

# Easy-Cocktail-Mixing

Tino Kederer  
Industrie-4.0-Informatik  
OTH Amberg-Weiden  
Amberg, Deutschland  
t.kederer@oth-aw.de

Michael Eichenseer  
Industrie-4.0-Informatik  
OTH Amberg-Weiden  
Amberg, Deutschland  
m.eichenseer1@oth-aw.de

Matthias Schießl  
Industrie-4.0-Informatik  
OTH Amberg-Weiden  
Amberg, Deutschland  
m.schiessl1@oth-aw.de

Jakob Götz  
Industrie-4.0-Informatik  
OTH Amberg-Weiden  
Amberg, Deutschland  
j.goetz2@oth-aw.de

Stefan Ries  
Industrie-4.0-Informatik  
OTH Amberg-Weiden  
Amberg, Deutschland  
s.ries@oth-aw.de

**Keywords—** *web, javascript, mongodb, cocktails, animation*

## I. EINLEITUNG

Im Rahmen der Vorlesung „Web-Anwendungsentwicklung“ haben sich verschiedene Teams zusammengefunden, um Projektarbeiten zu realisieren. Unser „Team Blau“ besteht aus 5 Studierenden des Studiengangs „Industrie-4.0-Informatik“, die diese Vorlesung als Wahlfach besuchen. Da keiner aus dem Team tiefergehende Erfahrung in der Web-Entwicklung hatte und diese Inhalte auch sonst im Studium nicht näher beleuchtet wurden, war es allen Teilnehmern sehr wichtig dies auf diesem Weg nachzuholen.

In dem hier geschilderten Projekt soll es um die Entwicklung einer einfach zu bedienenden und übersichtlichen Web-Anwendung gehen, die dem Nutzer hilft, gute Cocktails zu mischen. Das Ganze soll als Schritt-für-Schritt abzuarbeitende Anleitung realisiert werden, bei der der Nutzer erst einen Arbeitsschritt ausgeführt haben muss und dies mittels Interaktion bestätigt, bevor er den nächsten Schritt aufrufen kann.

Die ganze Anwendung soll durch eine Datenbank gestützt und unterstützt werden, so dass eine Nutzerverwaltung und verschiedene spezifische Nutzerfunktionalitäten möglich sind. Ebenfalls soll die Anwendung durch eine ansprechende Oberfläche und kleinere Animationen ein gutes Nutzererlebnis bieten.

## II. VERWANDTE ARBEITEN

### A. Thermomix

Als große Inspiration für die Arbeit, diente der Thermomix von Vorwerk, der es dem Nutzer ermöglicht, auch ohne große Kochfähigkeiten, komplexere Gerichte zu kochen. Da es in der Projektarbeit um die Entwicklung einer Web-Anwendung gehen sollte, war klar, dass man sich hier nur auf die Schritt-für-Schritt Zubereitung und das einfache Rezept-Handling als zu realisierende Funktionen beschränkt und keine Hardware-Realisierungen in irgendeiner Form anstrebt.

## III. ANFORDERUNGEN

Für das Projekt wurden verschiedene Anforderungen definiert, indem zuerst Funktionalitäten definiert wurden und

darauf aufbauend User-Stories entwickelt wurden, die dann nach dem MoSCoW-Prinzip eingeteilt wurden.

### A. Must

Ich als Benutzer möchte eine Website haben, die mir in einer Cocktailübersicht anzeigt, sodass ich sehe welche Cocktails es gibt

- Website muss aufgerufen sein
- Datenbankbindung muss aktiv sein
- Cocktailliste muss eingeblendet sein

Ich als Benutzer möchte ein detailliertes Rezept für einen Cocktail haben, damit ich diesen ebenso zubereiten kann.

- Der Benutzer muss einen Cocktail aus der Übersicht gewählt haben
- Eine neue Seite mit dem entsprechenden Rezept geht auf

### B. Should

Ich als Benutzer möchte eine Suchfunktion haben, in der ich meinen gewünschten Cocktailnamen eingebe, damit ich nicht alle Cocktails durchsuchen muss

- Eine Suchzeile muss auf der Weboberfläche eingeblendet sein
- Der Benutzer muss eine Eingabe im Suchfeld gemacht haben
- Nach Bestätigung muss dem Benutzer eine gefilterte Liste angezeigt werden

Ich als Benutzer möchte eine Step-by-Step Anleitung haben, damit ich eine detaillierte Zubereitungsanleitung bekomme.

- Der Benutzer muss einen Cocktail ausgewählt haben
- Der Benutzer muss die Zubereitung gestartet haben

### C. Could

Ich als Benutzer möchte mich in auf der Seite registrieren können, um auf verschiedene Features der Seite zuzugreifen

- Eine Loginfunktion muss vorhanden sein
- Um sich einzuloggen, muss man zuvor registriert sein
- Zur Registrierung muss der Benutzer einen Benutzernamen und eine E-Mail-Adresse hinterlegen

Ich als Benutzer möchte meine Lieblingscocktails in einer Favoritenliste speichern können, um auf diesen jederzeit schnell zugreifen zu können.

- Eine Favoritenliste muss eingeblendet sein
- Eine Funktion zum Hinzufügen in die Favoritenliste muss vorhanden sein
- Der Benutzer muss eingeloggt sein, um seine eigene Favoritenliste individuell anzuzeigen

Ich als Benutzer möchte meine zuletzt aufgerufenen Rezepte in einer Liste gespeichert haben, um auf diese im Nachhinein erneut zugreifen zu können

- Anhand der zuletzt aufgerufenen Rezepte soll eine Liste der letzten Rezepte erstellt werden
- Der Benutzer muss eingeloggt sein, um seine zuletzt benutzten Rezepte individuell anzuzeigen

Ich als Benutzer möchte meine eigens kreierte Cocktails in die Cocktailliste eintragen können, sodass diese auch von anderen Benutzern ausprobiert werden können

- Eine Funktion zum "Einschicken" von eigenen Rezepten muss vorhanden sein
- Der Benutzer muss eingeloggt sein

Ich als Benutzer möchte (evtl. fehlerhafte) Cocktails verbessern können und eventuell Zutaten hinzufügen können, sodass fehlerhafte Rezepte richtiggestellt werden

- Ein Reportsystem muss vorhanden sein, in dem der Benutzer Cocktails bemängeln bzw. verbessern kann
- Ein zu bemängelnder Cocktail muss ausgewählt sein

Ich als Benutzer möchte mich in auf der Seite einloggen können, damit ich Cocktails hinzufügen kann

- Der Benutzer muss seine E-Mail bestätigt haben

Ich als Benutzer möchte sehen, wie viele Kalorien das Rezept hat, damit ich diese in meinen Kalorientagebuch aufnehmen kann

- Zu jeder Zutat muss eine Kalorien Angaben hinterlegt sein
- Auf der Rezepthauptseite muss die Gesamtkalorienanzahl anhand der Zutaten angezeigt werden.

## IV. Methoden

### A. MongoDB

Für die anzubindende Datenbank, die die Nutzerverwaltung übernehmen soll und in welcher die Rezepte hinterlegt werden sollen, fiel die gemeinsame Entscheidung auf

MongoDB, weil ein Gruppenmitglied hier bereits Erfahrung vorzuweisen hatte und es außerdem Vorteile bezüglich einfacheren Handlings bietet.

### B. Node.js, Express

Für die Realisierung des Webservers, der dann die Anwendung zur Verfügung stellt, wurde sich auf Node.js in Kombination mit express.js geeinigt.

Node.js ist eine JavaScript-Laufzeitumgebung welches es erlaubt, JavaScript außerhalb eines Browsers auszuführen, sodass auch im Backend-Bereich mit JavaScript programmiert werden kann.

Express ergänzt als serverseitiges Webframework die Node.js-Laufzeitumgebung. Dieses erleichtert uns, durch das Bereitstellen von vorgefertigter Funktion, das Einrichten des Webservers.

### C. React.js

Für die Entwicklung im Frontend hat sich React.js als geeignet erwiesen. Da dies als einsteigerfreundlicher gilt und aufgrund der wenigen Erfahrung in der Gruppe bezüglich Web-Entwicklung war es klar, dass die Lernkurve an dieser Stelle nicht zu steil sein sollte. Außerdem gestattet es, dass es unterschiedliche Bibliotheken und Frameworks noch zusätzlich angebunden werden können.

### D. Hosting

Aufgrund von guten Erfahrungen in der Vergangenheit wurde sich für das Hosting auf eine Lösung über Amazon-AWS geeinigt. Dies bietet Vorteile durch seine Einfachheit und außerdem ist an diesem Punkt bereits Erfahrung vorhanden, da dies im Rahmen der Übungen zur Vorlesung bereits behandelt wurde. Ebenfalls steht uns hierfür ein ausreichendes Guthaben zur Verfügung, so dass hier keine Kosten für uns entstehen.

## V. REFERENZEN

- [1] Thermomix by "Vorwerk, 1883", <https://www.vorwerk.com>
- [2] MongoDB 11. Feb 2009, <https://www.mongodb.com>
- [3] Node.js 27. Mai 2009, <https://nodejs.org>
- [4] Express.js 16. November 2010, <https://expressjs.com>
- [5] React.js, 2013, <https://reactjs.org/>
- [6] Amazon Web Services 2006, <https://aws.amazon.com>