

„EinkaufsApp“ Pflichtenheft

Informationen zum Pflichtenheft

| | |
|------|--|
| An | Frau Prof. Dr. Wieland Gustav-Freytag-Str. 43-45, 04277 Leipzig |
| Über | Projekt EinkaufsApp |
| Von | EinkaufsApp |

Vorgelegt von

| | |
|-------------------|----------------------------|
| Projektleiter | Markus Hube |
| Gruppenmitglieder | Huong Dang |
| | Thomas Elias |
| | Viktor Fuchs |
| | Florian Graupeter |
| | Jannis Grohs |
| | Michael Hein |
| | Moritz Karsten |
| | Sebastian Kiepsch |
| | Annika Köstler |
| | Daniel Sawadenko |
| | Moritz Schaub |
| | Florian Schmitt |
| | Eric Sorgalla |
| E-Mail | markus.hube@hft-leipzig.de |

Inhaltsverzeichnis

| | |
|---------------------------------------|---|
| Informationen zum Pflichtenheft | 2 |
| Inhaltsverzeichnis | 3 |
| Abbildungsverzeichnis | 4 |
| Abkürzungsverzeichnis | 5 |
| 1. Funktionalitäten der App | Fehler! Textmarke nicht definiert. |
| 2. Anwendungsbereiche | Fehler! Textmarke nicht definiert. |
| Systemarchitektur | Fehler! Textmarke nicht definiert. |
| 1.1 Backend..... | Fehler! Textmarke nicht definiert. |
| 1.2 App..... | Fehler! Textmarke nicht definiert. |
| Use Cases | Fehler! Textmarke nicht definiert. |
| Quellen | Fehler! Textmarke nicht definiert. |

Abbildungsverzeichnis

| | |
|---------------------------------------|----|
| Abbildung 1: Systemarchitektur..... | 11 |
| Abbildung 2: Aufbau des Backend | 12 |

Abkürzungsverzeichnis

| | |
|-------|--|
| Abb. | Abbildung |
| PHP | Hypertext Preprocessor |
| HTML | Hypertext Markup Language |
| HfTL | Hochschule für Telekommunikation Leipzig |
| DWI13 | Duales Studium Wirtschaftsinformatik Jahrgang 2013 |
| JS | JavaScript |
| CSS | Cascading Style Sheets |
| App | Anwendungsprogramm |
| EAN | European Article Number |
| OS | Operating System |
| iOS | iPhone Operating System |
| HTTP | Hypertext Transfer Protocol |
| HTTPS | Hypertext Transfer Protocol Secure |
| Jade | A Rendering Engine |

1. Zielbestimmung

1.1. Musskriterien

Die App muss folgende Funktionalitäten erfüllen:

- Der Nutzer muss sich gegenüber eines Servers authentifizieren; ihm werden daraufhin seine Daten zugeordnet
- Identifizieren von Artikeln anhand ihrer EAN Nummer
- Interpretieren der EAN Nummer aus einen Strichcode mittels einer Scan-Funktion
- Zusammenfassen mehrerer Artikel zu einem Einkauf
- Zuordnen weiterer Informationen wie Preis, Menge und Namen zu einem Artikel während des Einkaufes
- Persistentes Speichern dieser Einkäufe
- Nutzer- und Gruppenverwaltung
- Zuordnen von Einkäufen zu anderen Nutzern oder einer Gruppe
- Durchführung verschiedener Auswertungen über gespeicherte Einkäufe

1.2. Wunschkriterien

Folgende Eigenschaften sind wünschenswert:

- Es können für die Einkäufe abzuarbeitende Listen vorbereitet werden
- Anhand des vergangenen Kaufverhaltens werden personalisierte Vorschläge gemacht
- Es findet während des Einkaufes ein Preisvergleich mit in der Nähe befindlichen Märkten statt
- Anderen Benutzern können Einkäufe zugewiesen werden
- Ein Nutzer wird darüber informiert, wenn jemand anderes ihm oder einer Gruppe in der er sich befindet einen Artikel zuweist
- Eine Funktionalität, um den Bestand der Artikel innerhalb eines Haushaltes nachzuvollziehen
- Hinzufügen „sonstiger Kosten“ zu den Einkäufen (sonstige Kosten sind alle Kosten, die nicht unmittelbar mit einer EAN Nummer in Verbindung gebracht werden können)

1.3. Abgrenzungskriterien

Folgende Aspekte sollen nicht umgesetzt werden:

- Für Jeden Aspekt der Wunschkriterien gilt, dass sie nicht in dem Bereich der zwingend umzusetzenden Anforderungen stehen
- Die App erhebt keinen Anspruch auf wirtschaftliche Abrechnungsgenauigkeit
- Es werden keine zusätzlichen Sicherheiten für Verbindungsausfälle vorgesehen – eine permanente Internetverbindung wird vorausgesetzt

2. Produkteinsatz

1.4. Anwendungsbereich

Die App soll bei jedem Einkauf genutzt werden können, um diesen zu dokumentieren. Ebenfalls soll sie im Nachhinein zu jeder Zeit eine Übersicht über die vergangenen Einkäufe und ihre Auswertungen bieten. Außerdem soll es möglich sein gegenüber anderen Nutzern eine transparente und nachvollziehbare Zusammenfassung gekaufter Artikel anzuzeigen.

1.5. Zielgruppe

Jede Privatperson, unabhängig vom Alter, kann diese App nutzen um Ihren Einkaufsvorgang zu unterstützen.

1.6. Betriebsbedingungen

Einzige Voraussetzung ist ein Smartphone mit entsprechend unterstützter Plattform um die App zu betreiben.

3. Systemarchitektur

Zur Lösung der gegebenen Problemstellung wird sowohl ein Datenbank- und Webserver entwickelt, der die notwendige Datenlogik verwaltet und aus dem Internet erreichbar ist, sowie eine Smartphone App, die dem Benutzer als bedienfreundliche Schnittstelle zur komfortablen Eingabe zusätzlicher Daten während des Einkaufes und zur übersichtlichen Auswertung dient.

Auf dem Server kommen zur Datenspeicherung eine MongoDB aus Skalierbarkeits- und Flexibilitätsgründen zum Einsatz. Weiterhin wird ein Webserver mittels Node.JS aufgrund der hohen Leistungsfähigkeit bei rechenarmen Webaufgaben realisiert. Hinzukommt das Tool Mongoose um als Schnittstelle zwischen Datenbank und Express, welches hier die Routes für den Nutzerzugriff bereitstellt, zu dienen.

Zur Entwicklung der App werden Ionic und AngularJS genutzt. Diese Frameworks ermöglichen das Arbeiten nach dem MVC Prinzip (Modell-View-Controller) und arbeitet sehr gut mit der erläuterten Backend Lösung zusammen. Hier werden entsprechend Views, Routes, Controller und Factories entwickelt.

4. Funktionen

Die Hauptfunktionalitäten werden durch eine API Schnittstelle zum Server bereitgestellt und umfassen:

- **User**
 - Eine Userliste ausgeben
 - Einen einzelnen User ausgeben
- **Gruppen**
 - Eine Liste aller Gruppen in der ein User ist
 - Eine Liste aller User in einer Gruppe
 - Erstellen einer Gruppe
 - Ändern einer bestehenden Gruppe
 - Löschen einer Gruppe
- **Artikel**
 - Einen Artikel ausgeben
 - Erstellen eines neuen Artikels
 - Ändern eines Artikels
 - Löschen eines Artikels
- **Artikel-Kosten-Elemente**
 - Ein Element aus dem Zusammenschluss Artikel und Kosten erstellen
 - Ein Element aus dem Zusammenschluss Artikel und Kosten finden
- **Einkäufe**
 - Eine Liste der Einkäufe die ein User erstellt hat anzeigen
 - Einen bestimmten Einkauf anzeigen
 - Liste aller Einkäufe an die ein User beteiligt war
 - Einen Einkauf erstellen
 - Einen bestehenden Einkauf ändern
 - Ein Element erstellen, dass einen kompletten Einkauf repräsentiert
 - Löschen eines Einkaufes
- **Einkaufsläden**
 - Eine Liste der Einkaufsläden
 - Einen spezifischen Einkaufsladen
 - Erstellen eines Einkaufsladens
 - Ändern eines Einkaufsladens
 - Löschen eines Einkaufsladens

Die App stellt noch einen Scanner bereit und bietet ein Interface zur komfortablen Eingabe aller Daten. Die entsprechenden, relevanten Daten können dem Datenbankschema entnommen werden.