

# Entwicklungsprojekt interaktive Systeme

*WS 2015/16*

**MS2**

**Teilnehmer:**

Vanakh Chea - 11081999

Manh Linh Bui - 11090174

**Dozenten:**

Prof. Dr. Kristian Fischer

Prof. Dr. Gerhard Hartmann

**Betreuer:**

Robert Gabriel

Jorge Pereira

# Inhaltverzeichnis

## [1. Zielhierarchie](#)

### [1.1. Strategische Ziele](#)

### [1.2. Taktische Ziele](#)

### [1.3. Operative Ziele](#)

## [2. Marktrecherche](#)

### [Zutaten-Matchingsysteme](#)

### [Kochplaner.de](#)

### [Chefkoch.de](#)

## [3. Domänenrecherche](#)

## [4. Alleinstellungsmerkmale](#)

## [5. Methodischer Rahmen \(MCI\)](#)

## [6. Kommunikationsmodell](#)

## [7. Risiken](#)

### [Projektspezifische Risiken](#)

#### [Gebrauchsuntaugliches UI](#)

#### [Netzwerkprobleme](#)

#### [Rechtliche Risiken](#)

### [Projektinterne Risiken](#)

#### [Ausfall eines Teammitgliedes](#)

#### [Zeitliche Knappheit](#)

### [Projektexterne Risiken](#)

## [8. Spezifikation der Proof of Concepts](#)

## [9. Architektur](#)

### [9.1. Architekturdiagramm](#)

### [9.2. Architekturbegründung](#)

# 1. Zielhierarchie

## 1.1. Strategische Ziele

Als strategisches Ziel ist vorgesehen, die Organisation und Planung eines gemeinsamen Kocheventss mit mehreren Personen zu vereinfachen und diese verschiedenen Technologien zu unterstützen. Bestehende Probleme und Hindernisse sollen untersucht und behandelt werden.

## 1.2. Taktische Ziele

Das taktisches Ziel dieses Projektes ist die Entwicklung eines interaktiven System, mit der die Teilnehmer eines gemeinsamen Kochevents die Auswahl eines Rezeptes und die nächsten planerischen Schritte vereinfacht wird.

//wesentliche Funktionen fehlen, Gebrauchstauglichkeit als Ziel

## 1.3. Operative Ziele

Zu den kurzfristigen Zielen gehören Methoden, die während des Projekts durchgeführt werden:

- Abwägung und Auswahl eines geeigneten MCI-Vorgehensmodells und daraus die entsprechenden Aktivitäten und Artefakte ableiten.
- ...
- Entwicklung eines PoC, um die Umsetzbarkeit des Projekts zu prüfen.
- Entwurf und Systemimplementation.
- Durchführung von Tests.

## 2. Marktrecherche

### Zutaten-Matchingsysteme

<http://www.supercook.com/#/recipes>

<http://myfridgefood.com/search-by-ingredients>

<http://www.recipematcher.com/index.php/search/match/>

<http://www.foodpair.com/>

<http://allrecipes.com/> mit App

Die oben genannten Konkurrenzprodukte erfüllen alle den selben Zweck: Der Benutzer teilt dem System mit, welche Zutaten er vorrätig hat und erhält dazu passende Rezepte. Gegebenenfalls kann man sich eine Einkaufsliste ausgeben lassen.

Ein Vorteil dieser Lösungen sind die Unmengen an Rezepte und die attraktive Präsentation.

Die Produkte erfüllen jedoch nur einen Teil der oben definierten Ziele.

### Kochplaner.de

Der Kochplaner unterstützt den Benutzer bei der Erstellung eines wöchentlichen Kochplans. Es werden Rezepte vorgeschlagen, die auf die persönlichen Wünsche abgestimmt sind. Zu den Parametern gehören unter anderem, die Schwierigkeitsstufe, die Zubereitungszeit oder bevorzugte Zutaten.

Das Konkurrenzprodukt richtet sich an Wohngemeinschaften oder Familien, für denen wirtschaftliche Vorteile und der Zeitersparnis erst über einen längeren Zeitraum spürbar(er) sind.

21.10 18:00-18:20

### Chefkoch.de

Die Seite chefkoch.de bietet den Benutzer eine große Anzahl an Rezepten. Der Benutzer hat die Möglichkeit die Internetseite ohne Registrierung zu nutzen und kann sich die

Zutatenliste für die Höhe der Portionenzahl errechnen lassen. Als registrierter Nutzer besitzt man die Möglichkeit Kommentare, Bewertungen zu den Rezepten abzugeben. Außerdem kann man sich eine Einkaufsliste erstellen lassen sowie die Rezepte anderer und eigene in sein persönliches digitale Kochbuch hinzufügen.

- Seite für Rezepte
- Rezept
  - skalierbar
  - votings
  - Step by Step Anleitung
- große Community (sehr bekannt)
  - jeder kann sein Rezept veröffentlichen

### 3. Domänenrecherche

- Kochutensilien
    - Besteck
    - Geschirr
    - Elektronik Geräte (Herd, Backofen, Mikrowelle, etc.)
    -
  - Köche
    - Vermarktung?
  - Kochbücher
    - Rezepte
      - Zutaten
      - Gewürze
        - Ort der Beschaffung
        - Nährwert
    - Nationalgerichte
    - Know-How
  - Veranstaltungsort
  - Vermarktung
-

- Zeit
- Benutzung
  - bei dreckigen Händen
    - Stimme?
    - Luft Gesten
    - Touch
- kulturelle Unterschiede
  - Art und Weise zu würzen
  - Maßeinheiten
  -

## 4. Alleinstellungsmerkmale

Nachdem im vorherigen Kapitel diverse Konkurrenzprodukte vorgestellt wurden und die Gemeinsamkeiten identifiziert sind, werden im folgenden Kapitel die Alleinstellungsmerkmale erläutert.

Die Idee des Zutatenmatching, wie es im vorherigen Kapitel beschrieben wurde, wird in diesem Projekt aufgegriffen und mit folgenden Merkmalen ergänzt:

### **Beteiligung mehrerer Personen**

Wie bereits in der Zielhierarchie erwähnt wurde, müssen mehrere Benutzer in den Vorbereitungsprozess eines Kocheventss mit einbezogen werden.

(...)

### **Abstimmung**

An einer bestimmten Stelle des Interaktionsablaufs erhält eine Gruppe von Dienstnutzer (hier: Gastgeber und Gäste) mehrere Rezeptionen, über die sie abstimmen können.

Die Abstimmung ist insofern praktisch, dass sie ortunabhängig durchgeführt werden kann.

### **Kostenersparnis**

Dieses System zielt darauf ab, bereit vorhandene Zutaten zu verwenden.

### **Kulturelle Erfahrung**

---

Ein Mehrwert wird sichtbar, wenn kulturell heterogene Gruppen einen Kochevent veranstalten wollten und das System nach dem Zutatenmatching ebenfalls kulturell abwechslungsreiche Rezepte empfiehlt.

### **Bereitstellung von Metainformationen**

Sofern die Geschäftsobjekte, sprich: die Rezepte, noch nicht qualitativ mit zusätzlichen Informationen angereichert sind, geschieht dies automatisch im Hintergrund.

Die Anwendung prüft beispielsweise die Zutatenliste auf Zutaten, die für die Ernährung des Benutzers ungeeignet sind. Davon betroffen sind beispielsweise Allergiker, Vegetarier oder laktoseintolerante Menschen.

## **5. Methodischer Rahmen (MCI)**

Die Auswahl eines geeigneten MCI Vorgehensmodells legt die Basis für den Entwicklungsprozess.

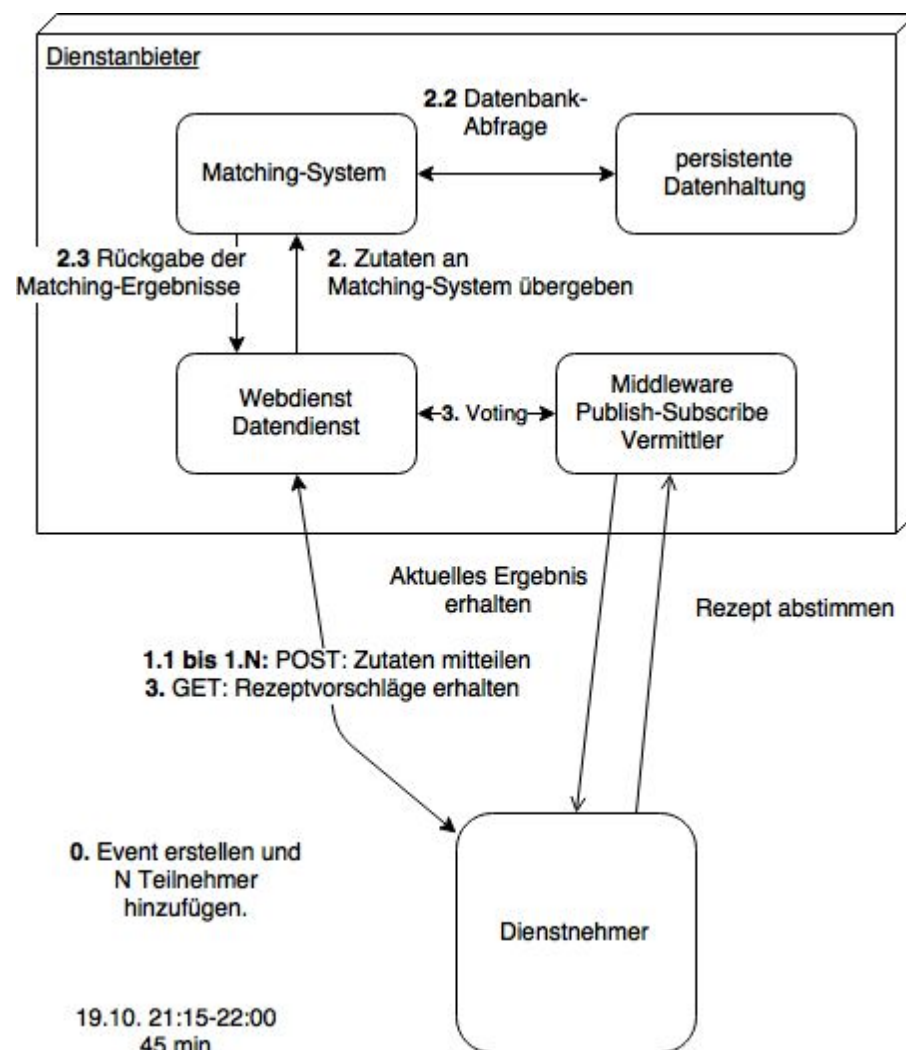
- nutzerzentriert oder nutzungszentriert Gestaltung?
- usercentered design: individuelle Interessen,
  
- Projektspezifisch
  - Aktivitäten
    - ISO 9241 (Teil 110 und 210) anschauen  
<http://www.procontext.com/aktuelles/2010/03/iso-9241210-prozess-zur-entwicklung-gebrauchstauglicher-interaktiver-systeme-veroeffentlicht.html>
    -
  - Methoden: methodische spezifischer Rahmen (MCI)
    - Beispiel
    - usercentered design:  
[https://de.wikipedia.org/wiki/Nutzerorientierte\\_Gestaltung](https://de.wikipedia.org/wiki/Nutzerorientierte_Gestaltung) angelehnt an die ISO 9241 210

- usability engineering lifecycle mayhew: komplex, zeitaufwendig,  
<http://wikis.hawk-hhg.de/wikis/fields/usability/field.php/Entwicklungsprozess/BenutzerzentrierterEntwicklungsprozess>
- Discount Usability: ungeeignet, da Usability Experten notwendig sind.  
Aus Kostengründen nicht tragbar.
- Scenario Based Usability Engineering:

**Kontextabhängige Gestaltung:** Die Kontextabhängige Gestaltung eignet sich sehr gut für Projekte mit vielen unterschiedlichen Zielgruppen, was nicht der Fall bei dem zu entwickelnden System ist. Außerdem ist dieses Modell für die Kommunikation im Team und mit den Kunden ausgelegt. Da das zu entwickelnde System keine Kunden hat, ist das Modell ungeeignet.

## 6. Kommunikationsmodell

<https://www.draw.io/#G0Bz9suCpEjxisaktYSF9KU0FqZE0>





grafik: 19.10: 21:15-22:00

text: 20.10 15:20-16:00

22.10 - 15min

Im folgenden Kapitel wird die Kommunikation zwischen dem Dienstgeber und den Dienstnehmer erläutert. Der Dienstnehmer kann sich, abhängig vom Anwendungsszenario, für eine Rolle entscheiden, die sich im Kommunikationsablauf jedoch nur geringfügig unterscheiden.

Die Dienstnehmer "Gastgeber" und "Gast" übermitteln jeweils eine **Liste mit den vorrätigen Zutaten** an den Dienstgeber. Es findet also **Komposition** statt, die von einem Nachrichtenvermittler verwaltet wird.

//In Echtzeit werden sämtliche Teilnehmer über Veränderungen informiert.

Das **Matching-System** des Dienstgebers ermittelt mithilfe dieser Informationen passende Rezepte und übermittelt diese an die Clients. Dabei greift er auf eine Datenbank oder 3rd-Party-APIs zu. (Hinweis auf Nutzungsrechte, PoC)

Die Dienstnehmer stimmen ab und das populärste Rezept mit den meisten Stimmen "gewinnt". An dieser Stelle muss man anmerken, dass die Abstimmungen nicht zeitgleich stattfinden müssen. (Pub/Sub -> Anforderung für PoC)

Am Ende wertet der Dienstgeber das Ergebnis aus und teilt jedem Dienstnehmer mit, welche Zutaten er zum Kochevent mitbringen soll. Notfalls müssen weitere Zutaten gekauft werden. Auf der Dienstnehmer-Seite wird das Rezept mit Informationen angereichert. (->Argument für User-centered Design)

## 7. Risiken

18.09. 30min.

21.09. 17:45-18:00

In diesem Kapitel werden die potenzielle Risiken dargelegt, die während es Entwicklungsprozesses auftreten oder von projektspezifischer Natur sind. Die Ergebnisse des PoC liefern Anhaltspunkte darüber, welche Risiken in welchem Umfang auftreten können und ob diese minimiert werden können.

- *gesellschaftliche Akzeptanz*
- *Hohe Entwicklungskosten (finanzielle Rahmensprengung)*
- *Technische Hindernisse*
- *Komplexität*

### Projektspezifische Risiken

#### Gebrauchsuntaugliches UI

Der Benutzer muss das Gefühl haben, dass er Interaktionen mit dem System effektiv, effizient und zufriedenstellend erledigen kann. Welchen Nutzen bringt ein System, welches zwar alle funktionalen Anforderungen erfüllt, aber eine mangelhafte Benutzungstauglichkeit aufweist?

#### Netzwerkprobleme

Ein verteiltes System setzt ein zuverlässiges Netzwerk voraus. Ansonsten kann die Anwendung nur unvollständig oder überhaupt nicht ausgeführt werden.

#### Rechtliche Risiken

Es muss geklärt werden, ob man eigene Rezepte in das eingepflegt werden sollen oder auf Fremdanbieter, beispielsweise über bestehende APIs, zurückgegriffen werden darf.

## **Projektinterne Risiken**

### **Ausfall eines Teammitgliedes**

Durch den Ausfall eines Teammitglieds während des Entwicklungsprozesses, kann es zu erheblichen Verzögerungen bei der Entwicklung kommen. Die Folgen wären ein zeitlicher Rückschlag bis zur Fertigstellung der Software.

### **Zeitliche Knappheit**

Erfordert eine Entwicklungsphase unerwartet mehr Zeit, müssen Wege gefunden werden, die ein Überschreiten der finalen Deadline verhindern können.

- kosten?

## **Projektexterne Risiken**

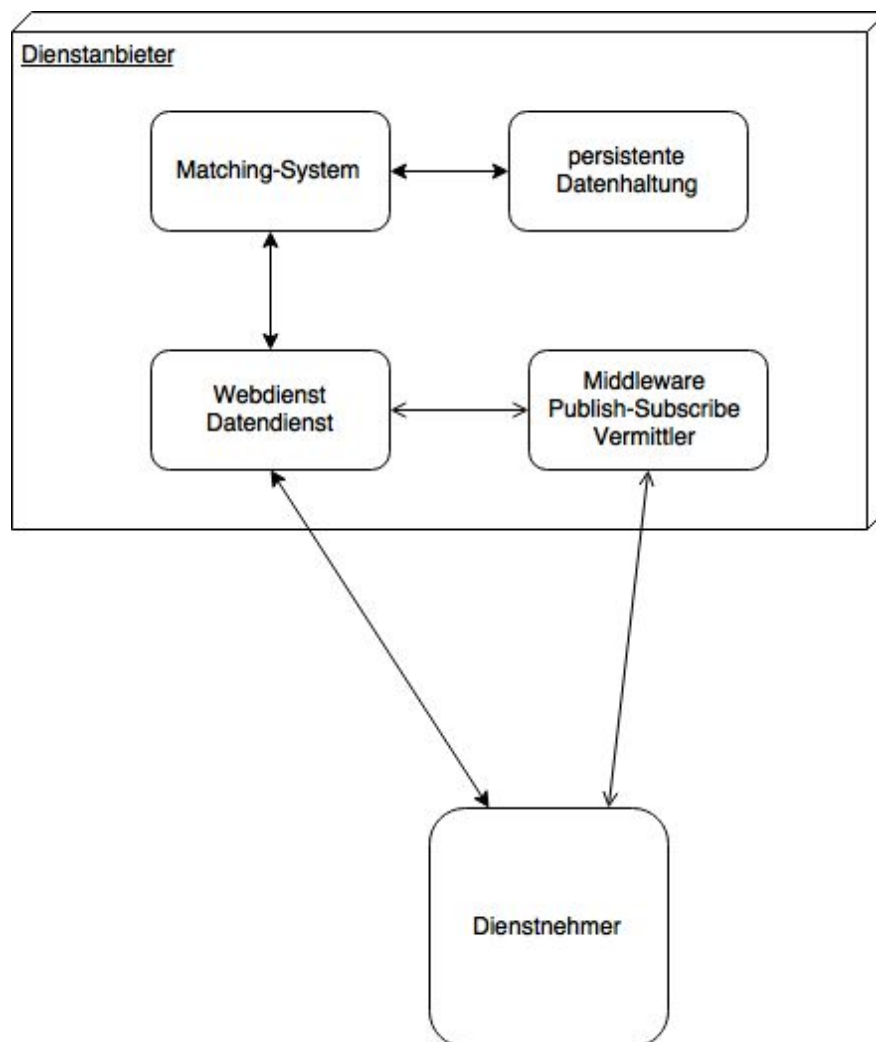
- jemand entwickelt dieselbe Idee und veröffentlicht diese vorher
- gesellschaftliche Akzeptanz //super kritisch, womöglich ein KO-Kriterium
- Inkompatibilität mit einzelnen Mobilgeräten

## **8. Spezifikation der Proof of Concepts**

## 9. Architektur

<https://www.draw.io/#G0Bz9suCpEjxisdXMyZ3hQUzJtekU>

<http://www.javaworld.com/article/2079190/scripting-jvm-languages/6-things-you-should-know-about-node-js.html>



//nicht fertig

### 9.1. Architekturdiagramm

Client: GUI, Chatclient?, Publish-Subscribe Sender und Listener, REST-Client, Datenaufbereitung

Server: Persistente Datenhaltung, Publish-Subscribe Listener

---

22.10. 17:30-18:30

### **Matchingkomponente**

Die Aufgabe dieser Komponente ist, die Zutaten der Teilnehmer eines Kochevents zu sammeln und Datenbankabfragen durchzuführen. Es sollen passende Rezepte zurückgegeben werden.

### **Persistente Datenhaltung**

Die persistente Datenhaltung wird aus einer Datenbank bestehen, in der alle Daten abgespeichert werden.

**// Welche Daten? im Kommunikationsmodell oder MS4 unter Datenstrukturen erläutern?**

### **Lokalisierungskomponente**

Seitens des Dienstnehmers arbeitet diese Komponente. Konkret bedeutet dies, dass Umwandlungen an den Temperatur- und Maßeinheiten an die vorherrschenden lokalen Gegebenheiten vorgenommen werden.

//Zusätzliche Infos; unklar, wie detailliert die Lokalisierung stattfinden soll

### **Dienst**

### **Dienstnehmer**

## **9.2. Architekturbegründung**