



# PFLICHTENHEFT PROJEKT LFH

## FFHS 2013

*Version:* 1.1

*Autor:* Jonas Alder, Patrick Bösch, Sandro Dallo, Andy Villiger

*Studiengang:* Bsc Inf 2011

*Ort:* Zürich

*Datum:* 002. Oktober 2013

## Dokumentenverlauf

VERSION	WER	BEMERKUNGEN	DATUM
<b>0.1</b>	DSO	Entwurf	15.09.2013
<b>0.2</b>	BOE	Anpassungen an der Struktur und am Inhalt	29.09.2013
<b>1.0</b>	BOE	Zeitplan eingefügt	30.09.2013
<b>1.1</b>	VIL	Zeitplan-Update, Controlling eingefügt, Fehler behoben	02.10.2013

# Inhaltsverzeichnis

## 1 Inhalt

2	Allgemeines .....	1
2.1	Sinn und Zweck .....	1
2.2	Lesekreis .....	1
2.3	Produktumfang .....	1
3	Gesamtübersicht .....	2
3.1	Ausgangslage.....	2
3.2	Ziel .....	2
3.3	Produktumfeld.....	2
3.3.1	System-Schnittstellen.....	2
3.3.2	Abgrenzungen .....	3
3.3.3	Software .....	3
3.3.4	Produkt-Funktionalität.....	4
4	Spezifische Anforderungen.....	5
4.1	Funktionale Anforderungen .....	5
4.1.1	Kundenregistratur.....	5
4.1.2	Online Katalog .....	5
4.1.3	Warenkorb .....	5
4.1.4	Zahlungsabwicklung.....	5
4.1.5	After-Sales.....	5
4.1.6	Content Management .....	5
4.1.7	Kundenverwaltung.....	5
4.2	Generelle Grundfunktionalität.....	5
4.2.1	System-Meldungen.....	5
4.3	Übersicht der spezifischen Anforderungen .....	6
4.3.1	Übersicht Modul Kunde.....	6
4.3.2	Übersicht Modul After Sales .....	6
4.3.3	Übersicht Modul Warenkorb.....	6
4.3.4	Übersicht Modul Zahlungsabwicklung .....	6
4.3.5	Übersicht Modul Online Katalog.....	6
4.3.6	Übersicht Modul Produktverwaltung.....	6
4.3.7	Übersicht Modul Kundenverwaltung .....	6

## Inhaltsverzeichnis

5	Leistungsanforderungen.....	7
5.1	Datenhaltung.....	7
5.2	Dauer .....	7
5.3	Sicherheit.....	7
5.4	Antwortzeit.....	7
6	Testen der Applikation .....	8
6.1	Unit-Tests.....	8
7	Zeitplan.....	9

## 2 Allgemeines

### 2.1 Sinn und Zweck

Das vorliegende Dokument dient als Grundlage und Raster für die Neuentwicklung des E-Shops für das MAVE Projekt der LFH. Zudem dient es für weitere Detailanalysen und dem Design der Applikation. Das Pflichtenheft beinhaltet alle Anforderungen, die an den E-Shop für das Projekt MAVE der LFH gestellt werden.

Der E-Shop ermöglicht es der LFH sowohl im Rahmen des Projekt MAVE Produktverkäufe in einem Online-Shop zu simulieren sowie auch einen echten Online Shop im Internet zu betreiben.

### 2.2 Lesekreis

Dieses Dokument richtet sich an die Projektleitung des Projektes MAVE der LFH, sowie an die betreuenden Dozenten.

Das Pflichtenheft ist soweit möglich in einer Sprache abgefasst, welche wenig technische Vorkenntnisse voraussetzt.

### 2.3 Produktumfang

Um in der Simulationslösung MAVE der LFH Warenverkäufe zu simulieren wird die webbasierte Anwendung E-Shop entwickelt. Die Applikation stellt eine Kundenregistrierung, einen Produktkatalog, einen Warenkorb und Zahlungsmöglichkeiten bereit, sowie die Verwaltungsmöglichkeit dieser Daten.

Die funktionalen Anforderungen sind im Kapitel 4.1 Funktionale Anforderungen auf Seite 5 abschliessend beschrieben.

## 3 Gesamtübersicht

### 3.1 Ausgangslage

Im Rahmen des MAVE Projektes der LFH haben wir uns entschlossen das Teilprojekt des E-Shop ESHOMO (Gülke, 2013) zu übernehmen. Die LFH als Auftraggeber erteilte den Auftrag zur Erstellung des E-Shops und lieferte ein Lastenheft. Die Vertretung der LFH im Projekt als Auftraggeber wird durch Frau Heimberg und Herrn Claes wahrgenommen. Das Projektteam der FFHS (mit wechselnden Rollen) setzt sich aus den Herren Alder, Bösch, Dallo und Villiger zusammen. Begleitet wird das Projekt von Herr Winiger und Herr Kamin.

MAVE ist ein Simulationswerkzeug und ist Modular konzipiert. Der E-Shop wird als Modul von MAVE erstellt und über Schnittstellen an die MAVE Infrastruktur angeschlossen. Das Modul des E-Shops trägt den Namen ESHOMO.

### 3.2 Ziel

Der Webshop soll als Modul innerhalb des Projektes MAVE dafür zuständig sein Verkäufe über eine Onlineplattform zu simulieren. Dazu wird ein E-Shop implementiert, welcher über Schnittstellen mit dem Überprojekt kommunizieren kann. Der E-Shop soll aber vom Projekt MAVE unabhängig lauffähig sein.

### 3.3 Produktumfeld

#### 3.3.1 System-Schnittstellen

Es werden HTTP-Schnittstellen für das Exportieren und Importieren von Kunden, Artikel und Bestellungen bereitgestellt. Die Daten werden dabei je nach Parameter im JSON Format (type=json) oder XML (type=xml [Standard]) geliefert. Mit dem id Parameter wird definiert, ob genau ein Object (id=Integer) oder alle (id=all) zurückgegeben werden. Mit dem optionalen Parameter since kann ein Unix-Timestamp übergeben werden, welcher die Datenlieferung auf die Datensätze einschränkt, welche nach dem gelieferten Datum eingefügt wurden.

Zur Übersicht nochmals alle Schnittstellen:

<b>Kunden lesen:</b>	Kunden aus dem E-Shop nach MAVE exportieren
<b>Kunden schreiben:</b>	Kunden aus MAVE in den E-Shop importieren
<b>Bestellungen lesen:</b>	Bestellungen aus dem E-Shop nach MAVE exportieren
<b>Bestellungen schreiben:</b>	Bestellungen aus MAVE in den E-Shop importieren
<b>Artikel lesen:</b>	Artikel aus dem E-Shop nach MAVE exportieren
<b>Artikel schreiben:</b>	Artikel aus MAVE in den E-Shop importieren

**Version lesen:** Version des E-Shop Moduls auslesen

Die Schnittstellen werden im Dokument Schnittstellenbeschreibung\_MAVE-3.pdf definiert.

### 3.3.2 Abgrenzungen

Die Schulung der Benutzer, sowie die Installation der Software und die Integration der Software in die Produktivumgebung ist nicht Teil dieses Auftrages und wir deshalb nicht in diesem Pflichtenheft behandelt.

In diesem Pflichtenheft werden alle Anforderungen definiert, welche umgesetzt werden. Aus Zeitgründen – die Projektzeit ist durch die FFHS begrenzt – können nicht alle Anforderungen umgesetzt werden.

### 3.3.3 Software

Der E-Shop wird mit der Programmiersprache Java (Vorgabe der FFHS) erstellt werden. Für die Client Seite werden HTML Dokumente generiert, welche mit der Technik des *Responsive Designs* erstellt werden, damit das System auf möglichst vielen Geräten benutzbar ist. Auf der Server Seite wird eine relationale Datenbank angesprochen, um die Daten zu persistieren.

### 3.3.4 Produkt-Funktionalität



Wie in der Abbildung zu erkennen ist, werden die Funktionalitäten des E-Shops in zwei Akteure aufgeteilt. Einerseits den Standardbenutzer, welcher nur auf eine begrenzte Anzahl der Funktionalitäten Zugriff hat und anderseits den Administrator des Systems, welcher Vollzugriff auf sämtliche Funktionen hat.

## 3.4 Controlling

Für dieses Projekt wird das Arbeitszeit- und Kostencontrolling durch den Controller der LFH sichergestellt. Ein wöchentlicher Arbeitsrapport wird von der Projektgruppe der FFHS an die Projektleiter LFH gesendet.

## 4 Spezifische Anforderungen

### 4.1 Funktionale Anforderungen

#### 4.1.1 Kundenregisteratur

Kunden können sich mit Hilfe eines Formulars registrieren und Ihre Daten eingeben. Sämtliche Informationen und getätigte Transaktionen können im Kundenprofil abgerufen werden. Der Zugang zum Kundenkonto erfolgt mit Benutzernamen und Passwort.

#### 4.1.2 Online Katalog

Eine Übersicht aller verfügbaren Produkte für den Kunden in einem Produktkatalog darstellen. Per Volltext-Suche soll der Kunde bei der Produktsuche unterstützt werden.

Verwaltung des Produktekataloges durch den Administrator oder Produktemangers.

#### 4.1.3 Warenkorb

Ein Kunde kann Produkte im Shop auswählen und dem Warenkorb hinzufügen. Der Kunde kann auch Anonym Produkte auswählen und dem Warenkorb hinzufügen, sobald ein Kunde zahlen möchte, wird er aufgefordert sich anzumelden oder zu registrieren. Der Kunde kann Produkte im Warenkorb jederzeit ändern und entfernen, die Anzahl eines Produktes bestimmen und neue Produkte hinzufügen.

#### 4.1.4 Zahlungsabwicklung

Der Kunde kann zwischen drei Zahlungsarten auswählen, Vorauskasse, Rechnung und per Kreditkarte.

#### 4.1.5 After-Sales

Der Kunde kann den Status der Lieferung jederzeit einsehen. Der Kunde kann nach dem Kauf die Produkte bewerten, diese können dann von anderen Kunden auch gelesen werden.

#### 4.1.6 Content Management

Die Produktedaten können manuell auf der Shop-Seite aktualisiert und bearbeitet werden. Die Daten können auch per XML und einem SOAP Web-Service importiert und aktualisiert werden.

#### 4.1.7 Kundenverwaltung

Die Kundendaten werden in der Datenbank abgelegt.

## 4.2 Generelle Grundfunktionalität

### 4.2.1 System-Meldungen

#### 4.2.1.1 Zweck

Der Benutzer soll mithilfe von System-Meldungen über den Status des Systems informiert werden.

## 4.3 Übersicht der spezifischen Anforderungen

### 4.3.1 Übersicht Modul Kunde

Beschreibung	Priorität
Registrierung des Kunden	1
Daten verwalten	1
Transaktionsübersicht	2
Login	1

### 4.3.2 Übersicht Modul After Sales

Beschreibung	Priorität
Produkt bewerten	2

### 4.3.3 Übersicht Modul Warenkorb

Beschreibung	Priorität
Hinzufügen von Artikeln zum Warenkorb	1
Anzahl der bestellten Artikel editieren	2
Artikel aus dem Warenkorb entfernen	1

### 4.3.4 Übersicht Modul Zahlungsabwicklung

Beschreibung	Priorität
Vorauskasse	1
Rechnung	1

### 4.3.5 Übersicht Modul Online Katalog

Beschreibung	Priorität
Suchen in der Produktliste	2
Auflisten der Produkte	1
Detaildarstellung der Produkte	1
Kategorien	2

### 4.3.6 Übersicht Modul Produktverwaltung

Beschreibung	Priorität
Produkt ins System eingeben	1
Produkt editieren	1
Produkt einer Kategorie hinzufügen	2
Kategorien hinzufügen	2
Kategorien editieren	2

### 4.3.7 Übersicht Modul Kundenverwaltung

Beschreibung	Priorität
Kunden editieren	1
Kunden sperren	2
Kunden löschen	1

## 5 Leistungsanforderungen

### 5.1 Datenhaltung

Sämtliche Daten des E-Shops werden in einer relationalen Datenbank persistiert. Zurzeit ist nicht definiert, welches genaue Produkt eingesetzt wird, ist aber auch noch nicht nötig, da ein ORM Framework eingesetzt wird. Da das System eine Neuentwicklung ist, muss ein Datenbank-Diagramm entworfen und umgesetzt werden.

### 5.2 Dauer

Sämtliche Daten werden in der Datenbank nicht physikalisch sondern logisch gelöscht. Somit ist es jederzeit möglich, die gelöschten Daten einzusehen und bei Bedarf wiederherzustellen. Für die Wiederherstellung der gelöschten Dateien werden gute Informatikkenntnisse vorausgesetzt.

### 5.3 Sicherheit

Die Sicherung der Datenbank muss in die Backup-Strategie des MAVE Projekts aufgenommen werden.

### 5.4 Antwortzeit

Der Server muss bei jeder Anfrage innerhalb von maximal 500ms antworten.

## 6 Testen der Applikation

### 6.1 Unit-Tests

Das Ziel der Unit-Tests ist es automatisiert die Programm Funktionalität der einzelnen Module zu testen.

Die Unit-Tests werden während der Realisierungsphase von der Gruppe LFH der FFHS geschrieben. Es wieder die einzelnen Module getestet.

## 7 Zeitplan

