

Proof of Concept

1. Falsche Informationsauswertung

Rezeptdatenbank wird abgefragt, um Rezepte für eine bestimmte Anzahl von Tagen ausgewählt werden. Die Rezepte werden nach den Angaben des Benutzers auf dem Server gefiltert und dem Client bereitgestellt.

Exit:

Die Rezepte werden passend zum Benutzer und seinen Angaben ausgewählt und akzeptiert.

Fail:

Die Rezepte entsprechen nicht den Vorstellung des Nutzers.

Fallback:

Der Benutzer hat die Möglichkeit jedes Rezept individuelle auszutauschen oder eine komplett neue Woche neu generieren zu lassen. Somit kann der User es sich aussuchen ob er ein weiteres Rezept generiert haben möchte oder direkt mit einer Suchfunktion ein bestimmtes Rezept bekommt.

2. Anpassung der Woche

Die Woche/Tage wurden vom Nutzer geplant und festgelegt. Aufgrund dessen wird die Einkaufsliste zusammengestellt.

Exit:

Der Benutzer kocht jeden Tag das gewünschte Gericht, wie es im Wochenplan geplant wurde.

Fail:

Der Nutzer hat an einem Tag nicht gekocht und dem entsprechend noch Lebensmittel über.

Fallback:

Der User kann sich entscheiden, ob die restliche Woche um einen Tag verschieben will, um seine Woche

3. Lebensmittelverwertung

Der Lebensmittelbestand ist die Grundlage der Resteverwertung. Der Nutzer hat die Möglichkeit Angaben zu seinen Resten zu machen und das System ermittelt dann mögliche Rezepte aus den Lebensmitteln

Exit:

Der Nutzer gibt die Lebensmittelreste an und aufgrund dieser Angaben können ausgewählte Rezepte ermittelt werden. Alle Reste können verwertet werden.

Fail:

Die Reste reichen nicht aus für ein Rezept. Somit wird der Nutzer die Lebensmittel nicht verwerten könne und letztendlich werden sie weggeworfen.

Fallback:

Die Suche nach Rezepten sollte erweitert werden, wodurch der Nutzer leider gezwungen ist nochmal einzukaufen. Jedoch konnten somit alle Reste verwertet werden.

4. Datenbank durch den Crawler füllen

Die Rezepte auf der Webseite „chefkoch.de“ werden durch einen Crawler in der Datenbank gespeichert.

Exit:

Der Crawler funktioniert einwandfrei und die Rezepte können konsistent gespeichert werden

Fail:

Die Rezepte sind nicht konsistent und sind somit nicht vergleichbar.

Fallback:

Die Datenbank wird per Hand gefüllt und kann durch den User erweitert werden.