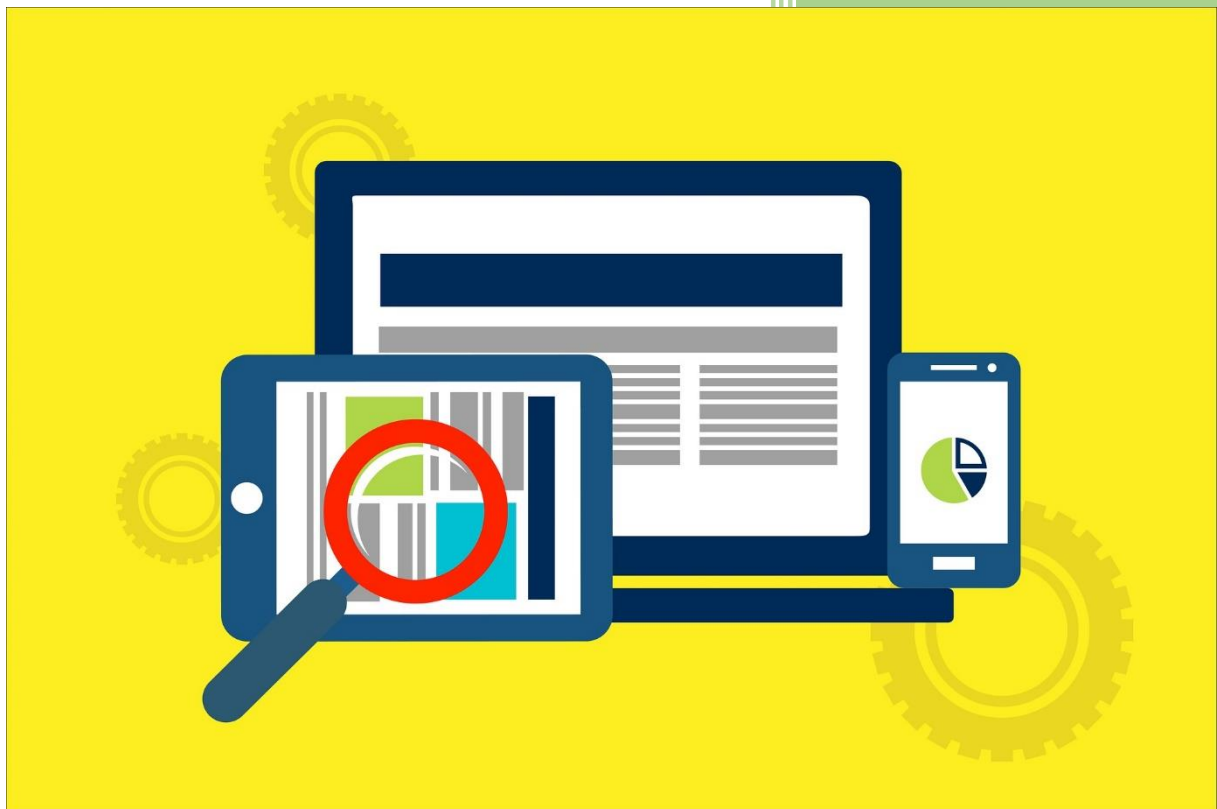


Analyse E-Shop Wako



Patrick Tomasi

Jakov Ratkovic

Lauren Arekhi

Inhaltsverzeichnis

1	Einleitung	2
1.1	Ressourcen	2
1.2	Initialisierung	2
2	IST-Zustand.....	3
2.1	Analyse der funktionalen Aspekte	3
2.1.1	Starseite	3
2.1.2	Angebote	3
2.1.3	Login	3
2.1.4	Registrieren	3
2.1.5	Suchen	3
2.2	Analyse nicht funktionale Aspekte	4
2.3	Layout.....	4
2.4	UML Zustandsdiagramm.....	5
2.5	UML Anwendungsdiagramm	5
2.6	HW Architektur	6
2.7	Datenbank.....	7
3	SOLL-Zustand	8
3.1	UML-Zustandsdiagramm	8
3.2	UML-Anwendungsdiagramm	9
3.3	Verbesserungsvorschläge	10
4	Testfälle.....	11
4.1	Login	11
4.1.1	Beschreibung:.....	11
4.1.2	Tests:	11
4.1.3	Erwartungen:	11
4.2	Registrierung.....	12
4.2.1	Beschreibung:.....	12
4.2.2	Tests:	12
4.2.3	Erwartungen:	12
4.3	Warenkorb	12
4.3.1	Beschreibung.....	12
4.3.2	Tests	12
4.3.3	Erwartungen.....	12

1 Einleitung

Dieses Dokument gilt der Analyse des E-Shops Wako.

Im ersten Schritt wird eine IST-Zustand Analyse gemacht. Dabei erstellen wir eine Übersicht über alle verwendeten Ressourcen der Webseite. Wir testen die aktuelle Seite und analysieren alle verwendeten Techniken.

Danach wird ein SOLL Zustand des Produktes erarbeitet. Ziel ist es Verbesserungsvorschläge zu präsentieren und Testfälle zu definieren.

Der Kunde hat uns eine ZIP Datei abgegeben. Und den Auftrag erteilt die Webseite in Betrieb zu nehmen und Analysieren.

1.1 Ressourcen

Verwendete Ressourcen, um die Webseite und die Datenbank in Betrieb nehmen zu können:

Webserver:

- Apache Webserver 2.4.41

Programmiersprachen:

- PHP-Version: 7.3.9
- HTML 5
- CSS 3

Datenbanktyp:

- MariaDB Version 10.4.6

Versionierung:

- GitHub

1.2 Initialisierung

Die Datenbank wurde mit phpMyAdmin und dem vom Kunden erhaltenem SQL-Script initialisiert.

Der Datenbank Name musste erstellt werden und danach konnte das Script ohne Probleme initialisiert werden.

Um die Webseite zu starten und um mit der Datenbank zu kommunizieren wurde XAMPP in der Version 3.2.4 verwendet.

Der Code für die Webseite ist in PHP. Dazu mussten alle Dateien in den htdocs Ordner von XAMPP kopiert werden. Damit die Webseite aufrufbar ist und mit der Datenbank kommunizieren kann muss in XAMPP Apache und MySQL laufen.

2 IST-Zustand

Die in der IST-Zustand ermittelten Daten werden einer detaillierten Analyse unterzogen. Sie dient der Ermittlung des aktuellen Zustands und dem Erkennen von Mängeln und Schwachstellen in der Applikation. Dies dient als Basis für alle weiteren Schritte des Projekts und ist essential für eine Verbesserung oder Erweiterung des Produkts.

2.1 Analyse der funktionalen Aspekte

Hier werden die einzelnen funktionalen Aspekte analysiert und dokumentiert.

2.1.1 Startseite

Mit einem Klick auf den Button Startseite gelangt der User zur Startseite

2.1.2 Angebote

Über den Button Angebote werden dem User Links zu den Kategorien angezeigt, die er anklicken kann, aber es wird nichts angezeigt. Die Daten werden nicht aus der Datenbank gelesen oder dem User angezeigt.

2.1.3 Login

Es ist ein Login Formular enthalten mit dem man den Benutzernamen wie das Passwort eingeben kann und Absenden. Der Login funktioniert nicht und man kann sich nicht als User anmelden.

2.1.4 Registrieren

In der Navigation ist auch ein Link Registrieren vorhanden. Mit Klick auf den Link funktioniert noch nichts. Man kann sich nicht als neuer User Registrieren.

2.1.5 Suchen

Im Header der Webseite unter dem Titelbild ist ein Suchfeld eingefügt mit der der User auf der Webseite etwas suchen kann. Die Suchfunktion bringt momentan keine Resultate.

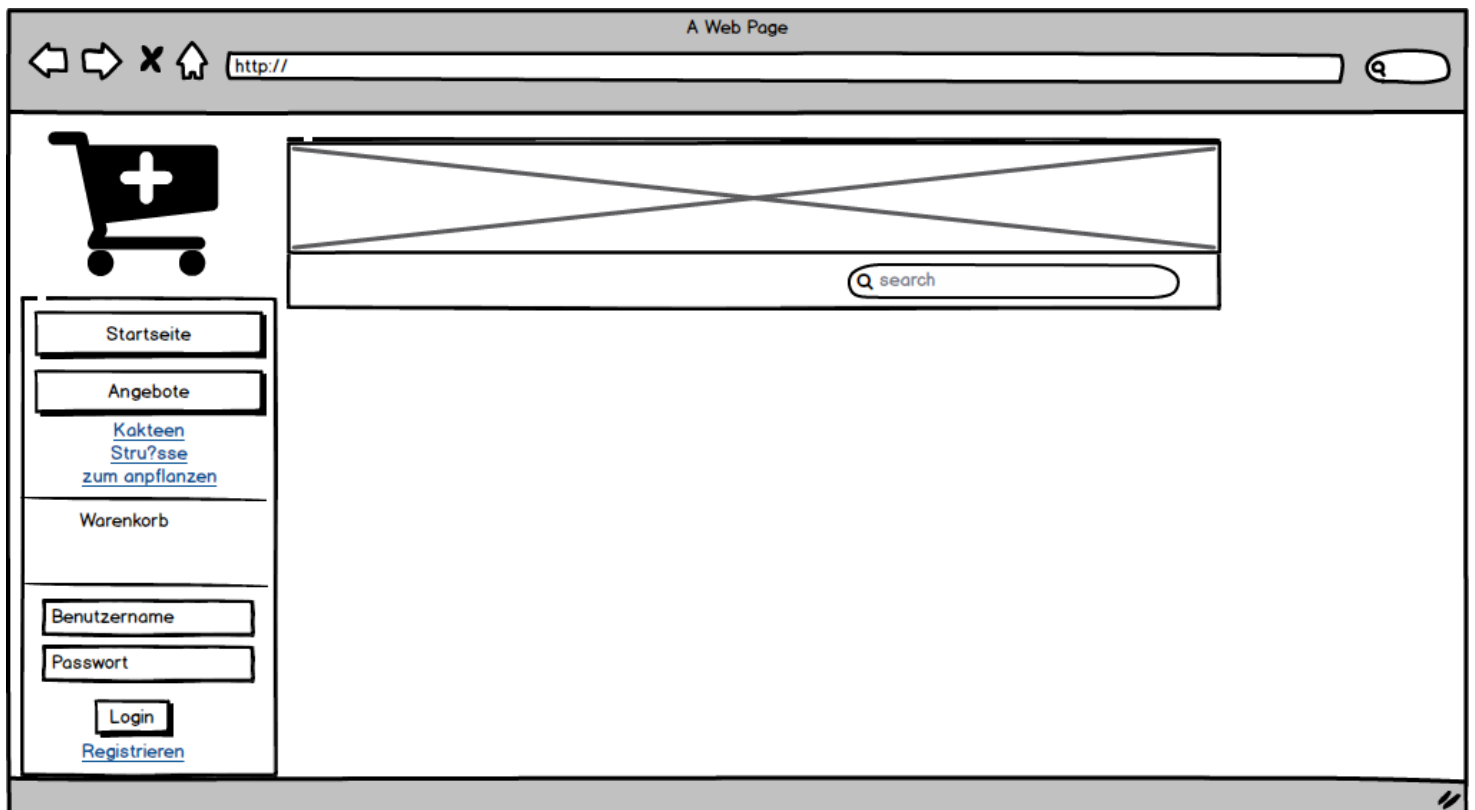
2.2 Analyse nicht funktionale Aspekte

In der Datenbank ist eine Tabelle Kategorien vorhanden aus der die Kategorien ausgelesen werden und dann auf der Webseite angezeigt werden.

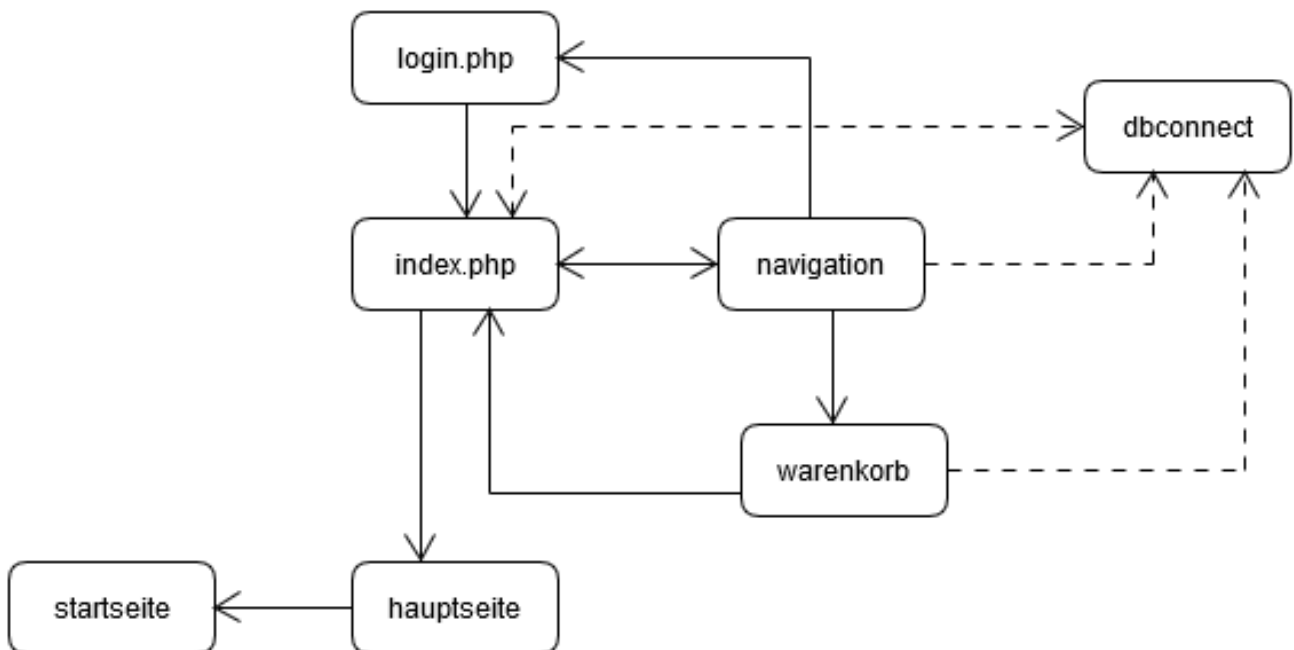
Es sind noch weitere Tabellen vorhanden für den Warenkorb, Bestellungen, Kunden, Produkte und Admins die keinen Einfluss in die Webseite haben.

2.3 Layout

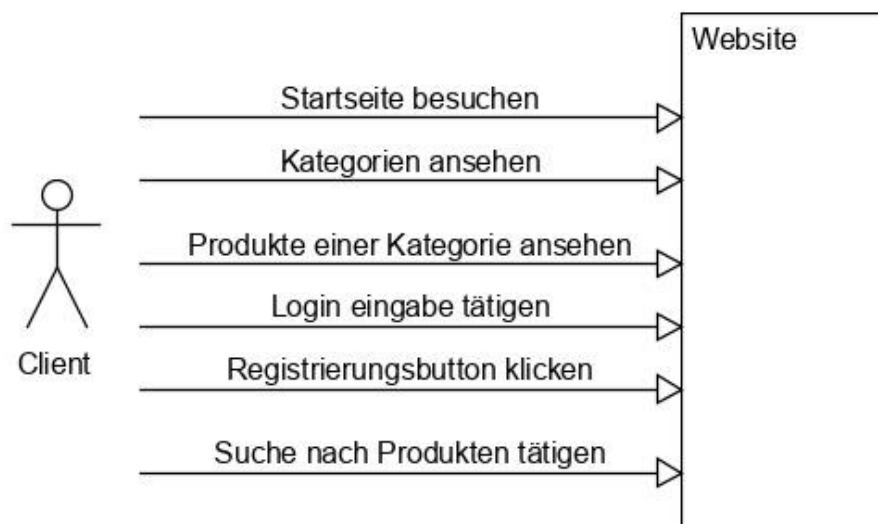
Mit einem Mockup wird kurz aufgezeichnet wie der momentane Stand der Webseite ist und was schon vorhanden ist. Da die ganzen Untermenüs wie Login, Registrieren und anzeigen von Angeboten momentan keine Resultate bringt ist das Mockup nur aus einer Seite.



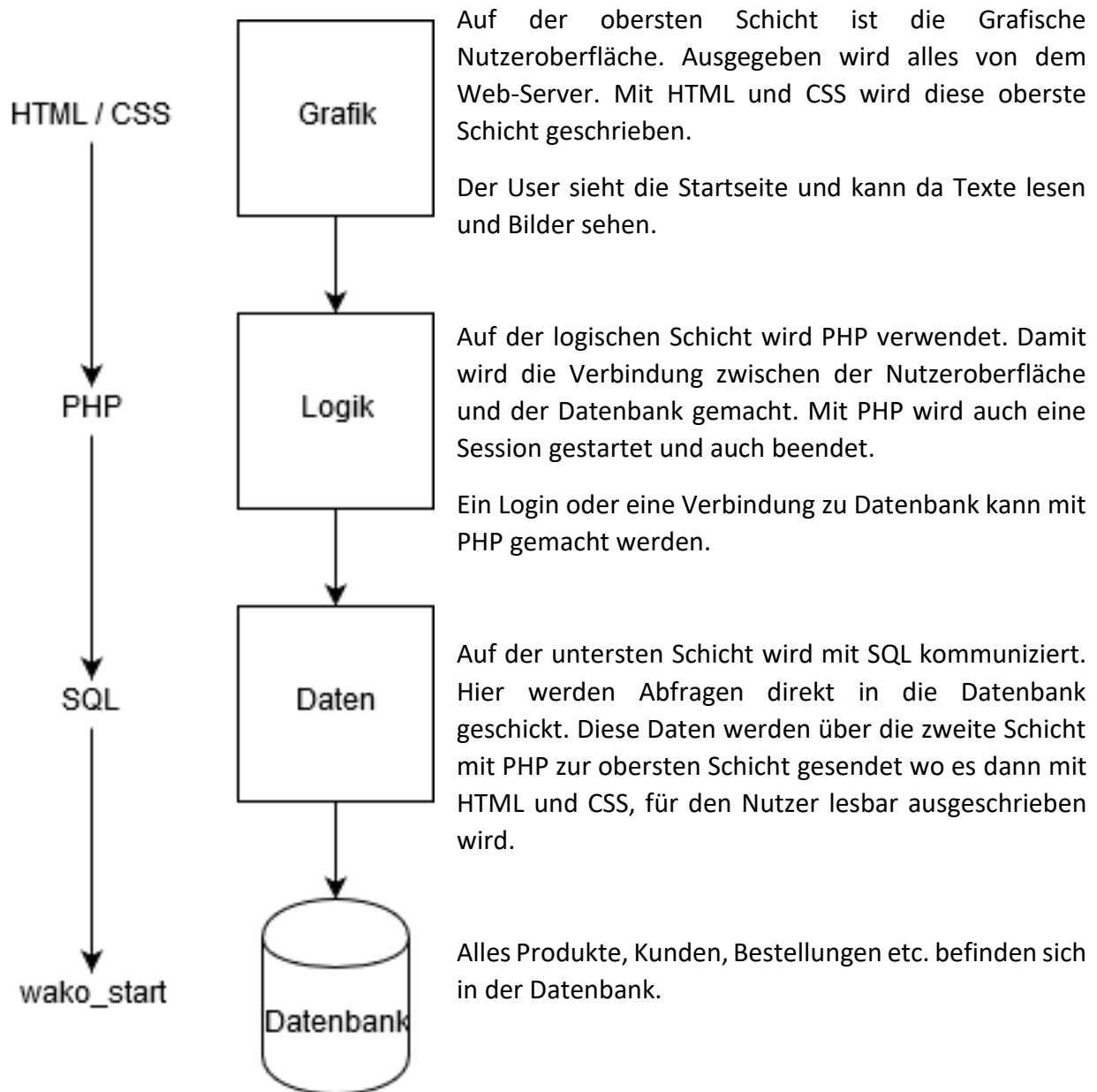
2.4 UML Zustandsdiagramm



2.5 UML Anwendungsdiagramm



2.6 HW Architektur



2.7 Datenbank

Der ist Zustand der Datenbank ist wie folgt:

Es sind 7 Tabellen vorhanden die schon teilweise mit Daten befüllt sind.

Die Datenbanken haben momentan noch keine Relation zu einander.

Die Datenbank sieht wie folgt aus.

wako_neu admins	
Admins_ID : int(11)	
Admins_Benutzername : varchar(40)	
Admins_Passwort : varchar(32)	

wako_neu kunden	
Kunden_ID : int(11)	
Kunden_Name : varchar(40)	
Kunden_Vorname : varchar(40)	
Kunden_Adresse : varchar(40)	
Kunden_PLZ : varchar(10)	
Kunden_Ort : varchar(40)	
Kunden_Land : varchar(40)	
Kunden_Benutzername : varchar(40)	
Kunden_Passwort : varchar(32)	
Kunden_Gesperrt : int(11)	

wako_neu besucher	
Besucher_ID : int(11)	
Besucher_Zeit : datetime	

wako_neu kategorien	
Kategorien_ID : int(11)	
Kategorien_Bezeichnung : varchar(40)	

wako_neu produkte	
Produkte_ID : int(11)	
Produkte_Name : varchar(40)	
Produkte_Bestellnr : varchar(40)	
Produkte_Beschreibung : text	
Produkte_Preis : double(10,2)	
Produkte_Geloescht : varchar(40)	
Produkte_Bildpfad : varchar(200)	
Kategorien_ID : int(11)	

wako_neu warenkoerbe	
Warenkoerbe_ID : int(11)	
Besucher_ID : int(11)	

wako_neu bestellungen	
Bestellungen_ID : int(11)	
Bestellungen_Anzahl : varchar(40)	
Bestellungen_Preis_Total : double(10,2)	
Produkte_ID : int(11)	
Auftraege_ID : int(11)	
Warenkoerbe_ID : int(11)	

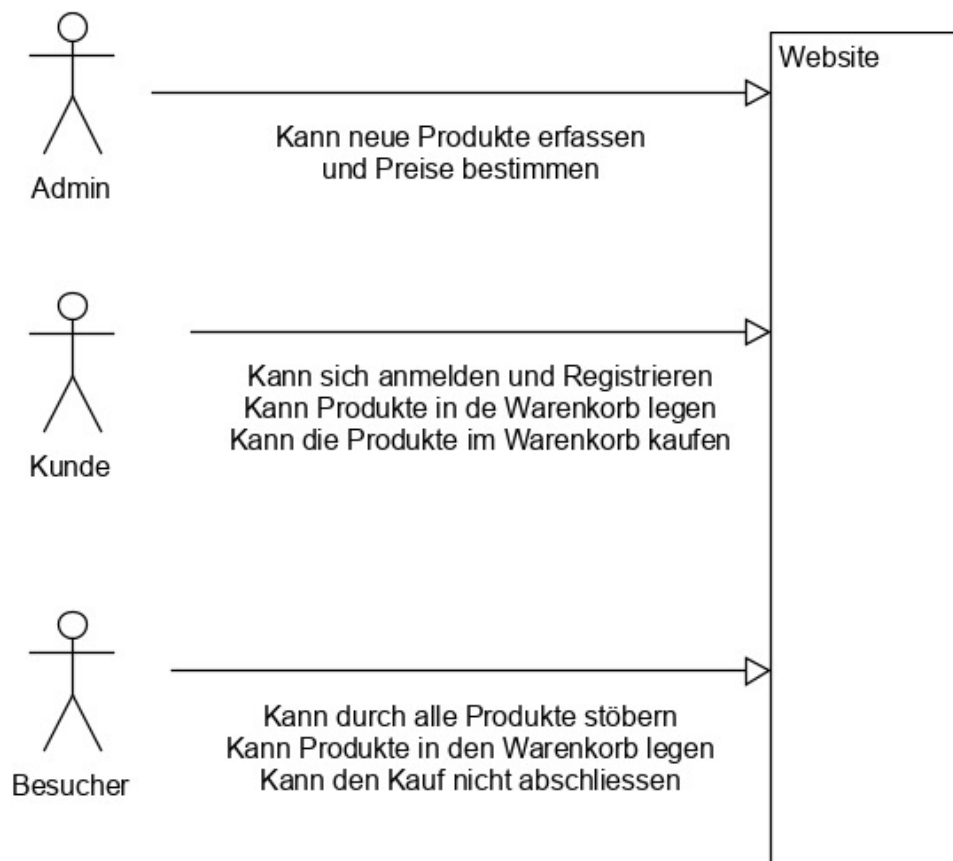
In der Applikation sind schon einige SQL-Abfragen eingebaut, diese sind mit der Technologie MySQLI implementiert worden.

Die Abfragen werden nicht prepared sprich eine SQL-Injection wäre hier problemlos möglich.

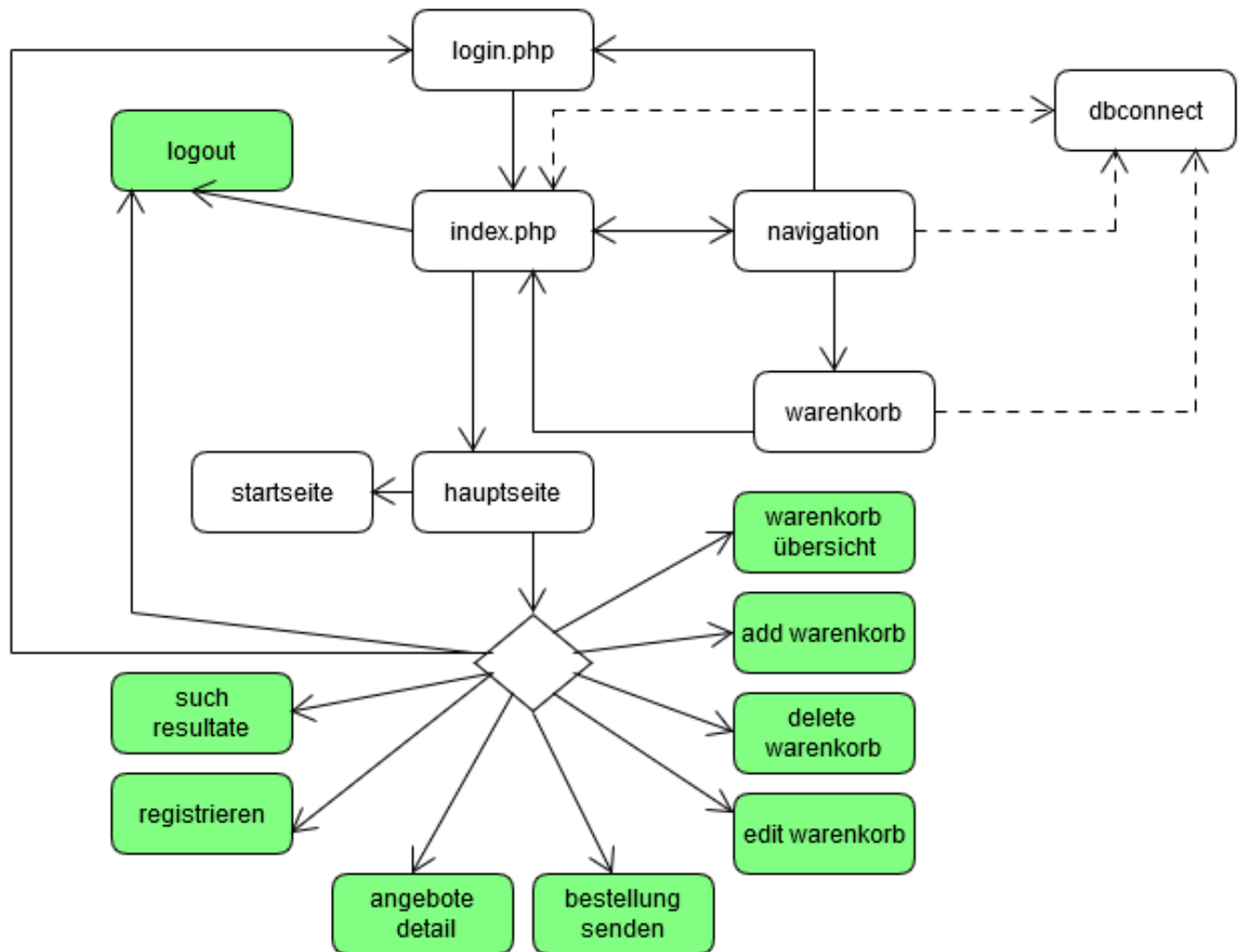
3 SOLL-Zustand

Der SOLL Zustand soll aufzeigen wie die Applikation auszusehen hat. Um den SOLL Zustand grafisch auch darzustellen wurde wie in der IST Analyse eines Zustands und Anwendungsdiagramm erstellt.

3.1 UML-Zustandsdiagramm



3.2 UML-Anwendungsdiagramm



3.3 Verbesserungsvorschläge

In den Verbesserungsvorschlägen möchten wir dem Kunden aufzeigen was verbessert werden kann oder auch muss um die Applikation Ordnungsgemäss zu Implementieren.

Login:

- Da das Login momentan nicht geht sollte die Funktion hinter dem Login implementiert werden.
- Die Datenbank muss mit den Entsprechenden Felder ergänz und angepasst werden.

Registrierungsformular:

- Da sich ein neuer Kunde auf der Webseite nicht registrieren kann müsste ein Registrierungsformular implementiert werden.
- Die Logik zur Registrierung.

Produkte Anzeigen:

- Um den Kunden eine Auswahl der vorhandenen Produkte anzeigen zu können sollte das Anzeigen der Produkte implementiert werden.

Suche:

- Der Suchbutton müsste angepasst werden damit der Kunde auch nach gewünschten Produkten auf der Webseite suchen kann.

Warenkorb:

- Um eine Bestellung auch auf der Webseite zu tätigen müsste einen Warenkorb implementiert werden.
- Die ganze Logik hinter dem Warenkorb.
- Datenbank Anpassungen.

Datenbank:

- Die Datenbank müsste überarbeitet werden. Einzelne Tabellen entfernt oder Spalten hinzugefügt werden.
- Die Datenbank braucht eine Relation.

4 Testfälle

Wir definieren hier 3 Tests, die wir beim Entwickeln oder Erweitern der Applikation durchführen werden.

Tests während der Entwicklung dienen dazu, die Applikation möglichst fehlerfrei in Betrieb zu nehmen. Mit Testfällen können wir sicher gehen das was wir programmiert haben auch wirklich funktioniert.

Mit unseren Testfällen können wir Zustände definieren, die wir erreichen wollen. Dies ist zum Beispiel ein Login, das funktioniert. Der User sollte sich ohne Probleme Einloggen können. Neue User sollten sich registrieren können.

Nebst dem Login wollen wir auch ein funktionierender Warenkorb implementieren.

Damit der Warenkorb ohne Probleme benutzt werden kann wird er auch von uns getestet.

Erst wenn alle Testfälle bestanden haben, haben wir unser Ziel erreicht.

4.1 Login

4.1.1 Beschreibung:

Um sicher zu stellen das die implementierte Login Logik richtig funktioniert werden wir das Login Verfahren testen.

4.1.2 Tests:

- Password wird verschlüsselt
- Die Abfrage mit der Datenbank funktioniert
- Validierung

4.1.3 Erwartungen:

Es wird erwartet das das Login zu 100% funktioniert. Das das eingetragene Password mit SHA256 verschlüsselt wird bevor es zur Datenbank gesendet wird zur Überprüfung. Die Eingaben sollen validiert werden. Der Registrierte User kann sich mit dem richtigen Benutzernamen und Passwort einloggen und eindeutig identifiziert werden.

4.2 Registrierung

4.2.1 Beschreibung:

Da ein komplett neues Registrierungsformular erstellt werden muss und auch die Logik dahinter implementiert werden muss möchten wir die Registrierung in die Testfälle integrieren.

4.2.2 Tests:

- Korrektes Formular mit allen nötigen Eingab Felder
- Validierung der Eingabefelder
- Passwort Verschlüsselung
- Korrektes abspeichern der Daten in der Datenbank

4.2.3 Erwartungen:

Es wird erwartet das das Registrierungsformular zu 100% funktioniert. Das die Eingabefelder alle nötigen Informationen für eine korrekte Registrierung enthalten. Das die einzelnen Felder Validiert werden und das das Passwort verschlüsselt wird. Es wird auch erwartet das die eingegebenen Daten korrekt in der Datenbank abgespeichert werden.

4.3 Warenkorb

4.3.1 Beschreibung

Der Warenkorb wird komplett neu implementiert. Um sicherzustellen das der Warenkorb richtig funktioniert werden wird den Warenkorb und die Implementierte Logik dahinter testen.

4.3.2 Tests

- Artikel lassen sich in den Warenkorb legen
- Artikel lassen sich bearbeiten oder Entfernen im Warenkorb
- Der Warenkorb enthält alle nötigen Informationen, und speichert den Warenkorb in der Datenbank ab

4.3.3 Erwartungen

Es wird erwartet das der Warenkorb zu 100% funktioniert. Das die gewünschten Artikel in den Warenkorb gelegt werden. Der Warenkorb muss bearbeitet werden können oder auch ein Artikel wieder entfernt werden. Der Warenkorb muss korrekt in die Datenbank gespeichert werden.