  
Betreuer: Dipl.-Inf. Univ. Ivan Monahov  
Abgabedatum: 12.12.2011



# Erklärung

Ich versichere, dass ich diese Bachelorarbeit selbständig verfasst und nur die angegebenen Quellen und Hilfsmittel verwendet habe.

München, den 12. Dezember 2011

Raffael Comi

## Zusammenfassung

ERP-Systeme sind aus dem heutigen Geschäftsalltag nicht mehr wegzudenken. Für Kleinstunternehmen, wie die in dieser Arbeit behandelten studentischen Getränkemärkte der Studentenstadt Freimann, ist der Anpassungsaufwand eines ERP-Systems an die eigenen Anforderungen jedoch sehr hoch.

In dieser Arbeit zeigen wir daher erstmalig auf, ob sich ein Hybrid-Wiki-System als Alternative zu einer ERP-basierten Lösung für einen studentischen Getränkemarkt eignet. Dazu beschreiben wir zunächst den Aufbau der Getränkemärkte der Studentenstadt Freimann und erarbeiten ein UML-Modell, welches diese Märkte repräsentiert. Dieses UML-Modell setzen wir im Hybrid-Wiki-System *Tricia* um. Damit können wir bereits zeigen, dass sich ein studentischer Getränkemarkt in *Tricia* abbilden lässt. Um jedoch alternativ zu einem ERP-System genutzt zu werden, müssen noch Funktionalitäten eines solchen implementiert werden. Folglich identifizieren wir zunächst, durch eine persönliche Befragung relevanter Akteure, mögliche Kennzahlen und priorisieren diese mittels einer Online-Umfrage unter den selben Akteuren. Abschließend erweitern wir unser bereits erstelltes Modell so, dass wir die gefragtesten Kennzahlen mittels OCL-Ausdrücken berechnen können und passen die Umsetzung in *Tricia* dementsprechend an.

Somit können wir zeigen, dass der Einsatz eines Hybrid-Wiki-Systems als Alternative zu einer ERP-basierten Lösung für einen studentischen Getränkemarkt möglich ist. Das Hybrid-Wiki-System *Tricia* muss dazu allerdings noch um einzelne Funktionalitäten erweitert werden.

# Inhaltsverzeichnis

<b>Abbildungsverzeichnis</b>	<b>IV</b>
<b>Verzeichnis der Listings</b>	<b>V</b>
<b>1. Einleitung</b>	<b>1</b>
1.1. Motivation . . . . .	1
1.2. Getränkemarkte in der Studentenstadt Freimann . . . . .	2
1.3. Rollen eines Getränkemarktes in der Studentenstadt Freimann .	5
1.4. Prozesse eines Getränkemarktes in der Studentenstadt Freimann	8
<b>2. Forschungsfragen, Forschungsziele und Artefakte</b>	<b>11</b>
<b>3. Abbildung eines studentischen Getränkemarktes im Hybrid-Wiki-System Tricia</b>	<b>13</b>
3.1. Erstellung des UML-Modells . . . . .	13
3.2. Umsetzung des Modells in Tricia . . . . .	16
<b>4. Ermittlung relevanter Kennzahlen</b>	<b>17</b>
4.1. Identifizierung möglicher Kennzahlen . . . . .	17
4.1.1. Erstellung des Fragebogens . . . . .	18
4.1.2. Ergebnisse . . . . .	18
4.2. Priorisierung der Kennzahlen . . . . .	23
4.2.1. Erstellung der Online-Umfrage . . . . .	23
4.2.2. Ergebnisse . . . . .	24
<b>5. Verortung und Implementierung der Kennzahlen im Hybrid Wiki</b>	<b>29</b>
5.1. Erweiterung des Modells . . . . .	29
5.2. Zugriff auf die Kennzahlen . . . . .	52
5.3. Anpassung des Hybridmodells in Tricia . . . . .	54
<b>6. Zusammenfassung und Ausblick</b>	<b>55</b>

<b>Literaturverzeichnis</b>	<b>57</b>
<b>A. Anhang</b>	<b>i</b>
A.1. Richtlinien für den Brotladen . . . . .	i
A.2. Richtlinien des Getränkeverkaufs des Hanns-Seidel-Hauses . . .	viii
A.3. Fragebogen zur Befragung beteiligter Akteure nach Kennzahlen für einen studentischen Getränkemarkt . . . . .	xi
A.4. Ergebnisse der Befragung . . . . .	xiv
A.5. Ergebnisse der Online-Umfrage . . . . .	xxi

# Abbildungsverzeichnis

1.1. Unterschiedliche Rollen bei einem studentischen Getränkemarkt in der Studentenstadt Freimann . . . . .	5
3.1. Erster Entwurf des UML-Modells . . . . .	14
3.2. Zweiter Entwurf des UML-Modells . . . . .	14
3.3. Finaler Entwurf des UML-Modells . . . . .	15
3.4. Das fertige UML-Modell . . . . .	16
4.1. Auswertung der Online-Umfrage - Alle Umfrageteilnehmer . . .	27
4.2. Auswertung der Online-Umfrage - Nach Gruppen von Akteuren	28
5.1. Das erweiterte UML-Modell . . . . .	30
5.2. Die Klasse <i>Account</i> . . . . .	31
5.3. Die Klasse <i>Batch</i> . . . . .	32
5.4. Die Klasse <i>Customer</i> . . . . .	32
5.5. Die Klasse <i>CustomerGroup</i> . . . . .	34
5.6. Die Klasse <i>DateTime</i> . . . . .	35
5.7. Die Klasse <i>Employee</i> . . . . .	38
5.8. Die Klasse <i>Manager</i> . . . . .	39
5.9. Die Klasse <i>Market</i> . . . . .	40
5.10. Die Klasse <i>Person</i> . . . . .	41
5.11. Die Klasse <i>Product</i> . . . . .	42
5.12. Die Klasse <i>ProductHistory</i> . . . . .	46
5.13. Die Klasse <i>Promotion</i> . . . . .	47
5.14. Die Klasse <i>Purchase</i> . . . . .	47
5.15. Die Klasse <i>PurchaseLineItem</i> . . . . .	48
5.16. Die Klasse <i>Sale</i> . . . . .	49
5.17. Die Klasse <i>SaleLineItem</i> . . . . .	50
5.18. Die Klasse <i>Supplier</i> . . . . .	51
5.19. Die Klasse <i>Transaction</i> . . . . .	51

## Verzeichnis der Listings

5.1. Constraints der Klasse <i>Account</i> . . . . .	31
5.2. Constraints der Klasse <i>Customer</i> . . . . .	33
5.3. Constraints der Klasse <i>CustomerGroup</i> . . . . .	34
5.4. Constraints der Klasse <i>DateTime</i> . . . . .	36
5.5. Constraints der Klasse <i>Employee</i> . . . . .	39
5.6. Constraints der Klasse <i>Market</i> . . . . .	41
5.7. Constraints der Klasse <i>Product</i> . . . . .	43
5.8. Constraints der Klasse <i>Sale</i> . . . . .	50
5.9. Constraints der Klasse <i>Transaction</i> . . . . .	52



# 1. Einleitung

## 1.1. Motivation

Das Betreiben eines studentischen Getränkemarktes ist eine komplexe und aufwändige Aufgabe. Für die ca. 2500 Bewohner der Studentenstadt Freimann (kurz *StuSta*), welche laut [Stu] die größte studentische Wohnanlage in Deutschland ist, gibt es sechs Verkaufsstätten in denen sowohl Getränke als auch Knabbereien, Tiefkühlwaren und Haushaltsartikel angeboten werden. Die meisten dieser studentischen Getränkemarkte verwenden kaum technische Hilfsmittel. So verwendet lediglich ein einziger Markt eine Registrierkasse und ein weiterer setzt einen Computer mit Kassensoftware ein. Die restlichen vier Märkte dokumentieren die Verkaufsvorgänge nach Bedarf noch immer in Papierform. In der Planung, Abrechnung und Buchhaltung hingegen setzen alle Betreiber immerhin Tabellenkalkulationsprogramme ein.

Für die Dokumentation der Ein- und Verkäufe, sowie für die Planung der Lagerbestände und das damit verbundene Rechnungswesen bietet sich der Einsatz von ERP<sup>1</sup>-Systemen an. Laut [Ni08, Seite 14] ist ERP-Software:

Integrierte Software zum Verwalten sämtlicher Unternehmensressourcen, wie z.B. Kapital, Betriebsmittel und Personal, sowie zur umfassenden Unterstützung und Abbildung betrieblicher Prozesse.

Die größten Anbieter von ERP-Software in Deutschland sind laut [Mi09] (nach Marktanteil sortiert): SAP, Infor, Microsoft, Sage und Oracle. Alternativ gibt es auch für diesen Bereich Open-Source-Lösungen wie beispielsweise Compiere (siehe [Com]), Openbravo (siehe [Opea]) oder der für den deutschen Markt angepasste Openbravo-Fork OpenZ (siehe [Opeb]). Eine Gemeinsamkeit aller ERP-Systeme ist, dass diese speziell an jedes Unternehmen angepasst werden

---

<sup>1</sup>Enterprise Resource Planning [Gr10]

müssen. Ein studentischer Getränkemarkt hat im Vergleich zu klassischen Getränkemarkten mehrere Besonderheiten, da für diesen keine konkrete Rechtsgrundlage existiert. Existierende ERP-Systeme für klassische Getränkemarkte müssen daher stark angepasst werden, was sehr hohe Aufwände zu Folge hat.

In dieser Arbeit soll evaluiert werden, ob die für einen studentischen Getränkemarkt nötigen Funktionalitäten eines ERP-Systems auch in einem Hybrid-Wiki-System umgesetzt werden können. Der Vorteil dieses Ansatzes liegt darin, dass im Hybrid-Wiki-System keine für einen solchen Kleinstbetrieb überflüssigen Funktionalitäten implementiert sind, welche die Anpassung aufwändig machen können. Dennoch kann das System flexibel genug gestaltet werden, dass es in unterschiedlichen studentischen Getränkemarkten eingesetzt werden kann. Als Vertreter dieser Kategorie wird in dieser Arbeit das Hybrid-Wiki-System *Tricia* (siehe [Tri]) verwendet.

## 1.2. Getränkemarkte in der Studentenstadt Freimann

In diesem Abschnitt werden die einzelnen Getränkemarkte der Studentenstadt Freimann in alphabetischer Reihenfolge vorgestellt. Die Benennung der einzelnen Märkte erfolgt hierbei nach der Übersichtsseite über Einkaufsmöglichkeiten innerhalb der StuSta aus deren internen Wiki-System (siehe [Stu11]<sup>2</sup>).

### Brotladen

Im Brotladen werden hauptsächlich Kaffee und Backwaren verkauft. Zusätzlich werden auch Getränke sowie Tabakwaren, Süßwaren, Snacks und Fertiggerichte angeboten. Geöffnet ist der Brotladen montags bis freitags von 7 bis 11 Uhr, sowie von 16 bis 18 Uhr. Am Wochenende ist er von 9 bis 13 Uhr geöffnet. Durch diese Öffnungszeiten ist er der einzige Markt in der StuSta, der länger als eine Stunde pro Tag geöffnet hat. Der Betreiber untersteht dem GHR<sup>3</sup> und

---

<sup>2</sup>Quelle nur innerhalb des Netzwerkes der Studentenstadt Freimann ohne Anmeldung zugänglich. Mittels eines dort erstellten Accounts ist die Seite auch innerhalb des Münchner Wissenschaftsnetzes erreichbar.

<sup>3</sup>Gesamtheimrat, Versammlung der Repräsentanten jedes Hauses in der Studentenstadt.

wird von diesem gewählt und durch dessen Wirtschaftsausschuss geprüft. Der GHR legt ebenfalls Richtlinien für den Brotladen fest (siehe Anhang A.1).

## **Getränkeladen EWH**

Der Getränkeladen des EWHs<sup>4</sup> bietet verschiedene Biere, alkoholfreie Getränke, Spirituosen und Fertiggerichte an. Geöffnet ist er montags, mittwochs und donnerstags von 20 bis 21 Uhr. In der vorlesungsfreien Zeit ist der Laden mittwochs geschlossen. Der Betreiber hat nur wenige Auflagen zu beachten. So gibt es für den Getränkeladen weder schriftliche Richtlinien, noch werden vom EWH Prüfungen der Buchführung jedweder Art durchgeführt.

## **Getränkeverkauf Blaues Haus**

Der Getränkeverkauf des Blauen Hauses befindet sich zwar in dessen Keller, ist jedoch auch von außen zugänglich. Es werden Biere und alkoholfreie Getränke sowie Süßigkeiten und die Wertmarken für die Waschmaschinen und Trockner des Blauen Hauses angeboten. Der Markt ist montags, dienstags, donnerstags und freitags, in der vorlesungsfreien Zeit montags, mittwochs und freitags, von 19 bis 20 Uhr geöffnet. Auch hier gibt es keine schriftlichen Richtlinien, jedoch wird hier die Buchführung in unregelmäßigen Abständen von Haussprechern<sup>5</sup> des Hauses geprüft.

## **Getränkeverkauf im Orangen Haus**

Der Getränkeverkauf im Orangen Haus ist, nach der Verkaufs- und Lagerfläche, der größte Getränkemarkt in der Studentenstadt Freimann. Er befindet sich im südlichen Keller des Hauses und ist montags bis samstags von 19 bis 20 Uhr geöffnet. Angeboten werden neben Bier, alkoholfreien Getränken und Spirituosen auch Süßwaren, Snacks, Fertiggerichte, Tiefkühlwaren, sonstige Lebensmittel und Haushaltsartikel. Auch hier gibt es nur mündliche Vorgaben für

---

<sup>4</sup>Egon-Wiberg-Haus

<sup>5</sup>Ein Haussprecher repräsentiert das Haus und dient den Bewohnern als Ansprechpartner bei Problemen. Er wird von den Bewohnern des Hauses gewählt.

den Betreiber. Dessen Buchführung wird jedoch regelmäßig von einem Haus-sprecher kontrolliert. In diesem Markt wird ein Computer mit Kassensoftware eingesetzt.

### **HSH-Getränkeverkauf**

Der Getränkeverkauf des HSHs<sup>6</sup> befindet sich nicht, wie die anderen Märkte, im Keller des eigenen Hauses, sondern, aufgrund von fehlenden Räumlichkeiten, im benachbarten Haus 10. Obwohl er dem einwohnerstärksten Haus (ungefähr 650 Bewohner) angehört, ist er, nach der Verkaufs- und Lagerfläche, der kleinste Getränkemarkt in der StuSta. Aufgrund der hohen Anzahl an Bewohnern des Hauses gibt es für den Getränkeverkauf strikte Vorgaben, festgehalten in entsprechenden Richtlinien (siehe Anhang A.2). So findet alle zwei Monate eine Inventur und Kassenprüfung durch den Getränkeverkaufsausschuss, welcher durch den Heimrat (siehe 1.3) des HSH bestimmt wird, statt. Der HSH-Getränkeverkauf ist montags bis samstags von 19 bis 20 Uhr geöffnet und bietet neben diversen Bieren, alkoholfreien Getränken und Spirituosen auch Süßwaren, Snacks, Fertiggerichte, einige Haushaltsartikel und die Wertmarken für die Waschmaschinen und Trockner des HSH an. Er ist der einzige Getränkemarkt in der Studentenstadt, welcher für bestimmte Kundengruppen eine Zahlung auf Rechnung anbietet. In diesem Markt wird eine Registrierkasse eingesetzt.

### **Rotes Haus - Laden**

Der Laden des Roten Hauses liegt direkt am Eingang der Studentenstadt Freimann und ist über eine Außentreppe an der Nordseite des Hauses erreichbar. Er ist täglich von 20 bis 21 Uhr geöffnet und bietet neben Bier, alkoholfreien Getränken und Spirituosen auch Süßwaren, Snacks, Fertiggerichte, sonstige Lebensmittel, Haushaltsartikel und die Wertmarken für die Waschmaschinen und Trockner des Roten Hauses an. Es gibt keine schriftlichen Richtlinien an die sich der Betreiber halten muss, jedoch wird regelmäßig von den Hausverantwortlichen überprüft, ob sich der Betreiber an die mündlichen Vorgaben hält.

---

<sup>6</sup>Hanns-Seidel-Haus

### 1.3. Rollen eines Getränkemarktes in der Studentenstadt Freimann

Im folgenden Abschnitt werden die Rollen eines studentischen Getränkemarktes in der Studentenstadt Freimann aufgelistet und erläutert.

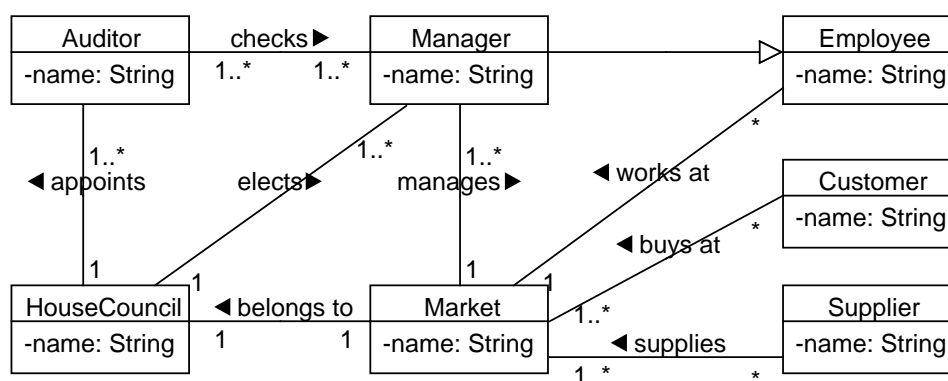


Abbildung 1.1.: Unterschiedliche Rollen bei einem studentischen Getränkemarkt in der Studentenstadt Freimann

#### Der Heimrat (*HouseCouncil*)

Der Heimrat ist die Versammlung der Stockwerkssprecher eines Hauses. Dieses Gremium ist die höchste Instanz der Heimselbstverwaltung innerhalb eines Hauses. Hier werden Richtlinien für den hauseigenen Getränkemarkt beschlossen, Betreiber gewählt und Prüfer entsandt. Fragestellungen, welche die Kompetenzen der Prüfer oder der Betreiber überschreiten, werden durch den Heimrat beantwortet.

#### Der Markt (*Market*)

Die Getränkemarkte gehören jeweils einem bestimmten Haus der Studentenstadt Freimann, repräsentiert durch dessen Heimrat, an und befinden sich meist im Keller des jeweiligen Hauses. Geöffnet sind die Verkaufsräume in der Regel nur eine Stunde am Tag. Zum Einkauf berechtigt ist jeder Bewohner der

StuSta. Externe Kunden dürfen nur als Gäste eines Bewohners bedient werden, da diese Getränkemarkte als Gemeinschaftseinrichtung ausschließlich für die Bewohner zur Verfügung stehen. Durch diese Begrenzung des Kundenkreises haben diese Getränkemarkte einen besonderen rechtlichen Status, der beispielsweise dazu führt, dass keine Steuern abgeführt werden müssen.

## **Der Betreiber (*Manager*)**

Die studentischen Getränkemarkte werden in der Regel von nur einer Person geleitet - dem Betreiber. Dieser trägt alleine die Verantwortung, dass:

- während der Öffnungszeiten immer ein Mitarbeiter vor Ort ist,
- Mitarbeiterlöhne korrekt berechnet werden,
- genügend Waren im Verkaufsraum sind,
- die Rechnungskunden eine korrekte Rechnung erhalten und diese bezahlen,
- alle Ein- und Auszahlungen korrekt dokumentiert werden,
- genügend liquide Mittel vorhanden sind, um Lieferantenrechnungen zu bezahlen und
- keine Verluste gemacht werden.

## **Die Prüfer (*Auditor*)**

Wie bereits beschrieben, gehört jeder Markt einem bestimmten Haus der Studentenstadt Freimann an. Entsprechend ist dieses Haus für die Kontrolle des Betreibers zuständig. Dafür werden je nach Haus auf verschiedene Weisen eine unterschiedliche Anzahl an Prüfern bestimmt. Diese handeln nach vom Heimrat (siehe 1.3) festgelegten Richtlinien. Hauptsächlich ist dies die Überprüfung, ob das Anfangskapital noch vorhanden ist. Dabei helfen sie teilweise auch bei der regelmäßigen Inventur. Ebenso überprüfen sie das vom Betreiber geführte Kassenbuch anhand aller schriftlichen Belege auf Korrektheit.

## Die Mitarbeiter/Verkäufer (*Employee*)

Die Mitarbeiter sind meist Bewohner des Hauses, dem der Markt, in welchem sie arbeiten, angehört. Sie haben feste Arbeitstage. Falls sie nicht arbeiten können, muss der Betreiber Ersatz finden und notfalls selbst die Aufgaben erledigen. Aufgaben der Mitarbeiter sind:

- Vor Öffnung des Verkaufsraumes Lieferungen korrekt einräumen.
- Zur Öffnungszeit Verkaufsvorgänge durchführen und diese nach den markt-spezifischen Vorgaben dokumentieren.
- Nach Ladenschluss die Regale wieder bestücken und den Verkaufsraum wieder in einen ordnungsgemäßen Zustand bringen.

Für eigene Einkäufe bekommen die Mitarbeiter einen festgelegten prozentualen Rabatt. Für jeden Markt gibt es einen fixen Grundstundenlohn. Dieser wird teilweise um einen Anteil vom Umsatz an den gearbeiteten Tagen erhöht. Der Lohn wird, abzüglich der Selbsteinkäufe, am Ende des Monats vom Betreiber in bar ausbezahlt.

## Die Kunden (*Customer*)

Hier wird zwischen zwei verschiedenen Arten von Kunden unterschieden:

**Barkunden** Sie bezahlen ihre gekauften Waren direkt vor dem Verlassen des Verkaufsraumes beim Verkäufer in bar.

**Rechnungskunden** Sie unterschreiben beim Verkäufer einen Beleg über die mitgenommenen Waren. Der Betreiber berechnet anhand der Belege einen eventuellen Rabatt und stellt dem Kunden monatsweise eine Rechnung per E-Mail zu. Beglichen werden kann die Rechnung einerseits in bar zur normalen Öffnungszeit beim Mitarbeiter im Verkaufsraum oder zu sonstigen Zeiten beim Betreiber und andererseits per Überweisung auf das Girokonto des Betreibers. Der Zeitraum, in welchem die Rechnung bezahlt werden muss, liegt im Ermessen des jeweiligen Betreibers.

## Der Lieferant (*Supplier*)

Der Betreiber bestellt beim Lieferanten die gewünschten Waren. Am nächsten Werktag stellt dieser die Waren zu. Wenige Tage später wird der Rechnungsbetrag vom Girokonto des Betreibers eingezogen und diesem schriftlich eine Rechnung zugeschickt.

## 1.4. Prozesse eines Getränkemarktes in der Studentenstadt Freimann

Im folgenden Abschnitt werden die Prozesse eines studentischen Getränkemarktes in der Studentenstadt Freimann aufgelistet und erläutert.

### Der Einkauf

Der Einkauf für die studentischen Getränkemarkte geschieht auf zwei verschiedene Arten:

**Lieferung** In der Regel werden den Märkten in der StuSta vor allem Getränke geliefert. Diese werden, je nach Bestand, vom Betreiber beim Lieferanten bestellt und am folgenden Arbeitstag geliefert. Lieferungen erfolgen nur montags bis freitags. Bei ebenerdigem Zugang zum Markt hat der Lieferant hierbei einen Schlüssel zum Verkaufsraum und sortiert die bestellte Ware grob ein, nimmt das vorhandene Leergut mit und hinterlässt einen Lieferschein auf dem er handschriftlich notiert, wie viel Leergut er zurücknimmt. Ist der Markt nur über eine Treppe zugänglich, erfolgt die Lieferung nur bis zu dieser Treppe. Der Betreiber muss dafür sorgen, dass eventuell zurückzugebendes Leergut bereits ebenerdig zur Abholung bereit steht und dass die gelieferte Ware über die Treppe in den Verkaufsraum beziehungsweise das Lager gelangt. Der Rechnungsbetrag wird in der Regel nach zwei Arbeitstagen per Bankeinzug von einem vorher festgelegten Girokonto eingezogen. Die Rechnung selbst erhält der Betreiber meist am Tag nach der Abbuchung per Post.



**Einkauf im Groß- und Einzelhandel** Auf diese Weise werden hauptsächlich die Produkte eingekauft, welche nicht geliefert werden. Hierzu informiert sich der Betreiber über die wöchentlichen Angebote bestimmter Groß- und Einzelhändler und erstellt eine Übersicht über lohnenswerte Angebote. Einmal pro Woche kauft er dann diese, zusammen mit dringend benötigten Artikeln, ein. Bezahlt wird hierbei mittels electronic cash (EC).

## **Die Lagerverwaltung**

Es müssen immer so viele Einheiten eines Produktes vorhanden sein, dass dieses während der Öffnungszeit des Marktes nicht ausverkauft ist. Dazu werden die Bestände regelmäßig kontrolliert und, nach Bedarf, durch Einkauf aufgefüllt. Ebenfalls wird die Haltbarkeit der Bestände überwacht. Artikel, welche ihr Mindesthaltbarkeitsdatum überschritten haben, werden aus dem Sortiment genommen.

## **Das Rechnungswesen**

Über jede Einnahme und Ausgabe muss Buch geführt werden. Dazu werden jegliche Belege gesammelt und in ein Kassenbuch eingetragen. Belege für den Verkauf werden in den Märkten mit Registrier- oder Computerkasse von diesen gespeichert und bei Bedarf ausgedruckt. In den anderen Märkten wird nach Ladenschluss händisch notiert, wie viel Geld eingenommen wurde. Überschüssiges Geld, welches nicht im Laden als Wechselgeld benötigt wird, wird vom Betreiber des Marktes der dortigen Kasse entnommen und zeitnah auf ein für diesen Zweck erstelltes Girokonto eingezahlt. Von diesem werden die Einkäufe per (EC-)Lastschrift abgebucht und StuSta-interne Rechnungen überwiesen. Ebenso ist es den Rechnungskunden möglich, ihre ausstehenden Rechnungen per Überweisung auf dieses Konto zu begleichen.

## **Der Verkauf**

Der Kunde sucht sich im Verkaufsraum die gewünschte Ware aus und bringt diese an die Kasse zum Verkäufer. Der Verkäufer kassiert nun bei einem Barkunden den entsprechenden Geldbetrag (nur Barzahlung ist möglich) oder no-

tiert bei einem Rechnungskunden dessen Kundennummer und lässt den Kunden einen entsprechenden Beleg unterschreiben (siehe dazu auch den entsprechenden Abschnitt in Kapitel 1.3).

## **Die Kassenprüfung**

Die Kassenprüfung läuft in zwei Schritten ab. Zuerst findet eine Inventur statt, bei der die Prüfer jeden Verkaufsartikel zählen und dessen Anzahl händisch notieren. Der Betreiber berechnet, anhand dieser Zählung, den Wert des Bestandes. Wenn der Betreiber die Abrechnung für den Abrechnungszeitraum abgeschlossen hat, wird diese von den Prüfern im zweiten Schritt auf Korrektheit geprüft.

## 2. Forschungsfragen, Forschungsziele und Artefakte

Das Ziel dieser Arbeit ist die Untersuchung, ob die für einen studentischen Getränkemarkt nötigen Funktionalitäten eines ERP-Systems auch in einem Hybrid Wiki implementiert werden können. Dazu muss zuerst analysiert werden, ob ein studentischer Getränkemarkt überhaupt in einem solchen System abgebildet werden kann. In dieser Arbeit werden wir dafür das Hybrid-Wiki-System *Tricia* verwenden. Weiter soll geklärt werden, welche Kennzahlen und Auswertungen den Stakeholdern durch das Hybrid-Wiki-System zur Verfügung gestellt werden müssen. Schließlich sollen diese Kennzahlen und Auswertungen in OCL<sup>7</sup> ausgedrückt und in das System integriert werden.

In dieser Arbeit werden folgende Forschungsfragen betrachtet:

**Forschungsfrage F1** *Wie lässt sich ein studentischer Getränkemarkt mittels eines Modells in einem Hybrid-Wiki-System abbilden?*

**Forschungsfrage F2** *Welche Auswertungen beziehungsweise Kennzahlen benötigen die beteiligten Stakeholder?*

**Forschungsfrage F3** *Wie lassen sich diese Kennzahlen mit OCL ausdrücken und in das erstellte Modell einbetten?*

Für die Abbildung eines studentischen Getränkemarkts in *Tricia* werden wir zunächst ein UML-Modell eines studentischen Getränkemarktes erstellen, um es daraufhin in *Tricia* mittels Hybrid umzusetzen. Das so entstandene Hybridmodell wird schließlich mit Instanzdaten befüllt. Um die Kennzahlenbedarfe

---

<sup>7</sup>Object Constraint Language

Forschungsfrage	Forschungsziele
F1	<ol style="list-style-type: none"> <li>1. Erstellung eines UML-Modells eines studentischen Getränkemarktes</li> <li>2. Umsetzung dieses Modells in Tricia mittels Hybrid</li> <li>3. Befüllung des erstellten Hybridmodells mit Instanzdaten</li> </ol>
F2	<ol style="list-style-type: none"> <li>1. Erstellung eines Fragebogens</li> <li>2. Interviews mit den Beteiligten</li> <li>3. Erstellung und Durchführung einer Online-Umfrage</li> <li>4. Erfassen, evaluieren und dokumentieren der Ergebnisse</li> </ol>
F3	<ol style="list-style-type: none"> <li>1. Übersetzung der Metriken in OCL</li> <li>2. Einbettung in passende Modellelemente</li> <li>3. Anpassung des Hybridmodells</li> </ol>

Tabelle 2.1.: Vorgehensweise zur Beantwortung der Forschungsfragen

der Stakeholder zu klären, werden wir mittels eines Fragebogens zunächst mögliche Kennzahlen identifizieren und diese im Anschluss mittels einer Online-Umfrage unter den selben Stakeholdern priorisieren. Die gefragtesten Metriken werden daraufhin in OCL übersetzt und in passende Modellelemente in Tricia eingebettet. Abschließend werden wir das bereits erstellte Hybridmodell so anpassen, dass wir auf die ermittelten Metriken zugreifen können. Anhand der Forschungsfragen werden die Forschungsziele in Tabelle 2.1 definiert.

## 3. Abbildung eines studentischen Getränkemarktes im Hybrid-Wiki-System Tricia

In diesem Kapitel wollen wir uns der Forschungsfrage F1 widmen. Dazu werden wir zuerst ein UML-Modell erstellen, das einen studentischen Getränkemarkt repräsentiert. Anschließend wird dieses Modell im Hybrid-Wiki-System *Tricia* umgesetzt und mit Instanzdaten befüllt.

### 3.1. Erstellung des UML-Modells

Um ein UML-Klassendiagramm zu erstellen, werden wir zunächst sogenannte Fachklassen identifizieren, welche vorerst keine Attribute enthalten<sup>8</sup>. Dazu verfolgen wir den Weg eines Verkaufsartikels in einem studentischen Getränkemarkt:

1. Ein Betreiber (*Manager*) tätigt (*orders*) eine Bestellung (*Purchase*). Die darin enthaltenen (*contains*) Produkte (*Product*) werden von einem Lieferanten (*Supplier*) geliefert (*delivers*).
2. Ein Betreiber erstellt (*creates*) eine Preisreduzierung (*Promotion*). Diese bewirbt (*promotes*) ein bestimmtes Produkt.
3. Ein Kunde (*Customer*) kauft (*buys*) eine Produktauswahl (*Sale*), welche bestimmte Produkte enthält (*contains*). Diese wird ihm von einem bestimmten Mitarbeiter (*Employee*) verkauft (*sells*).

Aus dieser Auflistung können wir bereits einige Klassen und deren Assoziationen untereinander herausziehen und erhalten einen ersten Entwurf des UML-Klassendiagramms (siehe Abbildung 3.1).

---

<sup>8</sup>Nach [Oe09, Seite 130].

3. Abbildung eines studentischen Getränkemarktes im Hybrid-Wiki-System Tricia

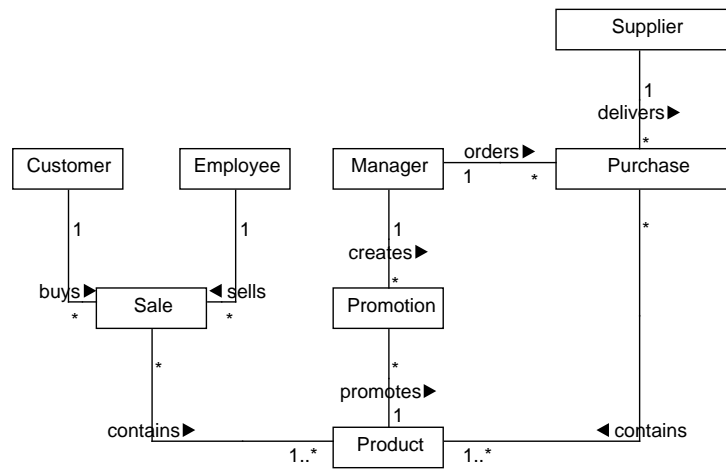


Abbildung 3.1.: Erster Entwurf des UML-Modells für einen studentischen Getränkemarkt

Wir können nun eine Vererbungshierarchie einsetzen, da Kunde und Mitarbeiter beides Personen sind und der Betreiber ein spezieller Mitarbeiter ist. Der Lieferant wird explizit nicht als Person behandelt, da es sich hier um eine StuSta-externe, juristische Person handelt. So erhalten wir einen zweiten Entwurf (siehe Abbildung 3.2).

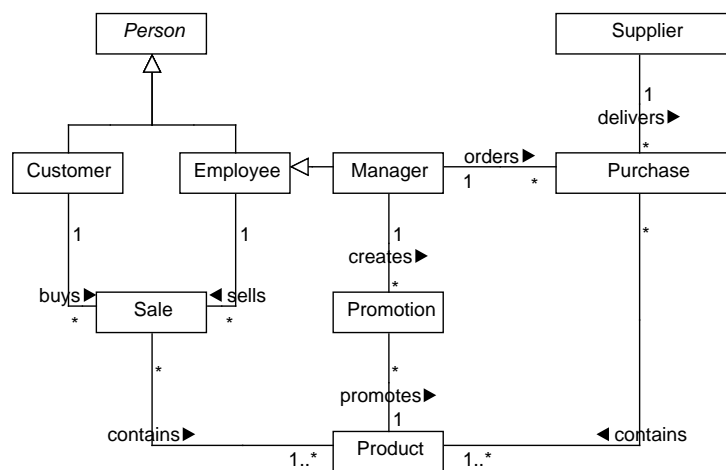


Abbildung 3.2.: Zweiter Entwurf des UML-Modells für einen studentischen Getränkemarkt

In einem nächsten Schritt werden wir die Assoziationen zwischen *Purchase*, *Product* und *Sale* so aufbrechen, dass wir weitere Details zu den Einzelposten

### 3. Abbildung eines studentischen Getränkemarktes im Hybrid-Wiki-System Tricia

(*PurchaseLineItem* beziehungsweise *SaleLineItem*) hinzufügen können. So erhalten wir das Modell aus Abbildung 3.3.

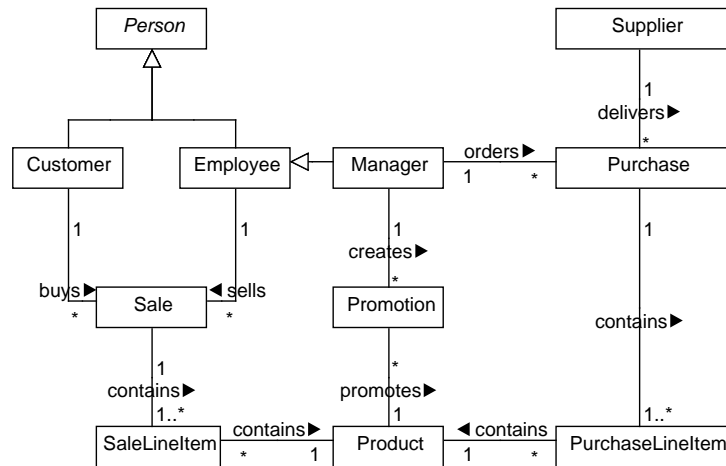


Abbildung 3.3.: Finaler Entwurf des UML-Modells für einen studentischen Getränkemarkt

Schließlich befüllen wir die Klassen mit Attributen und erhalten das fertige UML-Modell aus Abbildung 3.4. Alle Attribute werden noch ausführlich in Kapitel 5.1 erläutert, jedoch wollen wir hier folgende Attribute hervorheben:

- *Product.deposit*  
Pfand pro Produkteinheit.
- *Product.productType*  
Art des Produkts (als Enumeration).
- *PurchaseLineItem.lineDepositTax*  
Gesamtwert der Steuer auf den Pfandwert.

### 3. Abbildung eines studentischen Getränkemarktes im Hybrid-Wiki-System Tricia

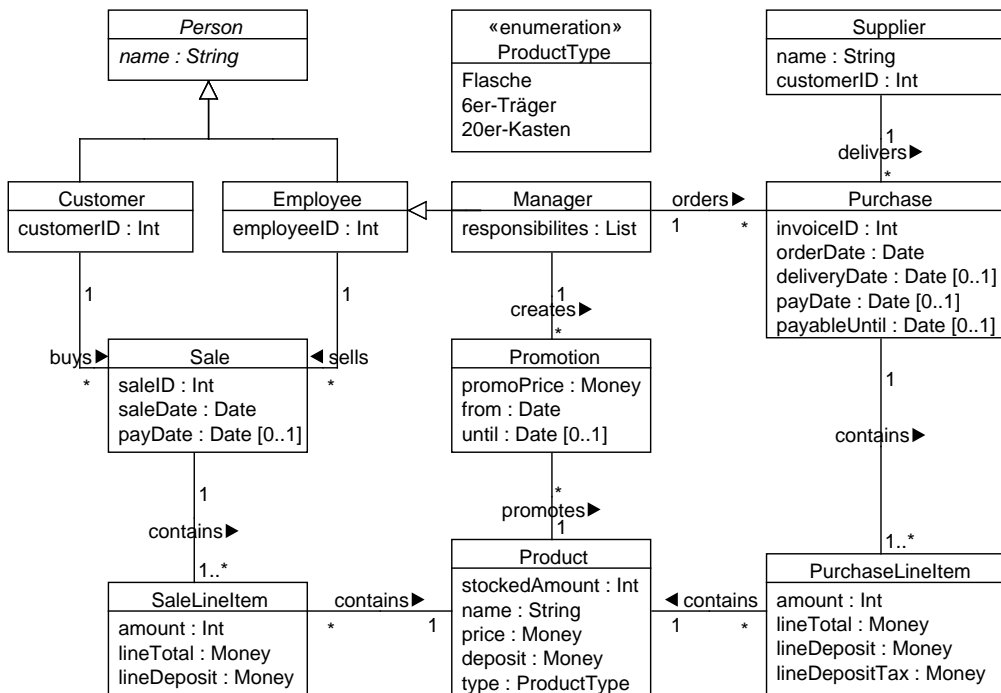


Abbildung 3.4.: Das fertige UML-Modell eines studentischen Getränkemarktes in der Studentenstadt Freimann

## 3.2. Umsetzung des Modells in Tricia

Um das Modell aus Abbildung 3.4 in Tricia umzusetzen, werden die Klassen als *Type Tags* implementiert. Diesen werden die entsprechenden Attribute hinzugefügt. Ebenso kann man die Assoziationen als Attribut in Tricia modellieren. Multiplizitäten werden dabei ebenfalls beachtet.



## 4. Ermittlung relevanter Kennzahlen

Um zu evaluieren, ob die für einen studentischen Getränkemarkt nötigen Funktionalitäten eines ERP-Systems auch in einem Hybrid Wiki implementiert werden können, muss vor allem herausgefunden werden, welche Funktionalitäten eines ERP-Systems überhaupt für einen studentischen Getränkemarkt nötig sind. In diesem Kapitel soll daher die Forschungsfrage F2 geklärt werden, indem die Kennzahlen definiert werden, welche in einem solchen System abgebildet werden müssen. Innerhalb dieser Arbeit ist unter *Kennzahl* eine betriebswirtschaftliche Kennzahl zu verstehen, welche nach [St67, Seite 62] wie folgt definiert ist:

Betriebswirtschaftliche Kennzahlen [...] sind Zahlen, die in konzentrierter Form über einen zahlenmäßig erfassbaren betriebswirtschaftlichen Sachverhalt informieren.

Darüber hinaus werden wir jedoch auch bestimmte Auflistungen und Verweise als Kennzahl in unserem Sinne werten. Dies trägt der Tatsache Rechnung, dass oftmals ein konkreter Verweis auf ein Element wichtiger ist, als eine bloße Aggregation dessen Eigenschaften.

Um die benötigten Kennzahlen herauszufinden, werden wir zuerst mittels einer Befragung der beteiligten Akteure alle Kennzahlen sammeln, welche für ein solches System in Frage kommen. Durch eine darauffolgende Online-Umfrage werden wir schließlich herausfinden, welche der ermittelten Kennzahlen die Akteure einsetzen würden.

### 4.1. Identifizierung möglicher Kennzahlen

Ein Ansatz zur Erhebung möglicher Kennzahlen ist die Durchführung von persönlichen *Face-to-face*-Interviews [Di07, Seite 373]. Der Vorteil an dieser Methode ist, dass der Interviewer den Befragten beim Ausfüllen des Fragebogens

unterstützen kann. Eine Beeinflussung des Befragten durch den Interviewer spielt hier nur eine untergeordnete Rolle, da dieses Interview rein quantitativ ist. Die Interviews führen wir mit den Experten auf diesem Gebiet, den Betreibern, Mitarbeitern und Prüfern der studentischen Getränkemarkte in der Studentenstadt Freimann, durch.<sup>9</sup>

##### 4.1.1. Erstellung des Fragebogens

Als Grundlage der Interviews dient uns ein Fragebogen. In diesem werden aus organisatorischen Gründen der Vor- und Nachname des Befragten, sowie dessen E-Mailadresse erfasst. Als weiteres persönliches Merkmal wird erfasst, zu welcher Gruppe von Akteuren (Betreiber, Mitarbeiter, Prüfer) und zu welchem Markt der Befragte gehört. Der Hauptteil der Befragung besteht aus drei Fragen. Zunächst wird die Frage gestellt, ob der Interviewte in seiner Funktion bereits Kennzahlen verwendet und falls ja, welche dies sind. Danach wird nach weiteren Kennzahlen gefragt, welche der Befragte in seiner Funktion gerne einsetzen würde. Schließlich wird ihm ein Kennzahlenkatalog vorgelegt, welcher mittels eines vorangegangenen Interviews mit dem Betreiber des HSH-Getränkeverkaufs erstellt wurde. Diesen soll der Befragte gegebenenfalls um weitere Kennzahlen ergänzen. Abschließend wird ermittelt, ob der Interviewpartner sich auch am zweiten Teil, der Online-Umfrage (siehe Kapitel 4.2), beteiligen würde.

Der Fragebogen mit zugehörigem Kennzahlenkatalog ist im Anhang dieser Arbeit zu finden (siehe A.3).

##### 4.1.2. Ergebnisse

Hauptansprechpartner für die einzelnen Märkte sind die Betreiber. Kontaktdaten von diesen stehen meist auf Listen, welche in den entsprechenden Häusern aushängen. Die Aktualität dieser ist jedoch nicht immer gegeben. An Kontaktdaten von Mitarbeitern oder Prüfern zu kommen ist nur über die Betreiber möglich. Wir haben von 19 Personen eine Rückmeldung bekommen. Von diesen 19 Personen waren auch alle bereit am Interview teilzunehmen. Jedoch konnte das Interview innerhalb einer Woche nur mit vier Personen durchgeführt

---

<sup>9</sup>Vgl. [Ka02].

#### 4. Ermittlung relevanter Kennzahlen

---

werden. Die anderen Personen nahmen nicht teil, da sie entweder aufgrund der vorlesungsfreien Zeit im Urlaub waren oder aus sonstigen Gründen keinen Termin wahrnehmen konnten. Bis zu Beginn der Vorlesungszeit konnten jedoch insgesamt 13 Akteure interviewt werden. Dies entspricht etwa der Hälfte der geschätzten 30 an den studentischen Getränkemarkten in der Studentenstadt Freimann beteiligten Akteure.

In der Ergebnistabelle im Anhang A.4 werden die Märkte in anonymisierter Form von den Buchstaben A bis E repräsentiert. Einige der Aussagen lassen jedoch direkte Rückschlüsse auf den Markt zu, weshalb bei diesen der Markt explizit genannt wird. Im Speziellen sind dies der Betreiber des HSH-Getränkverkaufs, mit welchem die Initialbefragung durchgeführt wurde, ein Mitarbeiter des Brotladens und ein Prüfer des HSH-Getränkverkaufs. Von den Befragten waren alle bereit an der Folgebefragung teilzunehmen.

Des Weiteren ist während der Befragung aufgefallen, dass viele der befragten Akteure nicht wussten was Kennzahlen sind oder sich dabei zumindest nicht sicher waren und den Interviewer um eine Erklärung baten (zur Erklärung dieser siehe auch Seite 17). Teilweise war den Befragten nicht bewusst, warum sie Kennzahlen überhaupt einsetzen sollten. Dies kann entweder daran liegen, dass sie die Kennzahlen für unwichtig halten oder daran, dass sie von einem möglicherweise erhöhtem Arbeitsaufwand abgeschreckt sind.

Einige Kennzahlen wurden von mehreren Personen genannt, jedoch unterschiedlich ausgedrückt. Diese werden im weiteren Verlauf zu einer einzigen Kennzahl zusammengefasst. Des Weiteren werden die Kennzahlen zur besseren Übersichtlichkeit in sechs Kategorien eingeteilt. Schließlich haben wir noch zehn weitere Kennzahlen der Liste hinzugefügt, bei welchen wir annotiert haben, aus welcher Quelle die Kennzahl stammt. Viele der in der Literatur angegebenen Kennzahlen sind nicht auf die studentischen Getränkemarkte anwendbar, da sich diese oftmals auf Werte beziehen, welche in einem studentischen Getränkemarkt keine Rolle spielen. Beispiele hierfür wären *Fremdkapital* und *Abschreibungen*.

Die durch die Befragung ermittelten Kennzahlen sind:

- **Mitarbeiter**

Kennzahlen, welche pro Mitarbeiter erhoben werden.

K0 Ist-Arbeitszeit

#### 4. Ermittlung relevanter Kennzahlen

---

- K1 Soll-Arbeitszeit
- K2 Durchschnittliche Arbeitszeit
- K3 Ausstehendes Gehalt
- K4 Tatsächlicher Stundenlohn
- K5 Umsatz pro Monat
- K6 Umsatzbeteiligung pro Monat
- K7 Wert der Selbsteinkäufe pro Monat

- **Barkunden**

Kennzahlen, welche sich auf die Anzahl an Barkunden und bestimmte Kundengruppen beziehen.

- K8 Anzahl Kunden pro Monat
- K9 Anzahl Kunden pro Tageszeit
- K10 Anzahl Kunden pro Wochentag
- K11 Anzahl Verkäufe pro Monat pro Kundengruppe
- K12 Durchschnittlicher Umsatz pro Kunde pro Kundengruppe

- **Rechnungskunden**

Kennzahlen, welche pro Rechnungskunde erhoben werden.

- K13 Aktuelle Schulden
- K14 Ältteste ausstehende Rechnung
- K15 Durchschnittliche Dauer, bis eine Rechnung bezahlt wird
- K16 Umsatz

- **Einkauf**

Kennzahlen, welche sich auf Bestellungen, Lieferungen und Verfügbarkeiten von Artikeln, sowie den Einkaufswert beziehen.

- K17 Liste der ausstehenden Lieferungen
- K18 Durchschnittliche Anzahl an Artikeln pro Bestellung
- K19 Durchschnittliche Anzahl an Pfand zurück pro Bestellung
- K20 Durchschnittliche Anzahl Bestellungen pro Monat

#### 4. Ermittlung relevanter Kennzahlen

---

- K21 Durchschnittliche Anzahl Bestellungen pro Wochentag
- K22 Durchschnittliche Dauer, bis Artikel wieder verfügbar, pro Artikel
- K23 Durchschnittliche Verfügbarkeit pro Artikel
- K24 Durchschnittlicher Wert des Pfands zurück pro Bestellung
- K25 Durchschnittlicher Warenwert pro Bestellung
- K26 Liste der zu bezahlenden Lieferungen
- K27 Einkaufsvolumenquote =  $\frac{\text{Einkaufsvolumen}}{\text{Umsatz}}$  [Ko03, Seite 388]

- **Artikel und Lager**

Kennzahlen, welche sich auf Artikel, deren Eigenschaften, Wert und Kundenbestellungen beziehen.

- K28 Aktueller Bestand pro Artikel
- K29 Aktueller Gesamtwert der Waren
- K30 Liste von Kunden nachgefragter Artikel (nicht im Sortiment)
- K31 Anzahl fehlender Artikel im Verkaufsregal pro Artikel
- K32 Anzahl MHD-Überschreitungen pro Artikel
- K33 Anzahl verkaufter Artikel pro Artikel pro Monat
- K34 Liste von Artikeln, welche oft zusammen gekauft werden
- K35 Differenz zwischen bestellter und gelieferter Menge pro Artikel
- K36 Durchschnittliche Dauer, bis Artikel ausverkauft ist (*Drehzahl*)
- K37 Durchschnittliche Dauer, bis eine Artikeleinheit verkauft wird
- K38 Einkaufspreis pro Artikel
- K39 Handelsspanne pro Artikel = VK-Preis – EK-Preis
- K40 Liste bald verderbender Waren
- K41 Liste der Vorbestellungen
- K42 Schwund pro Artikel pro Monat
- K43 Umsatz pro Artikel
- K44 Verkaufspreis pro Artikel

#### 4. Ermittlung relevanter Kennzahlen

---

K45 Verkaufte Artikel pro Tageszeit

K46 Verkaufte Artikel pro Wochentag

K47 Durchschnittlicher Bestand pro Artikel [Me11, Seite 95]

K48 Mindestbestand pro Artikel [Me11, Seite 96]

K49 Bestellpunkt pro Artikel [Me11, Seite 96]

K50 Lagerstruktur =  $\frac{\text{Lagerbestand}}{\text{Umsatz}}$  [Me11, Seite 96f]

K51 Lagerumschlag =  $\frac{\text{Verkaufte Artikel pro Monat}}{\varnothing \text{Bestand pro Artikel}}$  [Me11, Seite 98]

- **Finanzen**

Kennzahlen, welche sich auf die Finanzen beziehen.

K52 Summe der aktuell ausstehenden Forderungen

K53 Summe der aktuell ausstehenden Verbindlichkeiten

K54 Aktueller Ist-Kassenstand

K55 Aktueller Soll-Kassenstand

K56 Aktueller Wert der Betriebsstoffe (Verbrauchsmaterial)

K57 Aktueller Wert der Geschäftsausstattung

K58 Anzahl Abschöpfungen pro Monat

K59 Ausgaben pro Kategorie pro Monat

K60 Bareinnahmen pro Monat

K61 Bareinnahmen pro Tageszeit

K62 Bareinnahmen pro Wochentag

K63 Durchschnittlicher Abschöpfungsbetrag

K64 Durchschnittlicher Abstand zwischen zwei Abschöpfungen

K65 Einnahmen pro Kategorie pro Monat

K66 Gewinn pro Monat

K67 Münzeinlage/-entnahme pro Monat

K68 Summe der Rabattbeträge pro Monat

K69 Umsatz pro Monat

K70 Umsatz pro Tageszeit

K71 Umsatz pro vordefinierter Wetterkategorie

K72 Umsatz pro Wochentag

K73 Umsatzrentabilität =  $\frac{\text{Gewinn}}{\text{Umsatz}}$  [Me11, Seite 135]

K74 Mindestumsatz =  $\frac{\text{Fixkosten}}{\text{Handelsspanne}}$  [Me11, Seite 112f]

K75 Gesamtkapitalumschlag =  $\frac{\text{Umsatz}}{\text{Gesamtkapital}}$  [Me11, Seite 125]

K76 Vermögen pro Vermögensart [Me11, Seite 120f]

Durchschnittlich werden von den befragten Akteuren bereits 3,6 Kennzahlen benutzt<sup>10</sup>. Die Befragten würden noch im Schnitt 2,25 weitere Kennzahlen erheben<sup>11</sup>. Hervorzuheben wäre ansonsten, dass acht der 13 Befragten maximal drei Kennzahlen einsetzen und sich unter den Akteuren, welche fünf oder mehr (bis zu neun) Kennzahlen einsetzen, nur Betreiber und Prüfer befinden. Mitarbeiter setzten zum Befragungszeitpunkt maximal drei Kennzahlen ein.

## 4.2. Priorisierung der Kennzahlen

In der Folgebefragung wollen wir die durch die Interviews identifizierten Kennzahlen nach ihrer Relevanz beurteilen. Dazu erstellen wir eine Online-Umfrage, an der die selben Akteure wie bei den Interviews teilnehmen. Diese sollen hier ihre Präferenzen bezüglich der gesammelten Kennzahlen mitteilen.

### 4.2.1. Erstellung der Online-Umfrage

Unsere Online-Umfrage wurde mit der Software LimeSurvey (siehe [Lim]) erstellt. Sie besteht aus zwei Fragengruppen. In der ersten Gruppe werden nochmals Name und E-Mailadresse erfasst, um die Teilnehmer bei eventuell auftauchenden Problemen kontaktieren zu können. Diese personenbezogenen Daten werden jedoch nicht veröffentlicht. Daraufhin werden den Akteuren in der

---

<sup>10</sup>Mittelwert aus der Anzahl an Antworten auf Frage 1.

<sup>11</sup>Mittelwert der Summe der Anzahlen der Antworten auf Fragen 2 und 3. Der Betreiber des HSH-Getränkeverkaufs ist hiervon ausgenommen.

zweiten Fragegruppe alle identifizierten Kennzahlen in ihren Kategorien präsentiert. Die Akteure können bei jeder Kennzahl angeben, ob sie diese einsetzen würden oder nicht. Ebenso haben sie die Möglichkeit keine der beiden Antworten auszuwählen, was der Aussage *Keine Angabe* entspricht.

Die bereits interviewten Akteure werden mittels E-Mail an die erfasste E-Mailadresse eingeladen an der Umfrage teilzunehmen. Sie haben eine Woche Zeit, um die Umfrage auszufüllen.

### 4.2.2. Ergebnisse

Von den 13 Interviewteilnehmern haben elf an der Online-Umfrage teilgenommen. Dies entspricht einer Rückmeldequote von 85%. Die genauen Ergebnisse sind in Anhang A.5 nachzulesen.

Zusammenfassend kann man sagen, dass die Akteure der studentischen Getränkemarkte viele Kennzahlen verwenden würden. So würde jede der identifizierten Kennzahlen von mindestens einem Akteur eingesetzt werden (siehe Abbildung 4.1). 96% der Kennzahlen würden von mindestens drei Akteuren (27%) eingesetzt werden; 82% der Kennzahlen von mindestens fünf Akteuren (45%). Von den Betreibern werden mehr Kennzahlen gewünscht. Hier würden 95% der Kennzahlen von mindestens zwei Betreibern (40%) und 58% der Kennzahlen von mindestens vier Betreibern (80%) eingesetzt werden. Der Unterschied zwischen den Rollen der Akteure ist auch in Abbildung 4.2 zu erkennen.

Die beiden Abbildungen 4.1 und 4.2 sind wie folgt aufgebaut: Die Kennzahlen sind in der Reihenfolge ihres Kürzels angeordnet. Ein negativer Wert (rot) stellt die Anzahl aller Akteure dar, welche die jeweilige Kennzahl nicht einsetzen würden, ein positiver Wert (grün) die Anzahl der Akteure, welche die Kennzahl einsetzen würden. Hat ein Akteur keine Angabe zu einer Kennzahl gemacht, wurde dieser in der Abbildung bei der entsprechenden Kennzahl nicht berücksichtigt. Es sei bei Abbildung 4.2 noch angemerkt, dass fünf Betreiber, fünf Mitarbeiter und ein Prüfer befragt wurden.

Interessant wäre es noch anzumerken, dass die Metriken, welche sich auf die Tageszeit beziehen, hauptsächlich (aber nicht ausschließlich) von Akteuren des Brotladens gewünscht werden. Von den Akteuren anderer Getränkemarkte,



#### *4. Ermittlung relevanter Kennzahlen*

---

welche die Kennzahl einsetzen würden, wird möglicherweise eine genauere Auswertung (beispielsweise auf 15, 10 oder 5 Minuten genau) gewünscht.

Im weiteren Verlauf dieser Arbeit werden wir uns auf die meist gewünschten Kennzahlen beschränken. Dazu erstellen wir innerhalb jeder Kategorie anhand der Anzahl der Akteure, welche die Kennzahl einsetzen würden, eine Rangliste der Kennzahlen. Ausgewählt werden nun pro Kategorie die Kennzahlen auf den ersten drei Rängen. Dies führt dazu, dass wir pro Kategorie drei, bei gleicher Anzahl der die Kennzahl einsetzenden Akteure auch vier, Kennzahlen erhalten. Die 21 Kennzahlen, welche wir durch dieses Vorgehen erhalten, sind in Tabelle 4.1 aufgelistet. Die Reihenfolge der Kategorien in der Tabelle wurde entsprechend der Auflistung der Kennzahlen in Kategorie 4.1.2 gewählt. Innerhalb dieser Kategorien sind die Kennzahlen zuerst absteigend nach der Anzahl der Akteure, welche die Kennzahl einsetzen würden, sortiert und dann aufsteigend nach ihrem Kürzel.

#### 4. Ermittlung relevanter Kennzahlen

---

Anzahl der Akteure, welche die Kennzahl einsetzen würden		
Kategorie		[Kürzel] Kennzahl
Mitarbeiter	9	[K4] Tatsächlicher Stundenlohn
	9	[K5] Umsatz pro Monat
	8	[K0] Ist-Arbeitszeit
	8	[K6] Umsatzbeteiligung pro Monat
Bar- kunden	8	[K10] Anzahl Kunden pro Wochentag
	6	[K8] Anzahl Kunden pro Monat
	5	[K11] Anzahl Verkäufe pro Monat pro Kundengruppe
	5	[K12] øUmsatz pro Kunde pro Kundengruppe
Rechnungs- kunden	10	[K13] Aktuelle Schulden
	8	[K16] Umsatz
	7	[K14] Älteste ausstehende Rechnung
Einkauf	8	[K17] Liste der ausstehenden Lieferungen
	8	[K22] øDauer, bis Artikel wieder verfügbar, pro Artikel
	8	[K26] Liste der zu bezahlenden Lieferungen
Artikel und Lager	9	[K28] Aktueller Bestand pro Artikel
	9	[K36] Drehzahl
	9	[K40] Liste bald verderbender Waren
	9	[K51] Lagerumschlag
Finanzen	11	[K54] Aktueller Ist-Kassenstand
	11	[K55] Aktueller Soll-Kassenstand
	9	[K69] Umsatz pro Monat

Tabelle 4.1.: Die 21 Kennzahlen, welche im weiteren Verlauf dieser Arbeit verwendet werden.

#### 4. Ermittlung relevanter Kennzahlen



Abbildung 4.1.: Auswertung der Online-Umfrage.

#### 4. Ermittlung relevanter Kennzahlen



Abbildung 4.2.: Auswertung der Online-Umfrage, aufgeschlüsselt nach Gruppen von Akteuren.

## 5. Verortung und Implementierung der Kennzahlen im Hybrid Wiki

Schließlich wollen wir uns in diesem Kapitel der Forschungsfrage F3 widmen und das bereits erarbeitete Klassendiagramm aus Abbildung 3.4 so erweitern, dass die 21 Kennzahlen aus Tabelle 4.1 im resultierenden System erhoben werden können. Zur Darstellung der nötigen Bedingungen werden wir die OCL<sup>12</sup> verwenden. Diese bietet sich dadurch an, dass sie die UML um die formale Möglichkeit erweitert, Invarianten, sowie Vor- und Nachbedingungen für einzelne Methoden (sogenannte *Constraints*) vorzugeben.

### 5.1. Erweiterung des Modells

Um die ausgewählten Kennzahlen in unser bereits erstelltes Modell einbetten zu können, müssen wir die vorhandenen Klassen erweitern, sowie weitere Klassen hinzufügen. Nur so können wir die für die Kennzahlen nötigen Werte berechnen und speichern. Im Folgenden werden wir alle Attribute, Assoziationen und Methoden einer jeden Klasse des erweiterten Modells aus Abbildung 5.1 genau erläutern. Besonders hervorzuheben ist hierbei die Angabe der Multiplizität hinter einer Operation<sup>13</sup>. Ist keine Multiplizität angegeben, ist diese bei Attributen und Methoden [1] und bei Assoziationen [\*]. Ebenso werden wir gegebenenfalls für diese Klassen OCL-Ausdrücke definieren, welche für die Implementierung und Erhebung der Kennzahlen wichtig sind. Die Ausdrücke werden anhand von [WK99] und [Oe09, Kapitel 4.9] erstellt und so angepasst, dass sie durch die Software *Papyrus*<sup>14</sup> validiert werden können. Dazu werden wir die einzelnen Klassen in alphabetischer Reihenfolge betrachten.

---

<sup>12</sup>Object Constraint Language. Spezifikation verfügbar auf [OCL].

<sup>13</sup>Definiert in der *Superstructure*-Spezifikation von UML 2.4.1 unter Punkt 7.3.37. Spezifikation verfügbar auf [UML].

<sup>14</sup>Version 1.12. Siehe [Pap].

## 5. Verortung und Implementierung der Kennzahlen im Hybrid Wiki

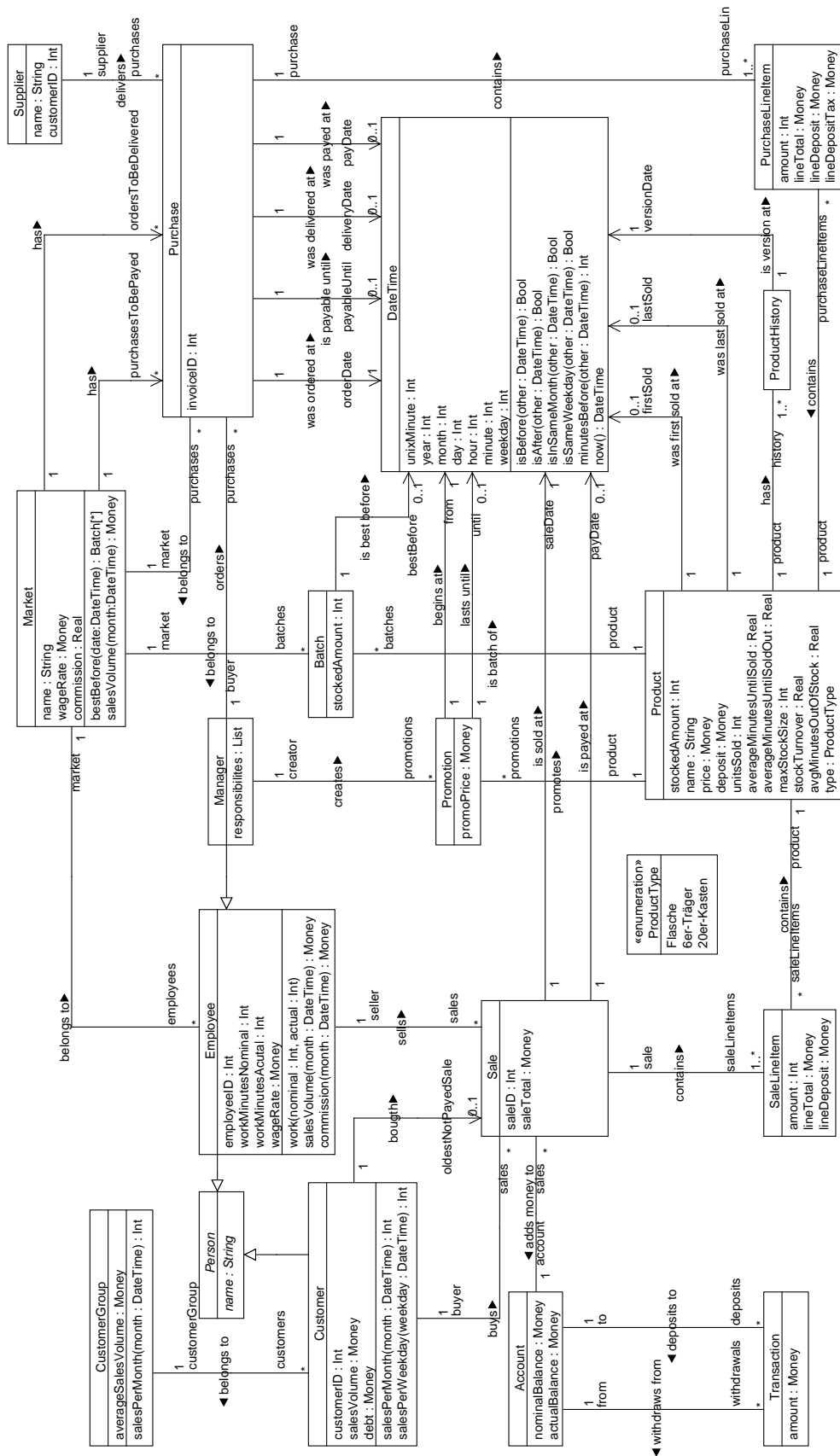


Abbildung 5.1.: Das erweiterte UML-Modell

## Die Klasse *Account*

Die Klasse *Account* stellt die buchhalterischen Konten dar.

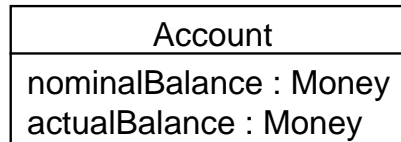


Abbildung 5.2.: Die Klasse *Account*

### Attribute

- `nominalBalance:Money`  
Soll-Stand des Kontos.
- `actualBalance:Money`  
Ist-Stand des Kontos.

### Assoziationen

- `withdrawals:Transaction`  
Transaktionen, welche von diesem Konto abbuchen.
- `deposits:Transaction`  
Transaktionen, welche auf dieses Konto buchen.
- `sales:Sale`  
Verkäufe, welche auf dieses Konto gebucht werden.

Listing 5.1: Constraints der Klasse *Account*

```
1 context Account inv:  
2   "actualBalance entspricht dem aktuellen Ist-Kassenstand (wird manuell eingegeben)."  
3  
4 context Account inv:  
5   self .nominalBalance=self.sales.saleTotal->sum()+self.deposits.amount->sum()-self.  
   withdrawals.amount->sum()
```

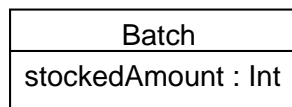


Abbildung 5.3.: Die Klasse *Batch*

## Die Klasse *Batch*

Die Klasse *Batch* spezifiziert einzelne Chargen eines Produktes (*product*).

### Attribute

- stockedAmount : Int  
Anzahl der Artikel auf Lager.

### Assoziationen

- bestBefore : DateTime [0..1]  
Ablaufdatum der Charge.
- market : Market [1]  
Markt, zu dem die Charge gehört.
- product : Product [1]  
Produkt, um dessen Charge es sich handelt.

## Die Klasse *Customer*

Die Klasse *Customer* leitet sich von der Klasse *Person* ab und repräsentiert einen Kunden.

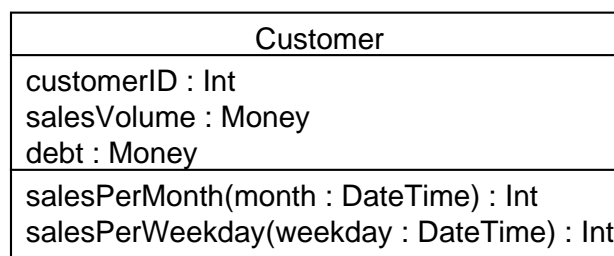


Abbildung 5.4.: Die Klasse *Customer*

### Attribute

- customerID : Int  
ID um den Kunden zu identifizieren.



- `salesVolume:Money`  
Gesamtumsatz des Kunden.
- `debt:Money`  
Summe der ausstehenden Beträge für nicht bezahlte Rechnungen.

### Assoziationen

- `sales:Sale`  
Verkaufsvorgänge, bei denen an diesen Kunden verkauft wurde.
- `oldestNotPayedSale:Sale[0..1]`  
Ältester Verkaufsvorgang, bei dem an diesen Kunden verkauft wurde, dieser aber noch nicht bezahlt hat.
- `customerGroup:CustomerGroup`  
Kundengruppe, welcher der Kunde angehört.

### Methoden

- `salesPerMonth(month:DateTime):Int`  
Anzahl der Verkaufsvorgänge, bei denen an diesen Kunden verkauft wurde, im Monat *month*.
- `salesPerWeekday(weekday:DateTime):Int`  
Anzahl der Verkaufsvorgänge, bei denen an diesen Kunden verkauft wurde, welche am Wochentag *weekday* stattfanden.

Listing 5.2: Constraints der Klasse *Customer*

```
1 context Customer::salesPerMonth(month:DateTime):Integer
2   post: result =self.sales->select(saleDate.isInSameMonth(month))->size()
3
4 context Customer::salesPerWeekday(weekday:DateTime):Integer
5   post: result =self.sales->select(saleDate.isSameWeekday(weekday))->size()
6
7 context Customer inv:
8   self.debt=self.sales->select(payDate->isEmpty()).saleTotal->sum()
9 and
10  self.oldestNotPayedSale=self.sales->select(payDate->isEmpty())->iterate(
11    s;
12    oldestNotPayedSale:Sequence(Sale)=Sequence{ }
13    if
14      oldestNotPayedSale->isEmpty()
15    or
```

```

16     oldestNotPayedSale->first().saleDate.isAfter(s.saleDate)
17     then
18         Sequence{s}
19     else
20         oldestNotPayedSale
21     endif
22     )->first()
23 and
24     self.salesVolume=self.sales.saleTotal->sum()

```

## Die Klasse *CustomerGroup*

Die Klasse *CustomerGroup* repräsentiert bestimmte Kundengruppen.

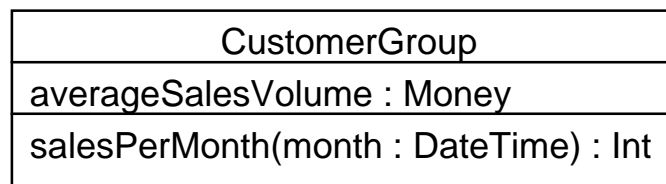


Abbildung 5.5.: Die Klasse *CustomerGroup*

### Attribute

- **averageSalesVolume**:Money  
Durchschnittlicher Umsatz der Kundengruppe.

### Assoziationen

- **customers**:Customer  
Kunden in der Kundengruppe.

### Methoden

- **salesperMonth(month:DateTime) : Int**  
Umsatz der Kundengruppe im Monat *month*.

Listing 5.3: Constraints der Klasse *CustomerGroup*

```

1 context CustomerGroup:salesPerMonth(month:DateTime):Integer
2     post: result=self.customers.salesPerMonth(month)->sum()
3
4 context CustomerGroup inv:
5     self.averageSalesVolume=self.customers.salesVolume->sum()/self.customers->size()

```

## Die Klasse *DateTime*

Da OCL keine Variablentypen zur Speicherung einer Zeit oder eines Datums hat, fügen wir, zur Repräsentation dieser, dem Modell die Klasse *DateTime* hinzu.

DateTime
unixMinute : Int year : Int month : Int day : Int hour : Int minute : Int weekday : Int
isBefore(other : DateTime) : Bool isAfter(other : DateTime) : Bool isInSameMonth(other : DateTime) : Bool isSameWeekday(other : DateTime) : Bool minutesBefore(other : DateTime) : Int now() : DateTime

Abbildung 5.6.: Die Klasse *DateTime*

### Attribute

- **unixMinute: Int**  
Vergangene Minuten seit dem 1. Januar 1970 00:00 Uhr UTC<sup>15</sup>.
- **year: Int, month: Int, day: Int, hour: Int und minute: Int**  
Jahr, Monat, Tag, Stunde und Minute des repräsentierten Zeitpunktes.
- **weekday: Int**  
Durch die Zahlen 0-7 repräsentierter Wochentag des Zeitpunktes.

### Methoden

- **isBefore(other: DateTime) : Bool**  
Gibt zurück, ob sich der durch das Objekt repräsentierte Zeitpunkt vor dem durch *other* repräsentierten Zeitpunkt befindet.
- **isAfter(other: DateTime) : Bool**  
Gibt zurück, ob sich der durch das Objekt repräsentierte Zeitpunkt nach dem durch *other* repräsentierten Zeitpunkt befindet.

---

<sup>15</sup>Universal Coordinated Time

- `isInSameMonth(other:DateTime): Boolean`  
Gibt zurück, ob sich der durch das Objekt repräsentierte Zeitpunkt und der durch *other* repräsentierte Zeitpunkt im selben Monat befinden.
- `isSameWeekday(other:DateTime): Boolean`  
Gibt zurück, ob der durch das Objekt repräsentierte Zeitpunkt am gleichen Wochentag ist wie der durch *other* repräsentierte Zeitpunkt.
- `minutesBefore(other:DateTime): Int`  
Gibt zurück, wie viele Minuten sich der durch das Objekt repräsentierte Zeitpunkt vor dem durch *other* repräsentierten Zeitpunkt befindet.
- `now(): DateTime`  
Gibt ein neues *DateTime*-Objekt zurück, welches den aktuellen Zeitpunkt repräsentiert.

Listing 5.4: Constraints der Klasse *DateTime*

```

1 context DateTime::isBefore(other:DateTime):Boolean
2   post: result = self.year < other.year or (self.year = other.year and (self.month < other.
3     month or (self.month = other.month and (self.day < other.day or (self.day = other.day
4     and (self.hour < other.hour or (self.hour = other.hour and (self.minute < other.
5     minute))))))
6
7 context DateTime::isAfter(other:DateTime):Boolean
8   post: result = self.year > other.year or (self.year = other.year and (self.month > other.
9     month or (self.month = other.month and (self.day > other.day or (self.day = other.day
10    and (self.hour > other.hour or (self.hour = other.hour and (self.minute > other.
11    minute))))))
12
13 context DateTime::isInSameMonth(other:DateTime):Boolean
14   post: result = (self.year = other.year) and (self.month = other.month)
15
16 context DateTime::isSameWeekday(other:DateTime):Boolean
17   post: result = (self.weekday = other.weekday)
18
19 context DateTime::minutesBefore(other:DateTime):Integer
20   post: result = other.unixMinute - self.unixMinute
21
22 context DateTime::now():DateTime
23   post: "Ein neues DateTime mit aktuellen Daten wird zurückgegeben."
24
25 context DateTime inv:
26 (

```

```
21 self . weekday=let
22   t : Sequence(Integer)=Sequence{0,3,2,5,0,3,5,1,4,6,2,4}
23 in
24   let
25     year : Integer=if self.month<3 then self.year-1 else self.year endif
26   in
27     (year+year.div(4)-year.div(100)+year.div(400)+t->at(self.month)+self.day).mod
28     (7)
29 )and(
30   self . unixMinute=let
31     daysToBeginOfMonth : Sequence(Integer)=Sequence
32     {0,31,59,90,120,151,181,212,243,273,304,334}
33   in
34     let
35       years : Integer=self.year-1970
36     in
37       let
38         leapYears : Integer=((self.year-1)-1968).div(4)-((self.year-1)-1900).div(100)
39         +((self.year-1)-1600).div(400)
40       in
41         let
42           unixMinute : Integer=self.minute+60*self.hour+(daysToBeginOfMonth->at(
43             self.month)+self.day-1)*60*24+(years*365+leapYears)*60*24
44         in
45           if self . month>2 and self.year.mod(4)=0 and (self.year.mod(100)<>0 or self.
46             year.mod(400)=0) then unixMinute+60*24 else unixMinute endif
47         )
48       )
49     )
50   )
51 )
```

Der Algorithmus zur Berechnung des Wochentages basiert hierbei auf dem unter [To93] nachzulesenden Algorithmus von T. Sakamoto.

## Die Klasse *Employee*

Die Klasse *Employee* leitet sich von der Klasse *Person* ab und repräsentiert einen Mitarbeiter.

### Attribute

- **employeeID: Int**  
ID, welche den Mitarbeiter identifiziert.

Employee
employeeID : Int workMinutesNominal : Int workMinutesActual : Int wageRate : Money
work(nominal : Int, actual : Int) salesVolume(month : DateTime) : Money commission(month : DateTime) : Money

Abbildung 5.7.: Die Klasse *Employee*

- **workMinutesNominal : Int**  
Soll-Arbeitszeit des Mitarbeiters.
- **workMinutesActual : Int**  
Ist-Arbeitszeit des Mitarbeiters.
- **wageRate : Money**  
Tatsächlicher Stundenlohn des Mitarbeiters.

### Assoziationen

- **market : Market [1]**  
Der Markt, in dem der Mitarbeiter beschäftigt ist.
- **sales : Sale**  
Verkaufsvorgänge, welche der Mitarbeiter durchgeführt hat.

### Methoden

- **work(nominal : Int, actual : Int)**  
Diese Methode soll aufgerufen werden, wenn ein Mitarbeiter arbeitet. Durch sie werden die Attribute *workMinutesNominal* und *workMinutesActual* um die Werte *nominal* beziehungsweise *actual* erhöht.
- **salesVolume(month : DateTime) : Money**  
Summe der Gesamtwerte der Verkaufsvorgänge, welche der Mitarbeiter im Monat *month* durchgeführt hat.
- **commission(month : DateTime) : Money**  
Umsatzbeteiligung, welche dem Mitarbeiter im Monat *month* zusteht.

Listing 5.5: Constraints der Klasse *Employee*

```
1 context Employee::work(nominal:Integer, actual:Integer)
2   post:
3     self . workMinutesNominal=self.workMinutesNominal@pre+nominal
4   and
5     self . workMinutesActual=self.workMinutesActual@pre+actual
6
7 context Employee::salesVolume(month:DateTime):Real
8   post: result =self . sales ->select (saleDate.isInSameMonth(month)).saleTotal->sum()
9
10 context Employee::commission(month:DateTime):Real
11   post: result =self . salesVolume(month)*market.commission
12
13 context Employee inv:
14   self . wageRate=self.sales.saleTotal->sum()*market.commission/self.
    workMinutesNominal+market.wageRate
```

## Die Klasse *Manager*

Die Klasse *Manager* leitet sich von der Klasse *Employee* ab und stellt einen Betreiber eines Marktes dar.

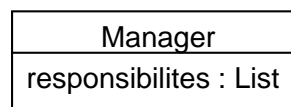


Abbildung 5.8.: Die Klasse *Manager*

### Attribute

- `responsibilites:List`  
Liste der Verantwortlichkeiten.

### Assoziationen

- `purchases:Purchase`  
Einkaufsvorgänge, an welchen der Betreiber beteiligt war.
- `promotions:Promotion`  
Preisreduzierungen, welche der Betreiber erstellt hat.

## Die Klasse *Market*

Die Klasse *Market* stellt den Markt als Gesamtes dar.

Market
name : String wageRate : Money commission : Real
bestBefore(date:DateTime) : Batch[*] salesVolume(month:DateTime) : Money

Abbildung 5.9.: Die Klasse *Market*

### Attribute

- `name:String`  
Name des Getränkemarktes.
- `wageRate:Money`  
Grundstundenlohn für alle Mitarbeiter des Marktes.
- `commission:Real`  
Prozentsatz der Umsatzbeteiligung der Mitarbeiter.

### Assoziationen

- `employees:Employee`  
Mitarbeiter des Marktes.
- `batches:Batch`  
Chargen auf Lager.
- `purchases:Purchase`  
Einkäufe des Marktes.
- `purchasesToBePayed:Purchase`  
Unbezahlte Lieferungen.
- `ordersToBeDelivered:Purchase`  
Noch nicht gelieferte Bestellungen.



## Methoden

- `bestBefore(date:DateTime):Batch[*]`  
Gibt die Chargen zurück, welche vor dem mittels *date* angegebenen Datum ablaufen.
- `salesVolume(month:DateTime):Money`  
Gibt zurück, wie viel Geld im Monat *month* umgesetzt wurde.

Listing 5.6: Constraints der Klasse *Market*

```
1 context Market::bestBefore(date:DateTime)
2   post: result=self.batches->select(bestBefore->notEmpty() and bestBefore.isBefore(
3     date))->asSet()
4
5 context Market::salesVolume(month:DateTime):Real
6   post: result=self.employees.salesVolume(month)->sum()
7
8 context Market inv:
9   self.ordersToBeDelivered=self.purchases->select(deliveryDate->isEmpty())
10  and
11  self.purchasesToBePaid=self.purchases->select(payDate->isEmpty())
```

## Die Klasse *Person*

Die Klasse *Person* dient als abstrakte Oberklasse für alle Personen.

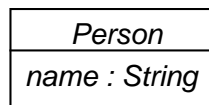


Abbildung 5.10.: Die Klasse *Person*

## Attribute

- `name:String`  
Der Name der Person.

## Die Klasse *Product*

Die Klasse *Product* repräsentiert ein Produkt.

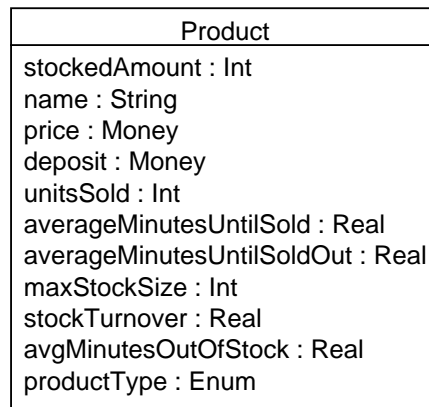


Abbildung 5.11.: Die Klasse *Product*

### Attribute

- **stockedAmount: Int**  
Anzahl an Produkteinheiten auf Lager.
- **name: String**  
Name des Produkts.
- **price: Money**  
Verkaufspreis einer Produkteinheit.
- **deposit: Money**  
Pfand pro Produkteinheit.
- **unitsSold: Int**  
Anzahl verkaufter Produkteinheiten.
- **averageMinutesUntilSold: Real**  
Durchschnittliche Dauer (in Minuten), bis eine Produkteinheit verkauft wurde.
- **averageMinutesUntilSoldOut: Real**  
Durchschnittliche Dauer (in Minuten), bis der maximal mögliche Lagerbestand verkauft wird.
- **maxStockSize: Int**  
Maximal möglicher Lagerbestand.
- **stockTurnover: Real**  
Lagerumschlag.

- `avgMinutesOutOfStock:Real`  
Durchschnittlicher Zeitraum (in Minuten), in welchem das Produkt ausverkauft war.
- `productType:Enum`  
Art des Produkts.

### Assoziationen

- `batches:Batch`  
Chargen dieses Produktes.
- `purchaseLineItems:PurchaseLineItem`  
Einkaufsposten, welche dieses Produkt beinhalten.
- `saleLineItems:SaleLineItem`  
Verkaufsposten, welche dieses Produkt beinhalten.
- `firstSold:DateTime[0..1]`  
Zeitpunkt, zu dem das Produkt zum ersten Mal verkauft wurde.
- `lastSold:DateTime[0..1]`  
Zeitpunkt, zu dem das Produkt zum letzten Mal verkauft wurde.
- `history:ProductHistory[1..*]`  
Aktuelle und ältere Versionen dieses Produkts.
- `promotion:Promotion`  
Preisreduzierungen für dieses Produkt.

Listing 5.7: Constraints der Klasse *Product*

```
1 context Product inv:
2   self .stockedAmount >= 0
3 and
4   self .history -> notEmpty()
5 and
6   not self .history -> exists(h:ProductHistory |
7     h.versionDate.isAfter(h.versionDate.now())
8   )
9 and
10  self .avgMinutesOutOfStock = (
11    let
12      soldOut:Bag(ProductHistory)=self.history->select(h1:ProductHistory |
13        h1.product.stockedAmount=0 and (
14          let
```

## 5. Verortung und Implementierung der Kennzahlen im Hybrid Wiki

---

```
15     latest : Sequence(ProductHistory)=self.history->iterate(
16         h2;
17         latest : Sequence(ProductHistory)=Sequence{}|
18         if
19             h2.versionDate.isBefore(h1.versionDate) and
20             (
21                 latest ->isEmpty() or
22                 h2.versionDate.isAfter ( latest ->first().versionDate)
23             )
24         then
25             Sequence{h2}
26         else
27             latest
28         endif
29     )
30     in
31         latest ->isEmpty() or latest->first().product.stockedAmount<>0
32     )
33 )
34 in
35     soldOut->iterate(
36         h1;
37         minutes:Integer=0|
38         let
39             firstAfter : Sequence(ProductHistory)=self.history->iterate(
40                 h2;
41                 firstAfter : Sequence(ProductHistory)=Sequence{}|
42                 if
43                     h2.versionDate.isAfter(h1.versionDate) and
44                     h2.product.stockedAmount>0 and
45                     (
46                         firstAfter ->isEmpty() or
47                         h2.versionDate.isBefore( firstAfter ->first().versionDate)
48                     )
49                 then
50                     Sequence{h2}
51                 else
52                     firstAfter
53                 endif
54             )
55         in
56             if
57                 firstAfter ->isEmpty()
58             then
```

## 5. Verortung und Implementierung der Kennzahlen im Hybrid Wiki

---

```
59         minutes+h1.versionDate.minutesBefore(h1.versionDate.now())
60     else
61         minutes+h1.versionDate.minutesBefore(firstAfter->first().versionDate)
62     endif
63 )/soldOut->size()
64 )
65 and
66 self.stockedAmount=self.purchaseLineItems.amount->sum()-self.saleLineItems.
    amount->sum()
67 and
68 self.unitsSold=self.saleLineItems.amount->sum()
69 and
70 self.firstSold=(self.saleLineItems->iterate(
71     sli ;
72     firstSold : Sequence(DateTime)=Sequence{|
73     if
74         firstSold->isEmpty()
75     then
76         Sequence{sli.sale.saleDate}
77     else
78         if
79             sli.sale.saleDate.isBefore( firstSold->first())
80         then
81             Sequence{sli.sale.saleDate}
82         else
83             firstSold
84         endif
85     endif
86 )->first()
87 and
88 self.lastSold=(self.saleLineItems->iterate(
89     sli ;
90     lastSold : Sequence(DateTime)=Sequence{|
91     if
92         lastSold->isEmpty()
93     then
94         Sequence{sli.sale.saleDate}
95     else
96         if
97             sli.sale.saleDate.isAfter( lastSold->first())
98         then
99             Sequence{sli.sale.saleDate}
100        else
101            lastSold
```

```
102     endif
103     endif
104   ))->first()
105 and
106   self .averageMinutesUntilSold=(self.lastSold.unixMinute-self.firstSold.unixMinute)/self
      .unitsSold
107 and
108   self .averageMinutesUntilSoldOut=self.averageMinutesUntilSold*self.maxStockSize
109 and
110   self .stockedAmount=self.batches.stockedAmount->sum()
111 and
112   self .stockTurnover=(365*24*60)/self.averageMinutesUntilSoldOut
```

Die Berechnung des Lagerumschlags erfolgt, mittels der umgestellten Gleichung aus [WKK96, S. 802], anhand der durchschnittlichen Dauer, bis der maximale Lagerbestand verkauft wurde.

## Die Klasse *ProductHistory*

Die Klasse *ProductHistory* repräsentiert die von Tricia bereitgestellte Versionsverwaltung der *Product*-Objekte.



ProductHistory

Abbildung 5.12.: Die Klasse *ProductHistory*

### Assoziationen

- `versionDate:DateTime`  
Zeitpunkt der Erstellung der Objektversion.
- `product:Product`  
Das historisierte Objekt.

## Die Klasse *Promotion*

Die Klasse *Promotion* repräsentiert eine Preisreduzierung.

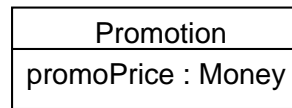


Abbildung 5.13.: Die Klasse *Promotion*

### Attribute

- `promoPrice:Money`  
Preis, zu dem das Produkt *product* im Aktionszeitraum angeboten wird.

### Assoziationen

- `from:DateTime[1]`  
Zeitpunkt, zu dem der Aktionszeitraum beginnt.
- `until:DateTime[0..1]`  
Zeitpunkt, zu dem der Aktionszeitraum endet.
- `product:Product[1]`  
Produkt, welches im Preis reduziert werden soll.
- `creator:Manager[1]`  
Betreiber, welcher die Preisreduzierung erstellt hat.

## Die Klasse *Purchase*

Die Klasse *Purchase* repräsentiert einen Einkaufsvorgang.

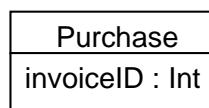


Abbildung 5.14.: Die Klasse *Purchase*

### Attribute

- `invoiceID:Int`  
ID, welche den Einkaufsvorgang identifiziert.

### Assoziationen

- `buyer:Manager[1]`  
Betreiber, welcher die Bestellung getätigt hat.

- `orderDate:DateTime[1]`  
Zeitpunkt der Bestellung.
- `payableUntil:DateTime[0..1]`  
Zahlungsfrist.
- `deliveryDate:DateTime[0..1]`  
Zeitpunkt der Lieferung.
- `payDate:DateTime[0..1]`  
Zeitpunkt der Zahlung.
- `supplier:Supplier[1]`  
Lieferant, bei welchem bestellt wurde.
- `market:Market[1]`  
Markt, zu welchem der Einkaufsvorgang gehört.
- `purchaseLineItems:PurchaseLineItem`  
Posten dieses Einkaufsvorganges.

## Die Klasse *PurchaseLineItem*

Die Klasse *PurchaseLineItem* stellt die einzelnen Posten eines Einkaufsvorgangs dar.

PurchaseLineItem
<code>amount : Int</code>
<code>lineTotal : Money</code>
<code>lineDeposit : Money</code>
<code>lineDepositTax : Money</code>

Abbildung 5.15.: Die Klasse *PurchaseLineItem*

### Attribute

- `amount: Int`  
Wie viele Produkteinheiten des Produkts *product* eingekauft werden.
- `lineTotal: Money`  
Gesamtwert dieses Postens.
- `lineDeposit: Money`  
Gesamtwert des Pfands für diesen Posten.



- `lineDepositTax:Money`  
Gesamtwert der Steuer auf den Pfandwert.

### Assoziationen

- `product:Product[1]`  
Produkt, welches eingekauft wird.
- `purchase:Purchase[1]`  
Einkaufsvorgang, zu dem der Posten gehört.

## Die Klasse *Sale*

Die Klasse *Sale* stellt einen Verkaufsvorgang dar.

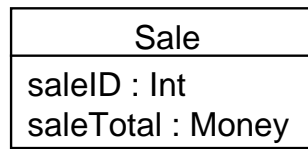


Abbildung 5.16.: Die Klasse *Sale*

### Attribute

- `saleID: Int`  
ID, welche den Verkaufsvorgang identifiziert.
- `saleTotal: Money`  
Gesamtwert des Verkaufsvorgangs.

### Assoziationen

- `saleDate:DateTime[1]`  
Zeitpunkt des Verkaufsvorgangs.
- `payDate:DateTime[0..1]`  
Zeitpunkt der Zahlung.
- `saleLineItems:SaleLineItem[1..*]`  
Die einzelnen Posten des Verkaufsvorgangs.
- `seller:Employee[1]`  
Mitarbeiter, welcher am Verkaufsvorgang als Verkäufer beteiligt ist.

- `buyer:Customer[1]`  
Kunde, welcher am Verkaufsvorgang als Käufer beteiligt ist.
- `account:Account[1]`  
Konto, auf das die Einnahmen gebucht werden.

Listing 5.8: Constraints der Klasse *Sale*

```
1 context Sale inv:  
2   self.saleTotal = self.saleLineItems.lineTotal->sum() + self.saleLineItems.lineDeposit->  
   sum()
```

## Die Klasse *SaleLineItem*

Die Klasse *SaleLineItem* stellt die einzelnen Posten eines Verkaufsvorgangs dar.

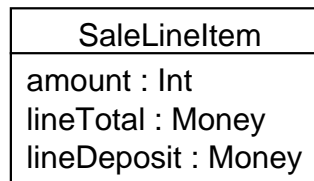


Abbildung 5.17.: Die Klasse *SaleLineItem*

### Attribute

- `amount: Int`  
Wie viele Produkteinheiten des Produkts *product* verkauft werden.
- `lineTotal: Money`  
Gesamtwert dieses Postens.
- `lineDeposit: Money`  
Gesamtwert des Pfands für diesen Posten.

### Assoziationen

- `sale: Sale[1]`  
Verkaufsvorgang, zu welchem dieser Posten gehört.
- `product: Product[1]`  
Produkt, welches verkauft wird.

## Die Klasse *Supplier*

Die Klasse *Supplier* repräsentiert einen Lieferanten.

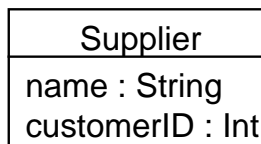


Abbildung 5.18.: Die Klasse *Supplier*

### Attribute

- `name: String`  
Name des Lieferanten.
- `customerID: Int`  
Kundennummer des Marktes bei diesem Lieferanten.

### Assoziationen

- `deliveries: Purchase`  
Einkaufsvorgänge, welche bei diesem Lieferanten durchgeführt wurden.

## Die Klasse *Transaction*

Die Klasse *Transaction* stellt die Buchungen zwischen den einzelnen Konten dar.

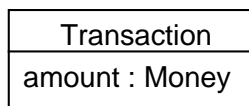


Abbildung 5.19.: Die Klasse *Transaction*

### Attribute

- `amount: Money`  
Geldbetrag, welcher transferiert wird.

### Assoziationen

- `from: Account[1]`  
Konto, von welchem der Betrag abgebucht wird.

- `to:Account[1]`  
Konto, auf das der Betrag gebucht wird.

Listing 5.9: Constraints der Klasse *Transaction*

```
1 context Transaction inv:  
2   self .from<>self.to  
3 and  
4   self .amount>0
```

## 5.2. Zugriff auf die Kennzahlen

Nachdem wir das Modell eines studentischen Getränkemarktes in der Studentenstadt Freimann (siehe Abbildung 3.4) erweitert haben, können wir aus diesem (siehe Abbildung 5.1) die von den Akteuren gewünschten Kennzahlen (siehe Tabelle 4.1) jeweils durch ein Attribut oder, wenn dafür eine weitere Angabe nötig ist, mittels einer Methode abrufen. Die Zuordnung der Attribute beziehungsweise Methoden der entsprechenden Objekte zu den Kennzahlen lautet wie folgt:

K0 Ist-Arbeitszeit pro Mitarbeiter  
`Employee.workMinutesActual`

K4 Tatsächlicher Stundenlohn pro Mitarbeiter  
`Employee.wageRate`

K5 Umsatz pro Monat pro Mitarbeiter  
`Employee.salesVolume(month)` im Monat `month`

K6 Umsatzbeteiligung pro Monat pro Mitarbeiter  
`Employee.commission(month)` im Monat `month`

K8 Anzahl Barkunden pro Monat  
`Customer.salesPerMonth(month)` bei dem `Customer`-Objekt, welches alle Barkunden darstellt, im Monat `month`

K10 Anzahl Barkunden pro Wochentag  
`Customer.salesPerWeekday(weekday)` bei dem `Customer`-Objekt, welches alle Barkunden darstellt, am Wochentag `weekday`

## 5. Verortung und Implementierung der Kennzahlen im Hybrid Wiki

---

- K11 Anzahl Verkäufe pro Monat pro Kundengruppe  
`CustomerGroup.salesPerMonth(month)` im Monat `month`
- K12 Durchschnittlicher Umsatz pro Kunde pro Kundengruppe  
`CustomerGroup.averageSalesVolume`
- K13 Aktuelle Schulden pro Rechnungskunde  
`Customer.debt`
- K14 Älteste ausstehende Rechnung pro Rechnungskunde  
`Customer.oldestNotPaidSale`
- K16 Umsatz pro Rechnungskunde  
`Customer.salesVolume`
- K17 Liste der ausstehenden Lieferungen  
`Market.ordersToBeDelivered`
- K22 Durchschnittliche Dauer, bis Artikel wieder verfügbar, pro Artikel  
`Product.avgMinutesOutOfStock`
- K26 Liste der zu bezahlenden Lieferungen  
`Market.purchasesToBePaid`
- K28 Aktueller Bestand pro Artikel  
`Product.stockedAmount`
- K36 Drehzahl  
`Product.averageMinutesUntilSoldOut`
- K40 Liste bald verderbender Waren  
`Market.bestBefore(date)` – Artikel, welche maximal bis zum Zeitpunkt `date` haltbar sind.
- K51 Lagerumschlag  
`Product.stockTurnover`
- K54 Aktueller Ist-Kassenstand  
`Account.nominalBalance`
- K55 Aktueller Soll-Kassenstand  
`Account.actualBalance`
- K69 Umsatz pro Monat  
`Market.salesVolume(month)` im Monat `month`

Auf weitere Kennzahlen kann, nach ihrer Implementierung, analog zu diesen zugegriffen werden.

### 5.3. Anpassung des Hybridmodells in Tricia

Bei der Anpassung des Hybridmodells in Tricia wurden die Klassen *DateTime* und *ProductHistory* nicht übertragen, da diese einen entsprechenden Variablentypen und die Wiki-eigene Historisierungsfunktion in unserem UML-Modell darstellen und bereits im Hybrid-Wiki-System integriert sein sollen. Beim Übertragen der anderen Erweiterungen des UML-Modells sind uns folgende Besonderheiten Tricias aufgefallen:

- In Tricia gibt es bisher noch keine Möglichkeit, um Werte aus Attributen zu berechnen. Auch einfache Rechenoperationen innerhalb der Attribute sind noch nicht implementiert. Daher haben wir die OCL-Ausdrücke, mittels einer Markierung durch jeweils ein Dollarzeichen (\$) zu Beginn und am Ende des Ausdruckes, hervorgehoben.
- Pro Assoziation kann nur ein Rollenname vergeben werden. Beim Objekt auf der anderen Assoziationsseite ist die Benennung durch die eingehenden Links (*Incoming Links*) schon vorgegeben und erfolgt nach dem Muster *Rollenname of*. Deshalb haben wir in unserer Implementierung eine redundante Assoziation in die Gegenrichtung erstellt. Dies kann jedoch zu Inkonsistenzen führen.
- Es besteht bisher keine Möglichkeit Methoden mit Eingabeparametern auszuführen, da, abgesehen von der fehlenden Implementierung der Berechnung, keine Parameter übergeben werden können.
- Die Erstellung eines Verkaufsvorganges dauert subjektiv, im Verhältnis zur Durchführung des Verkaufsvorganges an einer Registrierkasse, länger. Ebenso aufwändig ist die Erstellung eines Einkaufsvorganges. Bei den Stammdaten ist der zeitliche Aufwand zu vernachlässigen, da diese nur selten geändert werden. Bei den sonstigen Daten gibt es jedoch sehr viele Änderungen beziehungsweise vor allem neu erstellte Objekte und somit Wiki-Seiten, weswegen hierbei eine schnelle Abwicklung von Vorteil ist.

## 6. Zusammenfassung und Ausblick

Wir konnten mit dieser Arbeit zeigen, dass die für einen studentischen Getränkemarkt nötigen Funktionalitäten eines ERP-Systems im Hybrid-Wiki-System *Tricia* implementiert werden können. Zunächst haben wir dazu ein UML-Modell für einen studentischen Getränkemarkt erstellt. Dieses konnten wir in *Tricia* erfolgreich mittels *Type Tags* umsetzen. An der Befragung von Akteuren mehrerer studentischer Getränkemarkte in der Studentenstadt Freimann nahmen, trotz Verzögerungen durch sich aufgrund der vorlesungsfreien Zeit im Urlaub befindende Akteure, 13 Personen, also ungefähr die Hälfte aller möglichen Akteure, teil. Durch diese Befragung konnten wir identifizieren, welche Kennzahlen für einen studentischen Getränkemarkt in Frage kommen. Anhand der darauffolgenden Online-Umfrage konnten wir diese Kennzahlen danach priorisieren, wie viele Akteure diese einsetzen würden. Die Mehrheit der Befragten äußerten die Absicht mehr als die Hälfte der identifizierten Kennzahlen einzusetzen. Die 21 gefragtesten der 77 identifizierten Kennzahlen haben wir mittels OCL-Ausdrücken in ein UML-Modell, welches wir zuvor erweiterten, eingebettet. Somit konnten wir anhand von bestimmten Attributen und Methoden auf die Kennzahlen zugreifen. Schließlich haben wir das erweiterte UML-Modell, so weit möglich, in *Tricia* umgesetzt. Für eine funktionsfähige Umsetzung in *Tricia*, muss dieses allerdings erst noch um die Möglichkeit der Auswertung von OCL-Ausdrücken erweitert werden. Des Weiteren muss die Implementierung von Assoziationen und Methoden verbessert werden.

Da wir uns bei der Übersetzung der Kennzahlen in OCL-Ausdrücke auf die 21 wichtigsten Kennzahlen beschränkt haben, können weitergehend die restlichen von den Akteuren gewünschten Kennzahlen in OCL übersetzt werden. Hierzu muss allerdings auch das UML-Modell entsprechend erweitert werden.

Nicht unerheblich für die Arbeit mit *Tricia* wäre auch eine Betrachtung der anfallenden Datenmengen durch die hohe Anzahl an Objekten. Da ein Objekt einer Wikiseite entspricht, muss beispielsweise für jeden einzelnen Posten eines Verkaufsvorgangs eine eigene Wiki-Seite erzeugt werden.

Ebenfalls wäre eine Eignung Tricias als POS<sup>16</sup>-Software zu überprüfen. So stellt sich beispielsweise die Frage, ob es für eine schnelle Verkaufsabwicklung nötig ist eine andere Eingabemethode (beispielsweise über einen Barcodescanner) bereitzustellen oder eine ganz andere Implementierung zu wählen.

Wie wir in dieser Arbeit zeigen konnten, ist der Einsatz des Hybrid-Wiki-Systems *Tricia* als Alternative zu einer ERP-basierten Lösung für einen studentischen Getränkemarkt prinzipiell möglich, erfordert aber noch einige Anpassungen Tricias. Diese Arbeit kann als Einstieg für weitere Forschungen dienen, welche sich mit der Fragestellung beschäftigen, ob man ein Hybrid Wiki in bestimmten Unternehmen oder Unternehmensbereichen als ERP-System einsetzen kann.

---

<sup>16</sup>Point of Sale



## Literaturverzeichnis

- [Com] *Compiere Open Source ERP - A Modern, Low-cost ERP Software Solution.* Consona Holdings Inc. <http://www.compiere.com/>. abgerufen am 25. November 2011. 1.1
- [Di07] Diekmann, A.: *Empirische Sozialforschung: Grundlagen, Methoden, Anwendungen.* Rowohlt's Enzyklopädie. Rowohlt Taschenbuch Verlag. 2007. 978-3-499-55678-4. 4.1
- [Gr10] Gronau, N.: *Enterprise Resource Planning: Architektur, Funktionen und Management von ERP-Systemen.* Lehrbücher Wirtschaftsinformatik. Oldenbourg Wissensch.Vlg. 2010. 978-3-486-59050-0. 1
- [Ka02] Kanwischer, D.: *Experteninterviews - die Erhebung, Verwaltung und Dekonstruktion von Expertenwissen.* In *Qualitative Forschungsmethoden in der Geographiedidaktik.* Number 35 in Geographiedidaktische Forschungen. pages 91–112. Selbstverl. d. Hochschulverbandes für Geographie und ihre Didaktik e.V. 2002. 9
- [Ko03] Koppelman, U.: *Beschaffungsmarketing.* Springer-Lehrbuch. Springer. 2003. 978-3-540-40706-5. 4.1.2
- [Lim] *Kostenlose Umfragen erstellen mit LimeSurvey - Einfach und Open Source!* Carsten Schmitz. <http://www.limesurvey.org/>. abgerufen am 25. November 2011. 4.2.1
- [Me11] Meyer, C.: *Betriebswirtschaftliche Kennzahlen und Kennzahlen-Systeme.* Verl. Wiss. und Praxis. 2011. 978-3-896-73599-7. 4.1.2
- [Mi09] Miedl, W.: *Der deutsche ERP-Markt bleibt zersplittert.* *Computerwoche.* 34. August 2009. 1.1

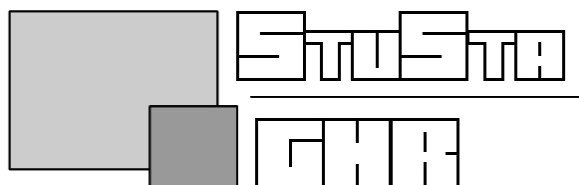
- [Ni08] Nielsen, L.: *Vorgehensmodell zur ERP-Einführung in kleinen und mittelständischen Unternehmen (KMU): Ein Modell aus der Perspektive eines Softwarehauses*. Grin Verlag. Juni 2008. 978-3-638-94778-7. 1.1
- [OCL] *OMG Modeling and Metadata Specifications*. Object Management Group. [http://www.omg.org/technology/documents/modeling\\_spec\\_catalog.htm#OCL](http://www.omg.org/technology/documents/modeling_spec_catalog.htm#OCL). abgerufen am 25. November 2011. 12
- [Oe09] Oestereich, B.: *Analyse und Design mit UML 2.3: Objektorientierte Softwareentwicklung*. Oldenbourg. 2009. 978-3-486-58855-2. 8, 5.1
- [Opea] *Openbravo Home Page - Web-based Open Source ERP*. Openbravo, S.L.U. <http://www.openbravo.com/de/>. abgerufen am 25. November 2011. 1.1
- [Opeb] *OpenZ ERP Warenwirtschaft - Zimmermann Software | OpenZ ERP Warenwirtschaft*. Zimmermann-Software. <http://www.openz.de/OpenZ-ERP-Warenwirtschaft.html>. abgerufen am 25. November 2011. 1.1
- [Pap] *Papyrus UML - Welcome to Papyrus UML web site*. Commissariat à l'énergie atomique. <http://www.papyrusuml.org/>. abgerufen am 25. November 2011. 14
- [St67] Staehle, W.: *Kennzahlen und Kennzahlensysteme: Ein Beitrag zur modernen Organisationstheorie*. Bamberger Fotodruck R. Rodenbusch. 1967. 4
- [Stu] *Studentenstadt Freimann: Studentenstadt*. StuStaNet e.V. <http://www.stusta.mhn.de/stusta.html.de#stusta>. abgerufen am 25. November 2011. 1.1
- [Stu11] *Einkaufen in der StuSta*. StuStaNet e.V. November 2011. [https://wiki.stusta.mhn.de/index.php?title=Einkaufen\\_in\\_der\\_StuSta&oldid=10124](https://wiki.stusta.mhn.de/index.php?title=Einkaufen_in_der_StuSta&oldid=10124). abgerufen am 25. November 2011. 1.2
- [To93] Tomohiko Sakamoto: *Re: Day-of-week algorithm NEEDED!* comp.lang.c. April 1993. <http://groups.google.com/group/comp.lang.c/msg/ce353009f08558b9>. abgerufen am 25. November 2011. 5.1

- [Tri] *infoAsset : Tricia - Hybrid Wikis for Enterprise Collaboration and Information Management*. infoAsset AG. <http://www.infoasset.de/wikis/infoasset/tricia-hybrid-wikis>. abgerufen am 25. November 2011. 1.1
- [UML] *OMG Modeling and Metadata Specifications*. Object Management Group. [http://www.omg.org/technology/documents/modeling\\_spec\\_catalog.htm#UML](http://www.omg.org/technology/documents/modeling_spec_catalog.htm#UML). abgerufen am 25. November 2011. 13
- [WK99] Warmer, J.; Kleppe, A.: *The object constraint language: precise modeling with UML*. Addison-Wesley object technology series. Addison Wesley Longman. 1999. 978-0-201-37940-2. 5.1
- [WKK96] Weygandt, J.; Kieso, D.; Kell, W.: *Accounting principles*. Wiley. 1996. 978-0-471-03661-6. 5.1

## A. Anhang

### A.1. Richtlinien für den Brotladen

# Richtlinien für den Brotladen



Fassung vom 14. Februar 2011

## Präambel

Die Brotladenrichtlinien werden zum Wintersemester 2010/11 einer gravierenden Änderungen unterzogen. Dies geschieht, um die Wirtschaftlichkeit des Brotladens wiederher- und damit den Fortbestand des Brotladens sicherzustellen.

Eine Betreiberstelle, nämlich die des Einkäufers, wird abgeschafft. Stattdessen soll es nur noch einen Betreiber geben, der sämtliche Betreiberaufgaben innehat. Zur Entlastung steht es ihm frei einen Teil dieser Aufgaben an einen oder mehrere Bereichsverantwortliche abzugeben. Die Verantwortung verbleibt aber weiterhin beim Betreiber. Die Bereichsverantwortlichen sind angemessen vom Betreiber zu entlohnen.

Diese Änderung findet am regulären Ende der Amtszeit der bisherigen Einkäuferin statt. Diese Stelle wird auf Beschluss des GHR nicht mehr ausgeschrieben und stattdessen diese neue Version der Brotladenrichtlinien verabschiedet. Der gegenwärtige Brotladenbuchhalter wird damit ohne weitere Wahl zum Brotladenbetreiber.

Diese Regelung gilt nur einmalig bei der Einführung dieser neuen Richtlinien und stellt damit einen Sonderfall dar. Mit dem Ende der Amtszeit des gegenwärtigen Brotladenbuchhalters und künftigen Brotladenbetreibers ist die eine Stelle des Brotladenbetreibers entsprechend der GHR-Geschäftsordnung auszuschreiben. Die Präambel ist danach aus den Richtlinien zu streichen.

## § 1. Grundsätze:

- I. Der Brotladen ist eine Gemeinschaftseinrichtung des GHRs und steht als solche nur den Bewohnern der Studentenstadt und deren Gästen zur Verfügung.  
Der Brotladen ist daher kein öffentlicher Wirtschaftsbetrieb.
- II. Der Brotladen soll in erster Linie nicht gewinnorientiert, sondern kostendeckend betrieben werden.
- III. Die Organe des Brotladens sind:
  - Betreiber des Brotladens
  - Bereichsverantwortliche
  - Wirtschaftsausschuss

Sie sollen in enger Zusammenarbeit ihre Aufgaben durchführen und engen Kontakt mit den Mitarbeitern halten.

## § 2. Betreiber:

Aufgaben und Kompetenzen des Betreibers:

- I. Der Betreiber tritt in Einzelvertretung des GHRs als Vertragspartner der Zulieferfirmen auf.

- II. Er soll den Betrieb des Brotladens organisieren und durchführen.
- III. Er muss die Bestellmenge der Produkte kontrollieren und dem Bedarf entsprechend bestellen bzw. einkaufen.
- IV. Er soll Kasse und Finanzen abrechnen, die täglichen Quittungen der Mitarbeiter, sowie Rechnungen und Belege zur Prüfung aufbewahren und darüber Buch führen (nach den Grundsätzen ordnungsgemäßer Buchführung).
- V. Er soll die Mitarbeiter und die Bereichsverantwortliche für ihren Aufwand entschädigen (die Auszahlung der Aufwandsentschädigung erfolgt monatlich).
- VI. Er kann nach Absprache mit dem Wirtschaftsausschuss Mitarbeiter, welche sich nicht bewährt haben, von der Einteilung ausschließen.
- VII. Er muss erhaltene Rechnungen der Zulieferfirmen baldmöglichst begleichen. Dabei müssen die Rechnungen soweit möglich vom Konto abgebucht werden. Außerdem soll auf den chronologisch sortierten Rechnungen die Kontoauszugsnummer vermerkt werden.
- VIII. Er soll einen monatlichen Wirtschaftsbericht dem Wirtschaftsausschuss vorlegen. Dieser muss folgende Daten enthalten:
  - Monatliche Inventur mit Anfangs- und Endbestand
  - Auflistung der täglichen Einnahmen und der Mitarbeiterquittungen (Kopie des Originals)
  - Auflistung aller Ausgaben:
    - bezahlte Rechnungen bzw. erhaltene Lieferscheine
    - Lohnauszahlung
    - monatliche Bilanz (Gewinn bzw. Verlust des Monats)
    - Aktueller Kassen- und Kontostand
    - Sonstiges
- IX. Er ist dazu verpflichtet, bei ansteigenden Defiziten den Wirtschaftsausschuss sofort davon in Kenntnis zu setzen.
- X. Er soll nötige Änderungen in Einkauf, Organisation, Ausrüstung und Einrichtung nach Absprache mit dem Wirtschaftsausschuss und etwaigen Bereichsverantwortlichen vornehmen.
- XI. Er muss mit etwaigen Bereichsverantwortlichen und mindestens einem Mitglied des Wirtschaftsausschusses eine monatliche Inventur durchführen.
- XII. Er soll jedes Semester eine Werbeaktion durchführen.
- XIII. Er soll Mitarbeiter einteilen, anwerben und anlernen.

### **§ 3. Bereichsverantwortliche:**

Aufgaben und Kompetenzen des/der Bereichsverantwortlichen:

- I. Zu seiner Unterstützung ist der Betreiber befugt – in Einvernehmen mit dem Wirtschaftsausschuss – einen oder mehrere Mitarbeiter als Bereichsverantwortliche zu benennen.
- II. Dieser Bereichsverantwortliche übernimmt gegen eine angemessene Aufwandsentschädigung einen Teil der in § 2 genannten Aufgaben des Betreibers.

- III. Der Bereichsverantwortliche erhält als Mitarbeiter seine Aufwandsentschädigung vom Brotladenbetreiber.
- IV. Der Bereichsverantwortliche hat sich vor seiner Ernennung im GHR vorzustellen.
- V. Die Hauptverantwortung – auch für die von einem Bereichsverantwortlichen übernommenen Aufgaben – liegt weiterhin beim Brotladenbetreiber.
- VI. Bei einem Betreiberwechsel erlöschen die alten Stellen der Bereichsverantwortlichen. Die Inhaber dieser Stellen bleiben jedoch weiterhin Mitarbeiter des Brotladens. Gemäß §3.1 steht es nach seiner Wahl dem neuen Betreiber frei, Bereichsverantwortliche in ihren Stellen zu bestätigen.

#### **§ 4. Wirtschaftsausschuss:**

Der Wirtschaftsausschuss wird vom GHR nach dessen Satzung gewählt.

Aufgaben und Kompetenzen des Wirtschaftsausschusses:

- I. Der Wirtschaftsausschuss ist der direkte Vertreter des GHRs und erstattet dem GHR regelmäßig Bericht.
- II. Er ist das Kontroll- und Überwachungsorgan des Brotladens bzw. des Betreibers, des/der Bereichsverantwortlichen und der Mitarbeiter.
- III. Er ist Ansprechpartner in Personalführungsfragen.
- IV. Er vermittelt bei Verhandlungen mit der Hausverwaltung, dem Studentenwerk und dem GHR.
- V. Er hat ein Vetorecht bei Änderungen der Öffnungszeiten, Sortiment, Preisgestaltung etc. und bei größeren Neuanschaffungen, sowie bei der Berufung und Entlohnung der Mitarbeiter und der Bereichsverantwortlichen.
- VI. Er hat dem GHR-Kassenwart monatlich Bilanz des Brotladens vorzulegen.
- VII. Er kann jederzeit die Kasse überprüfen, die Buchführung kontrollieren und die Geschäftsbücher einsehen.
- VIII. Er muss den GHR informieren, falls keine monatliche Abrechnung des Buchhalters vorliegt.
- IX. Er muss Buch führen über:
  - die monatliche Inventur
  - die monatliche Bilanz des Brotladens
  - die Gewinnausschüttungen an GHR und Betreiber
  - die Brotladenrichtlinien
- X. Der Wirtschaftsausschuss besteht aus zwei Personen, die jeweils 30,- Euro im Monat als Entlohnung vom GHR erhalten.
- XI. Bei nachweislichem Fehlverhalten ist der GHR befugt, den Wirtschaftsausschuss abzuwählen. Werden beide Mitglieder des Wirtschaftsausschusses gleichzeitig ihres Amtes enthoben werden, so kann der GHR mit unmittelbarer Wirkung für die Dauer eines Monats einen Interim-Ausschuss ernennen, während die Ämter neu ausgeschrieben werden.

## **§ 5. Gewinn- bzw. Verlustbeteiligung:**

- I. Der Gewinn wird wie folgt aufgeteilt:
  - 60% erhält der GHR
  - 30% erhält der Betreiber
  - 10% wird einem Bonuskonto zugeführt
- II. Eventuelle Verluste werden von der GHR-Kasse getragen.
- III. Die aus einem Teil des Gewinns zu bildenden Bonusrücklagen werden nach der erfolgten Entlastung des Betreibers durch den GHR an jeweiligen Betreiber ausgezahlt. Der GHR kann in Absprache mit dem Wirtschaftsausschuss die Auszahlung der Bonusrücklagen bei Nichterfüllung der Betreiberaufgaben oder Verstoß gegen diese Richtlinien per Beschluss mit absoluter Mehrheit kürzen oder ganz verweigern. Nicht ausgezahlte Rücklagen fallen an die GHR-Kasse.
- IV. Einrichtungsgegenstände des Brotladens sind GHR-Besitz und werden daher von der GHR-Gewinnbeteiligung angeschafft und instant gehalten (z.B. Reparaturen). Einrichtungsgegenstände werden nach Absprache mit dem Wirtschaftsausschuss angeschafft.

Die Gewinnausschüttung erfolgt nach Prüfung durch den Wirtschaftsausschuss.

## **§ 6. Bezahlung der Betreiber und Mitarbeiter:**

- I. Der Betreiber erhält:
  - 150 Euro als monatliche Aufwandsentschädigung
  - 2 Semester Wohnzeitverlängerung pro Amtsperiode
- II. Die Mitarbeiter erhalten eine angemessene Aufwandsentschädigung, die der Betreiber in Einvernehmen mit dem Wirtschaftsausschuss festlegt.

## **§ 7. Wahl der Brotladenbetreiber:**

- I. Der Betreiber wird jeweils für 2 Semester gewählt.
- II. Die Betreiberstelle muss vom Organisationstutor entsprechend der GHR-Geschäftsordnung ausgeschrieben werden.
- III. Bewerbungsvoraussetzungen:
  - ordnungsgemäß immatrikulierter Studentenstadtbewohner mit mindestens zwei Semester Restwohnzeit.
  - mindestens ein Monat Arbeitserfahrung im Brotladen
  - Organisationserfahrung
  - Grundkenntnisse und Erfahrung mit Buchhaltung
- IV. Der Betreiber wird vom GHR nach dessen Geschäftsordnung gewählt. Bekommt keiner der Bewerber im 1. Wahlgang die absolute Mehrheit, entscheidet im sofort anschließenden 2. Wahlgang eine Stichwahl mit einfacher Mehrheit.
- V. Bei nachweislichem Fehlverhalten ist der GHR befugt, den Brotladenbetreiber abzuwählen.
- VI. Über eine Entlastung des Betreibers entscheidet der GHR mit mit  $\frac{2}{3}$  - Mehrheit der anwesenden Stimmberechtigten.



## **§ 8. Übergabe der Ämter nach Ablauf der Amtszeit:**

- I. Am Ende jeder Amtsperiode muss vom Betreiber eine Inventur und ein Jahresabschluss durchgeführt werden. Dabei muss eine Bilanz erstellt werden.
- II. Der Gewinn muss an GHR und Betreiber ausgeschüttet werden.
- III. Das Amt des Betreibers wird nach der Prüfung durch den Wirtschaftsausschuss an den neuen Betreiber übergeben. Die Übergabe erfolgt bei einem Treffen des alten und des neuen Betreibers und des Wirtschaftsausschusses.  
Diese Übergabe hat vor Amtsantritt des neuen Betreibers zu erfolgen.
- IV. Der neue Betreiber muss sich noch vor Amtsantritt einarbeiten.
- V. Der neue Betreiber erhält ein Startkapital in Höhe von 3000 Euro in Geld und Warenwert. Dieser Betrag muss vom GHR gesichert werden.

## **§ 9. Kassenführung und Entlastung:**

- I. Durch den Verwalter einer "GHR Unterkasse" ist ein Bankkonto anzulegen. Auf diesem werden ausschließlich Gelder der Unterkasse verwaltet.
- II. Der Verwalter rechnet regelmäßig die Unterkasse mit der GHR-Kasse ab. Für die einzelnen Unterkassen gelten folgende Abrechnungszeiträume:
- III. Brotladen: Jeweils zum Quartalsende  
Appartementbörse: Jeweils zum Quartalsende  
Bandprobenraum: Jedes Semester (zum 30.6. und zum 31.12.)  
GHR Anlage: Jedes Semester (zum 30.6. und zum 31.12.)
- IV. Die Unterkassen werden von den GHR-Kassenprüfern mindestens einmal pro Semester geprüft. Die GHR Kassenprüfer erstatten hierüber dem GHR Bericht.
- V. Bei einem Wechsel des Verwalters ist die Unterkasse durch die GHR- Kassenprüfer zu prüfen, die dem GHR hierüber Bericht erstatten. Erst anschließend kann die Kasse an den neuen Verwalter übergeben werden. Prüfung und Übergabe der Kasse haben bis spätestens 4 Wochen nach der Wahl zu erfolgen. Der bisherige
- VI. Verwalter kann nur nach erfolgter Prüfung und Übergabe der Kasse vom GHR entlastet werden.
- VII. Nach Ablauf seiner Amtszeit bzw. der Beendigung seiner Tätigkeit und dem Vorliegen des Berichts der GHR-Kassenprüfer, stimmt der GHR über die Entlastung des Verwalters ab. Über die Entlastung entscheidet der GHR mit mit  $\frac{2}{3}$  - Mehrheit der anwesenden Stimmberechtigten.

## **§ 10 Sonstiges:**

Diese Richtlinien treten am 14.02.2011 per GHR-Beschluss in Kraft.  
Sie sind dem GHR, den Mitarbeitern und dem Wirtschaftsausschuss zugänglich zu machen.  
Der GHR kann diese Richtlinie mit  $\frac{2}{3}$  - Mehrheit der anwesenden Stimmberechtigten ändern.  
Änderungsanträge müssen mit der Tagesordnung bekannt gemacht werden.

Nach einem Semester soll der Erfolg dieser neuen Richtlinien durch den

Wirtschaftsausschuss und den GHR überprüft und die Richtlinien gegebenenfalls abgeändert oder angepasst werden.

## **A.2. Richtlinien des Getränkeverkaufs des Hanns-Seidel-Hauses**

## **Richtlinien des Getränkeverkaufs des Hanns-Seidel-Hauses**

### **§1 Allgemeines**

1. Der Getränkeverkauf ist eine Einrichtung zum Wohle der Bewohner des Hanns-Seidel-Hauses. Der Getränkeverkauf ist kein öffentlicher Wirtschaftsbetrieb.
2. Der Getränkeverkauf befindet sich in einem Kellerraum des Hauses 10, worüber ein Mietvertrag zwischen dem HSH und dem Haus 10 existiert.
3. Der Getränkeverkauf wird vom Heimrat des HSH betrieben. Dazu wählt dieser für den Zeitraum von einem Jahr einen Getränkewart, der als Betreiber den Getränkeverkauf leitet.
4. Bei Amtsantritt wird zwischen dem Betreiber und dem Getränkeverkaufsausschuss des Heimrats des HSH ein entsprechender Vertrag gemäß diesen Richtlinien für ein Jahr festgelegt und von beiden Seiten unterzeichnet. Bei Wiederwahl verlängert sich die Geltungsdauer des Vertrages dementsprechend.
5. Die ordentliche Kündigungsfrist beträgt 2 Monate.
6. Die zu gewährleistenden Mindestöffnungszeiten des Getränkeverkaufs sind Montag bis Samstag, 19:00 Uhr bis 20:00 Uhr, ausgenommen gesetzliche Feiertage. In der vorlesungsfreien Zeit (gemäß TU) steht es dem Getränkewart frei einen Ruhetag pro Woche einzuführen. Längere Öffnungszeiten sind Ermessenssache des Getränkewarts.
7. Der Getränkewart kann weiterhin in Absprache mit dem Getränkeverkaufsausschuss pro Kalenderjahr einen Betriebsurlaub des Getränkeverkaufs von maximal einer Woche festlegen.
8. Die Preise werden durch den Getränkewart festgelegt und unterliegen der Kontrolle des Getränkeverkaufsausschusses. Die Preise entsprechen dem in der Studentenstadt üblichen Niveau und garantieren die Konkurrenzfähigkeit gegenüber anderen Getränkeverkäufen in der Studentenstadt.
9. Bei Hausfesten, die vom Heimrat beziehungsweise HuTT des HSH organisiert werden, liefert der Getränkeverkauf zu Bezugspreisen.

### **§2 Der Getränkewart**

1. Das Amt des Getränkewarts ist ein hausinternes Amt des HSH. Seine Wahl richtet sich nach den Vorgaben der Geschäftsordnung des Heimrats des HSH.
2. Der Getränkewart muss immatrikulierter Student und Bewohner der Studentenstadt, bevorzugt des HSH, sein.
3. Der Getränkewart ist für die Organisation und den reibungslosen Ablauf der Verkaufsstelle verantwortlich. Der Getränkewart arbeitet in der Regel nicht als Thekenkraft, sollte aber bei Ausfall einer Thekenkraft einspringen. Die Aufgaben des Getränkewarts sind unter anderem
  - das Einstellen, Anlernen und die Kontrolle der Thekenkräfte
  - Bestellungen und Entgegennahme von Lieferungen
  - Überprüfung und Bezahlung von Rechnungen und Löhnen
  - Kontoführung, Kostenabrechnung und genaue Buchführung
  - Abschließen von Verträgen in seinem Namen (vgl. §3, Satz 4)
  - Zahlung der Miete gemäß Mietvertrag an Haus 10
  - Erstellung eines Kassenberichts in Zusammenarbeit mit dem Getränkeverkaufsausschuss (mindestens halbjährlich)
  - Verkauf der Waschmünzen zu dem vom Heimrat festgesetzten Preis und Abrechnung mit der Hauskasse. Der Erlös aus den Waschmünzen ist an die Hauskasse abzuführen, wobei der Getränkewart der Hauskasse maximal den Gegenwert von 6.000 Waschmünzen schuldig sein darf. Der Getränkewart erhält hierfür als Aufwandsentschädigung 0,5 v.H. Bonus in Form von Waschmünzen.
4. Der Betreiber erhält monatlich € 205,- als fixen Grundlohn sowie 50 v.H. des Reingewinns. Der Reingewinn errechnet sich aus dem Umsatz abzüglich aller Kosten (inklusive Löhne der Thekenkräfte/des Einkäufers, Grundlohn des Betreibers, Miete, usf.).
5. Der Betreiber haftet nur bei grob verlässigem oder vorsätzlichem Verhalten.

### **§3 Der Getränkeverkaufsausschuss**

1. Der Getränkeverkaufsausschuss des HSH wird vom Heimrat des HSH gewählt und ist für alle Belange des Getränkeverkaufs zuständig.
2. Er besteht aus mindestens vier gewählten Mitgliedern, die Bewohner des HSH sind. Zusätzlich ist der Vertreter der Hauskasse Mitglied des Ausschusses.
3. Der Vertreter der Hauskasse beruft nach Absprache mit den Mitgliedern des Ausschusses die Sitzungen ein und legt den Termin der Inventur fest. Er ist verpflichtet, eine Ausschusssitzung einzuberufen, wenn mindestens 50 v.H. der Mitglieder des Getränkeverkaufsausschusses dies wünschen.
4. Die Aufgaben des Ausschusses sind unter anderem:
  - mindestens halbjährliche Inventur und Überprüfung der Buchführung des Getränkeverkaufs
  - Kontrolle der Preise und Vergütungen
  - Mitsprache bei Verhandlungen mit Lieferanten
  - Information des Heimrats über den Getränkeverkauf
5. Bei Abschluss von Verträgen muss die Zustimmung des Getränkeverkaufsausschusses vorliegen.
6. Bei groben Verstößen gegen diese Richtlinien kann der Getränkeverkaufsausschuss den Getränkewart vorübergehend bis zur nächsten Sitzung des Heimrats suspendieren und/oder eine vorübergehende Schließung des Getränkeverkaufs vornehmen.

### **§4 Personal und Betrieb**

1. Die Thekenkräfte müssen in der Studentenstadt, bevorzugt im HSH, wohnen.
2. Die Thekenkraft erhält einen Lohn in Höhe von € 8,- pro Stunde zuzüglich 1 v.H. des Tagesumsatzes.
3. Die Tageskasse mit ausreichend Wechselgeld ist vor der Öffnung des Verkaufs von der Thekenkraft beim Getränkewart abzuholen und nach Beendigung des Verkaufs wieder bei ihm abzugeben.
4. Die Thekenkräfte sind verpflichtet dem Getränkewart jeden Abend eine Abrechnung vorzulegen.
5. Pro Abend arbeitet eine Thekenkraft.
6. In den Räumlichkeiten des Getränkeverkaufs hat der Getränkewart und die jeweilige Thekenkraft Hausrecht.
7. Bei Bedarf kann der Getränkewart einen Einkäufer einstellen. Dieser erhält pro Fahrt einen Lohn in Höhe von € 15,-.
8. Die Thekenkräfte sowie der Einkäufer unterstehen dem Getränkewart.

### **§5 Finanzen**

1. Das Grundkapital des Getränkeverkaufs des HSH beträgt € 12.500,- als zinsfreies Darlehen aus der Hauskasse. Die Existenz der Summe wird mindestens halbjährlich sowie bei Übergabe durch eine Inventur sichergestellt.
2. Nach jeder Inventur und Abrechnung werden die Überschüsse an die Hauskasse überwiesen. Der Gewinnanteil des Betreibers, der sich aus der Inventur ergibt, wird ebenso erst nach der vollständigen Abrechnung des Gewinns ausbezahlt.
3. Auf Verlangen des Getränkeverkaufsausschusses hat der Betreiber für jeden Monat bis spätestens zum 10. des Folgemonats einen vollständigen Monatsbericht inklusive Kostenabrechnung vorzulegen.
4. Der Betreiber führt ein separates Girokonto für den Betrieb des Getränkeverkaufs.
5. Der Verkauf erfolgt im Allgemeinen nur gegen Barzahlung.

### **§6 Gültigkeit**

1. Diese Richtlinien treten am 23.11.2004 per Heimratsbeschluss in Kraft.
2. Änderungen bedürfen der Zustimmung des Heimrats mit absoluter Mehrheit.

Ver.: 1.1 / 13.11.2004

aufgestellt durch Tuure (0321, Haussprecher)

**A.3. Fragebogen zur Befragung beteiligter  
Akteure nach Kennzahlen für einen  
studentischen Getränkemarkt**

# Befragung beteiligter Akteure nach Kennzahlen für einen studentischen Getränkemarkt

Umfrage zur Bachelorarbeit „Hybrid Wiki als Alternative zu einer ERP-basierten Lösung für einen studentischen Getränkemarkt“ von Raffael Comi

Name, Vorname: \_\_\_\_\_  *Mitarbeiter* Markt: \_\_\_\_\_  
E-Mailadresse: \_\_\_\_\_  *Betreiber*  *Prüfer*

1. Verwenden Sie im Rahmen Ihrer Tätigkeit konkrete Kennzahlen?
Falls ja, welche?
2. Welche (weiteren) Kennzahlen würden Sie erheben/verwenden?
3. Wir haben in früheren Befragungen bereits Kennzahlen identifiziert und diese in einem Kennzahlenkatalog erfasst (siehe Beiblatt). Ist dieser Katalog aus Ihrer Perspektive vollständig, oder würden Sie diesen um weitere Kennzahlen (bitte mit konkreter Angabe) erweitern?

- Ich erkläre mich bereit an einer Folgebefragung via Internet teilzunehmen.  
Dauer: ca. 10 Minuten – Einladung erfolgt an oben genannte E-Mailadresse

# ***Kennzahlenkatalog für einen studentischen Getränkemarkt***

## **Umsatz**

- pro Zeitabschnitt / Zeiteinheit
- pro Mitarbeiter
- pro Kunde

## **Bestellungen**

- pro Zeitabschnitt / Zeiteinheit
- Anzahl an Kästen
- Menge an Pfand
- Warenwert
- ausstehende
- zu bezahlende

## **Kunden**

- pro Zeitabschnitt / Zeiteinheit
- Dauer bis Rechnungen bezahlt
- Aktuelle Schulden

## **Produkte**

- pro Zeitabschnitt / Zeiteinheit verkauft
- Dauer bis Produkt verkauft
- Produkte die oft zusammen eingekauft werden
- aktueller Bestand
- Dauer bis zur nächsten Produktauffüllung

## **Sonstiges**

- Bareinnahmen pro Zeitabschnitt / Zeiteinheit
- Anzahl Münzentnahme/-einlage pro Zeitabschnitt und Münzwert
- aktueller Kassenstand
- Gewinn pro Zeitabschnitt / Zeiteinheit
- Ausgaben/Einnahmen nach Kategorie
  - Personal
  - Fixkosten
  - Waren



## **A.4. Ergebnisse der Befragung**

Markt	Funktion	Frage 1	Frage 2	Frage 3
HSH	Betreiber	<ul style="list-style-type: none"> <li>● Umsatz pro Periode</li> <li>● Umsatz pro Mitarbeiter</li> <li>● Umsatz pro Kunde</li> <li>● Aktuelle Kundenschulden</li> <li>● Verkaufte Produkte pro Periode</li> <li>● Aktueller Kassenstand</li> <li>● Gewinn pro Periode</li> <li>● Einnahmen</li> <li>● Ausgaben</li> </ul>	<p>siehe Kennzahlenkatalog des Fragebogens in Anhang A.3</p>	

Markt	Funktion	Frage 1	Frage 2	Frage 3
C	Betreiber	<ul style="list-style-type: none"> <li>• Tagesumsatz</li> <li>• Gehälter</li> <li>• Verbrauchsmaterial</li> <li>• Waren</li> <li>• Produktverkäufe</li> <li>• Rabatte</li> <li>• Inventar</li> </ul>	<ul style="list-style-type: none"> <li>• Wetterabhängigkeit</li> </ul>	<ul style="list-style-type: none"> <li>• Schwund</li> </ul>

Markt	Funktion	Frage 1	Frage 2	Frage 3
E	Betreiber	<ul style="list-style-type: none"> <li>• Gewinn pro Periode</li> <li>• Handelsspanne EK/VK</li> <li>• MHD-Liste</li> <li>• Umsatz=Kassenbestand-Wechselgeld</li> <li>• Abschöpfung</li> </ul>	<ul style="list-style-type: none"> <li>• Inventurvergleich zwischen Perioden</li> <li>• Artikelabverkaufsmenge</li> <li>• Umsatz pro Artikel</li> </ul>	<ul style="list-style-type: none"> <li>• Kunden Vorbestellungen Bestellwünsche</li> <li>• Produkte "Drehzahl" Preis</li> </ul>
B	Betreiber	<ul style="list-style-type: none"> <li>• Gewinn pro Periode</li> </ul>	<ul style="list-style-type: none"> <li>• Verkaufte Produkte pro Periode</li> </ul>	
B	Mitarbeiter			

Markt	Funktion	Frage 1	Frage 2	Frage 3
F	Mitarbeiter	<ul style="list-style-type: none"> <li>• Stundenlohn</li> <li>• Umsatzbeteiligung</li> <li>• Arbeitszeit</li> </ul>	<ul style="list-style-type: none"> <li>• Tatsächliche Arbeitszeit</li> <li>• Angefragte Produkte</li> </ul>	<ul style="list-style-type: none"> <li>• Haltbarkeitsdaten</li> </ul>
D	Betreiber	<ul style="list-style-type: none"> <li>• Umsatz pro Periode</li> </ul>		
F	Mitarbeiter	<ul style="list-style-type: none"> <li>• Kassenstand</li> <li>• Fehlende Artikel im Regal</li> </ul>	<ul style="list-style-type: none"> <li>• Eigener Umsatz</li> <li>• Gehalt pro Zeitabschnitt</li> <li>• Kundenanzahl pro Arbeitstag</li> </ul>	

Markt	Funktion	Frage 1	Frage 2	Frage 3
A	Betreiber	<ul style="list-style-type: none"> <li>• Umsatz pro Tag</li> <li>• Gewinn pro Periode</li> <li>• Kassenstand</li> <li>• Gehälter pro Periode</li> <li>• Umsatz pro Mitarbeiter</li> <li>• Einnahmen</li> <li>• Ausgaben</li> </ul>	<ul style="list-style-type: none"> <li>• Anzahl Kunden, welche zum Normalpreis einkaufen</li> <li>• Anzahl Kunden, welche zum rabattierten Preis einkaufen</li> <li>• Kassendifferenz</li> </ul>	<ul style="list-style-type: none"> <li>• Anzahl abgelaufener Artikel pro Produkt</li> </ul>
Brotladen	Mitarbeiter	<ul style="list-style-type: none"> <li>• Kassenstand</li> <li>• Differenz bestellte-gelieferte Menge</li> <li>• Selbsteinkauf</li> </ul>	<ul style="list-style-type: none"> <li>• Differenz Umsatz normal und Umsatz bei bestellten Frühstücksangeboten</li> <li>• Kunden pro Zeiteinheit</li> </ul>	

Markt	Funktion	Frage 1	Frage 2	Frage 3
F	Mitarbeiter	<ul style="list-style-type: none"> <li>• Umsatz pro Mitarbeiter</li> <li>• Kassenstand</li> </ul>	<ul style="list-style-type: none"> <li>• Anzahl nachzufüllender Produkte in Auslage</li> </ul>	<ul style="list-style-type: none"> <li>• Bestandswert</li> </ul>
F	Betreiber	<ul style="list-style-type: none"> <li>• Verkaufte Produkte pro Periode</li> </ul>		
HSH	Prüfer	<ul style="list-style-type: none"> <li>• Bestandsbewertung</li> <li>• Gewinn pro Periode</li> <li>• Umsatz pro Periode</li> <li>• Kassenstand</li> <li>• Forderungen</li> <li>• Verbindlichkeiten</li> </ul>		<ul style="list-style-type: none"> <li>• Älteste ausstehende Rechnung pro Kunde</li> <li>• Vergleich Umsatz mit vergangener Periode</li> <li>• Vergleich Gewinn mit vergangener Periode</li> </ul>

## **A.5. Ergebnisse der Online-Umfrage**

$K_n$  repräsentiert die Kennzahl  $n$

✓ bedeutet "würde ich einsetzen"

✗ bedeutet "würde ich nicht einsetzen"

○ bedeutet "keine Angabe"



A. Anhang

Markt	K0	K1	K2	K3	K4	K5	K6	K7	K8	K9	K10	K11	K12	K13	K14	K15
Funktion	K16	K17	K18	K19	K20	K21	K22	K23	K24	K25	K26	K27	K28	K29	K30	K31
	K32	K33	K34	K35	K36	K37	K38	K39	K40	K41	K42	K43	K44	K45	K46	K47
	K48	K49	K50	K51	K52	K53	K54	K55	K56	K57	K58	K59	K60	K61	K62	K63
	K64	K65	K66	K67	K68	K69	K70	K71	K72	K73	K74	K75	K76			
B	✓	✓	○	○	○	○	✓	✓	○	○	○	○	○	✓	✓	○
	○	○	○	✓	○	○	○	○	○	○	○	✓	✓	✓	○	○
	○	✓	✓	○	○	○	✓	✓	✓	○	○	✓	○	○	○	○
	○	○	○	✓	○	○	✓	✓	○	○	○	○	○	○	○	○
	○	○	✓	○	○	✓	○	○	○	✓	○	✓	○			
	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
F	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×
	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×
	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×
	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×
	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
E	✓	✓	✓	✓	✓	○	○	○	○	○	○	○	○	○	○	○
	✓	✓	✓	✓	✓	○	○	○	○	○	○	○	○	○	○	○
	✓	✓	✓	✓	✓	×	✓	✓	✓	✓	✓	✓	✓	○	○	○
	✓	✓	✓	✓	✓	✓	✓	✓	✓	×	✓	✓	✓	✓	✓	✓
	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

A. Anhang

Markt	Funktion	K0	K1	K2	K3	K4	K5	K6	K7	K8	K9	K10	K11	K12	K13	K14	K15	
		K16	K17	K18	K19	K20	K21	K22	K23	K24	K25	K26	K27	K28	K29	K30	K31	
		K32	K33	K34	K35	K36	K37	K38	K39	K40	K41	K42	K43	K44	K45	K46	K47	
		K48	K49	K50	K51	K52	K53	K54	K55	K56	K57	K58	K59	K60	K61	K62	K63	
		K64	K65	K66	K67	K68	K69	K70	K71	K72	K73	K74	K75	K76				
HSH	Prüfer	X	X	✓	✓	✓	X	✓	X	✓	X	X	✓	✓	✓	✓	✓	
		X	✓	X	X	X	X	✓	✓	X	X	X	✓	✓	✓	X	X	X
		X	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	X	X	X	X	X
		X	✓	✓	X	✓	✓	✓	✓	✓	✓	✓	X	○	✓	○	X	X
		X	✓	✓	X	✓	✓	✓	X	✓	X	✓	✓	✓	X	X	X	X
C	Mitarbeiter	✓	X	X	X	X	✓	○	X	X	✓	✓	X	X	✓	X	X	
		✓	✓	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
		✓	X	X	✓	✓	X	✓	✓	○	✓	X	X	X	✓	✓	✓	✓
		X	X	✓	✓	X	○	✓	✓	✓	X	X	X	X	✓	✓	✓	✓
		X	X	✓	X	X	✓	✓	X	X	○	X	X	○	○	○	○	○
A	Betreiber	✓	○	○	✓	✓	✓	✓	✓	✓	○	✓	✓	✓	✓	✓	✓	
		✓	✓	✓	✓	✓	○	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	○	✓	✓
		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	○	○	✓	✓	○	✓	○
		○	✓	✓	✓	✓	✓	✓	○	✓	✓	✓	✓	✓	○	○	✓	○

A. Anhang

Markt	Funktion	K0	K1	K2	K3	K4	K5	K6	K7	K8	K9	K10	K11	K12	K13	K14	K15				
		K16	K17	K18	K19	K20	K21	K22	K23	K24	K25	K26	K27	K28	K29	K30	K31				
		K32	K33	K34	K35	K36	K37	K38	K39	K40	K41	K42	K43	K44	K45	K46	K47				
		K48	K49	K50	K51	K52	K53	K54	K55	K56	K57	K58	K59	K60	K61	K62	K63				
		K64	K65	K66	K67	K68	K69	K70	K71	K72	K73	K74	K75	K76							
F	Mitarbeiter	X	X	✓	✓	✓	✓	✓	X	X	X	X	✓	✓	✓	X	✓				
		✓	✓	X	X	X	X	✓	✓	X	X	X	✓	✓	✓	✓	✓	✓			
		✓	✓	✓	X	✓	✓	X	✓	○	✓	✓	✓	✓	X	X	X	X	X		
		X	X	X	X	X	✓	✓	✓	✓	○	X	X	X	X	X	X	X	X	X	
		✓	✓	✓	✓	✓	✓	✓	✓	✓	○	X	✓	X	X	X	X	X	X	X	
F	Mitarbeiter	✓	✓	✓	✓	✓	✓	✓	✓	X	X	✓	X	X	✓	X	X				
		✓	✓	✓	X	X	X	✓	✓	✓	X	X	X	○	✓	✓	✓	✓	✓	✓	
		✓	X	X	X	✓	✓	✓	✓	X	✓	✓	✓	X	✓	✓	X	X	X	X	X
		✓	X	X	X	X	X	X	✓	✓	✓	✓	✓	X	X	X	X	X	X	X	X
		✓	X	X	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
F	Mitarbeiter	X	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓				
		✓	✓	X	X	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓			
		✓	✓	✓	X	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	
		✓	X	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
		X	✓	✓	✓	✓	✓	✓	X	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

A. Anhang

Markt	K0	K1	K2	K3	K4	K5	K6	K7	K8	K9	K10	K11	K12	K13	K14	K15
	K16	K17	K18	K19	K20	K21	K22	K23	K24	K25	K26	K27	K28	K29	K30	K31
<b>Funktion</b>	K32	K33	K34	K35	K36	K37	K38	K39	K40	K41	K42	K43	K44	K45	K46	K47
	K48	K49	K50	K51	K52	K53	K54	K55	K56	K57	K58	K59	K60	K61	K62	K63
	K64	K65	K66	K67	K68	K69	K70	K71	K72	K73	K74	K75	K76			
C	✓	✓	✓	✗	✓	✓	✗	✗	✓	✓	✓	✓	✓	✓	✓	✓
	✓	✗	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
F	✗	✗	✓	✓	✓	✓	✓	✓	✓	✗	✓	✗	✗	✓	✓	✓
	✓	✓	✓	✓	✓	○	✗	○	✗	✓	✓	✗	✓	✓	✓	✗
	✓	✓	✓	✗	✓	✓	✓	✓	✓	✓	✗	✓	✓	✗	✓	✗
	✓	✓	✗	✓	✓	✓	✓	✓	✗	✗	✗	✓	✓	✗	✓	○
	✓	✓	✓	✓	✗	✓	✓	✓	✗	✗	✓	✗	✓	✓	✓	✓