

Architekturbegründung

Unsere Grundarchitektur besteht aus einem Client und einem Server, wobei zusätzlich noch ein mobiler Client berücksichtigt wird.

Client

Der Client beinhaltet eine Präsentationslogik, Anwendungslogik, Kommunikation und Daten/Ressourcen.

Die Präsentationslogik hat die Aufgabe, den Nutzer ein Anpassbares Interface zu bieten. Hierbei hat der Benutzer die Möglichkeit farbliche Änderungen vorzunehmen oder bestimmte Elemente des Interfaces zu verschieben. Grundsätzlich lassen sich bei der Zusammenstellung der Rezepte Filtereinstellungen vornehmen, die ebenfalls durch die Präsentationslogik erfüllt werden. Um den eine Auswahl der Rezepte vornehmen zu können, kann der Benutzer aus den vorgeschlagenen Rezepte auswählen und seiner Woche hinzufügen. Der letzte wichtige Punkt in der Präsentationslogik enthält den der Lebensmittelverwaltung. Hier kann der Benutzer die Lebensmittelreste angeben und verwalten. Die Auswertung dieser Angaben findet jedoch in der Anwendungslogik des Clients statt.

In der Anwendungslogikkomponente werden die angegebenen Lebensmitteldaten ausgewertet und dem Server übertragen. Die Daten werden hierbei auf das ungefähre Ablaufdatum geprüft und falls keine Rezepte zu dem Lebensmittelresten passen, wird der Nutzer darauf aufmerksam gemacht, dass die einzelnen Lebensmittel ablaufen. Eine weitere Anwendung ist die Einkaufszettelgenerierung. Diese filtert die Zutaten der Rezepte heraus und generiert aus den diesen Daten eine Einkaufsliste, die sich an den vorher angegebenen Einkaufstagen anpasst.

Damit der Client nicht ständig am Internet angeschlossen sein muss oder es zu Komplikationen kommt falls das Internet ausfallen sollte, werden alle relevanten Daten auf dem Client lokal gespeichert.

Um den Client zu realisieren, wird als Programmiersprache Java verwendet. Java wurde ausgewählt, da sie Plattformunabhängig anwendbar und objektorientiert ist. Die Vorkenntnisse und die somit geringere Einarbeitungszeit unterstützten die Entscheidung.

Kommunikation

Die Kommunikation zwischen dem Server wird mittels http ermöglicht, da nur die Rezeptdaten, der Lebensmittelbestand sowie die Einkaufsliste dem Client bzw. dem Server übertragen werden muss, reicht eine Client-Server Architektur vollkommen aus. Der Server muss aus der Sicht des Clients nur in der Lage sein, die gefilterten Rezepte zu "liefern".

Als Datenaustauschformat wurde JSON gewählt, da sich dort die Daten einfach und kompakt strukturieren lassen.

Server

Der Server beinhaltet die Komponenten Anwendungslogik, Kommunikation und die Datenbank.

Die Datenbank wird durch MongoDB realisiert. Diese wurde ausgewählt, da sie einfach zu implementieren ist und bei einfachen Datenbankabfragen genauso schnell ist wie MySQL. Da wir nur die Rezepte, saisonale Daten und ggf. die Lebensmittel in der Datenbank speichern und diese nur aus der Datenbank gelesen werden müssen, ist eine relationelle Datenbank nicht von Nöten. Das Filtern dieser Daten ermöglicht die Anwendungslogik des Servers. Dieser Vorgang wird die Filterangaben des Clients realisiert.

Die Lebensmittelreste werden dem Server vom Client übermittelt und von der Anwendungslogik ausgewertet. Hierbei sind die Mengenangabe der einzelnen Lebensmittelreste entscheidend, da über die Mengenangabe und der Kombination der Lebensmittel die Zutaten der einzelnen Rezepte überprüft und bei einem Erfolg den Client gegeben werden kann. Letztendlich sollen noch die saisonalen Informationen berücksichtigt und eingebunden werden, welche Aufschluss darüber gibt ob bestimmte Zutaten in der jeweiligen Zeit vorhanden sind.

Der Server wird mit Node.js programmiert, da dort wenig Einarbeitungszeit benötigt wird und Node.js alle Funktionen bietet, die der Server könne muss.

Mobiler Client

Der Mobile Client hat die Aufgabe den Benutzer in der mobilen Umgebung “zur Seite zu stehen”. D.h. im genaueren, dass der Mobile Client nur dafür benutzt wird die Einkaufsliste, die vorher auf dem nicht mobilen Client erstellt wurde, lokal zu speichern und abrufbar zu machen. Hierbei wird also nur ein http Request benötigt um die Daten vom Server anzufordern und diese dann durch einen Response zu speichern. Es wird somit kein

Der Mobile Client wird so gesehen als verlängerte Arm des Clients verwendet, um den Benutzer auch im Einkaufszentrum mobil die Einkaufsliste anzeigen zu können.

Hinzu kommt, dass der Benutzer die Rezepte, die für die Woche geplant wurden, angucken kann.

Als Betriebssystem wird Android und als Programmiersprache Java benutzt um den mobilen Client zu realisieren.