

Sammlung von Kochrezepten mit Filterung

Dokumentenentwicklung

Version	Beschreibung	Datum
0.1	Erste Erstellung des Dokuments.	27.04.2015
0.2	Erweiterung mit Usecase- Diagramm, textueller Funktionsbeschreibung und GUI Skizzen	03.05.2015
0.3	Sequenz Diagramme für Funktionen und ER Diagramm für DB	18.05.2015

Version	Autor	Dateiname	Letzte Änderung	Druckdatum	Seite
1.0		Wifi_Projektdokumentation _Tobolka_2605	26.05.2015	26.05.2015	1 / 19

Inhaltsverzeichnis

1	Vision	3
1.1	Vision und Kurzbeschreibung des Projekts	3
2	Grobe Spezifikation	3
2.1	Zusammenhang mit bereits bestehenden Systemen	3
2.2	Überblick über die geforderte Funktionalität.....	3
2.3	Wesentliche Qualitätsanforderungen und Rahmenbedingungen	4
3	Detaillierte Spezifikation.....	4
3.1	Akteure des Systems (Personas)	4
3.2	Detaillierte Funktionale Anforderungen (Szenarios & Screens)	4
3.2.1	Namenssuche für Rezepte	5
3.2.2	Manuelle Filterung	6
3.2.3	Verwendung von Custom-Filtern	6
3.2.3.1	Filter erstellen	6
3.2.3.2	Filter bearbeiten	6
3.2.3.3	Filter löschen	7
3.2.4	Neue Zutaten anlegen	7
3.2.5	Liste gefundener Rezepte anzeigen.....	7
3.2.6	Einzelnes Rezept anzeigen.....	7
3.2.7	Neues Rezept anlegen	8
3.3	Schnittstellen.....	8
3.3.1	Benutzerschnittstellen (GUI).....	8
3.3.1.1	Start-Bildschirm mit Namens-Suche	8
3.3.1.2	Suche verfeinern.....	9
3.3.1.3	Zutaten-Selektor.....	10
3.3.1.4	Liste gefundener Rezepte	10
3.3.1.5	Rezeptanzeige	11
3.3.2	Systemschnittstellen.....	12
3.4	Nicht-Funktionale Anforderungen	12
3.4.1	Vorgaben zu Hardware und Software	12
3.4.2	Performance	12
3.4.3	Resources und Anforderungen an die Hardware	12
3.4.4	Security & Safety	12
3.4.5	Reliability	12
3.4.6	Maintenance.....	12
3.4.7	Portability / Skalierbarkeit / Wiederverwendbarkeit	12
3.4.8	Useability	12
3.5	Systemabgrenzung, Systemarchitektur und Datenhaltung	13
3.6	Rahmenbedingungen	19
4	Begriffsbestimmungen und Abkürzungen	19
5	Benutzerhandbuch.....	19

Version	Autor	Dateiname	Letzte Änderung	Druckdatum	Seite
1.0		Wifi_Projektdokumentation _Tobolka_2605	26.05.2015	26.05.2015	2 / 19

1 Vision

1.1 Vision und Kurzbeschreibung des Projekts

Ziel ist die Sammlung von Kochrezepten, die teils in Form von Artikeln im TOUR Magazin vorliegen, teils auf „fliegenden Zetteln“ und anderen Quellen, in einer zentralen Datenbank. Es kann gezielt nach Rezepten gefiltert werden, die bestimmte Kriterien erfüllen. Dabei wird es voreingestellte Filter geben zum Beispiel „Low-Carb“ Rezepte für Eiweiß-reiche und gleichzeitig Kohlehydrat-arme Kost vs. Rezepte mit reichlich Kohlehydraten für Trainings- oder Touren-Tage. Oder Vegetarische/Nicht vegetarische Rezepte. Man kann nach Rezepten mit bestimmten Zutaten suchen, oder bestimmte Zutaten ausschließen (Zum Beispiel „kein Paprika“). Als Benutzer soll man mehrere individuelle Filter (also Kombinationen aus bestimmten Kriterien) speichern können.

In der ersten Version wird eine Rezeptdatenbank erstellt und die dazugehörige Software mit übersichtlicher grafischer Benutzeroberfläche für die Abfrage, Filterung und Darstellung (und Eingabe) von Rezepten.

Mögliche Erweiterungen für spätere Versionen sind eine Web-Version des Programms und die Möglichkeit, automatisch Einkaufslisten mit einer Einkaufslisten-App zu erstellen.

Die Besonderheit des Projekts liegt in den umfangreichen-Filterungsmöglichkeiten.

2 Grobe Spezifikation

2.1 Zusammenhang mit bereits bestehenden Systemen

Das Projekt wird als stand-alone Software implementiert. Es besteht kein Zusammenhang mit bestehenden Systemen.

2.2 Überblick über die geforderte Funktionalität

Rezepte aus verschiedenen Ausgaben des TOUR Magazins (Fachmagazin für Rennrad-Radsport) und aus anderen Quellen (Ausdrucke aus dem Internet, teils handgeschriebene Rezepte und von Kochkursen kopierte Rezepte) werden in einer Datenbank zentral und dauerhaft gespeichert.

Dazu kommt die Möglichkeit, dort mittels verschiedener Filter-Kriterien nach Rezepten zu suchen. Es werden vorgefertigte Filter implementiert (Low-Carb Rezepte für die schlanke Linie, Kohlehydrat-reiche Kost für Trainings-Tage oder Tage vor Touren, vegetarische Rezepte).

Rezepte können nach bestimmten Zutaten gefiltert werden (Zum Beispiel: „Rezept mit Rindfleisch“), oder auch nach bestimmten Ausschlusskriterien („kein Paprika“).

Nach der Suche soll eine Liste mit Rezepten erscheinen, die die Suchkriterien erfüllen. Das gewählte Rezept wird danach übersichtlich dargestellt.

Insgesamt soll auch die Rezeptsammlung übersichtlich sein. Ziel ist nicht eine möglichst große Rezeptsammlung mit hunderten oder gar tausenden Rezepten. Sondern eine Sammlung, die groß genug ist, um genügend Abwechslung zu bieten und mit der es Spaß macht, nach einem Rezept zu suchen.

Version	Autor	Dateiname	Letzte Änderung	Druckdatum	Seite
1.0		Wifi_Projektdokumentation _Tobolka_2605	26.05.2015	26.05.2015	3 / 19

2.3 Wesentliche Qualitätsanforderungen und Rahmenbedingungen

Die Software wird in Java geschrieben (JDK 1.8) bei der Datenbank handelt es sich um eine relationale Derby Datenbank. Datenbank und Software kommunizieren über JDBC miteinander und laufen auf Rechnern, auf denen auch Java 1.8 (JRE 1.8) läuft.

Die Anforderung an die Hardware entsprechen den Anforderungen, die von Java 1.8 vorausgesetzt werden.

Es ist vereinbart, dass das fertige Projekt am 22.06.2015 abgegeben wird und am 29.06.2015 live (am Rechner) präsentiert wird.

3 Detaillierte Spezifikation

3.1 Akteure des Systems (Personas)

Die Software soll in der ersten Version von einem beliebigen menschlichen User benutzt werden können. Dieser interagiert über eine grafische Benutzeroberfläche (GUI) mit der Software, welche wiederum mit der Datenbank interagiert, in der die Rezepte gespeichert sind. In Kurzform ist im Weiteren die Rede der Personas User, System und Datenbank.

3.2 Detaillierte Funktionale Anforderungen (Szenarios & Screens)

Zunächst werden alle Funktionen grob gegliedert in einem Usecase Diagramm (Abbildung 1) dargestellt. Auf einzelne Methoden wird dabei nicht eingegangen. Das System, Benutzer und die verbundene Datenbank werden dargestellt, sowie die Funktionen, die das System leisten kann und nicht zuletzt die Beziehungen dieser einzelnen Elemente zueinander.

Die einzelnen Funktionen werden danach in den Abschnitten 3.2.1-3.2.7 Form von Tabellen geschildert, um Struktur und Übersichtlichkeit zu gewährleisten.

Version	Autor	Dateiname	Letzte Änderung	Druckdatum	Seite
1.0		Wifi_Projektdokumentation _Tobolka_2605	26.05.2015	26.05.2015	4 / 19

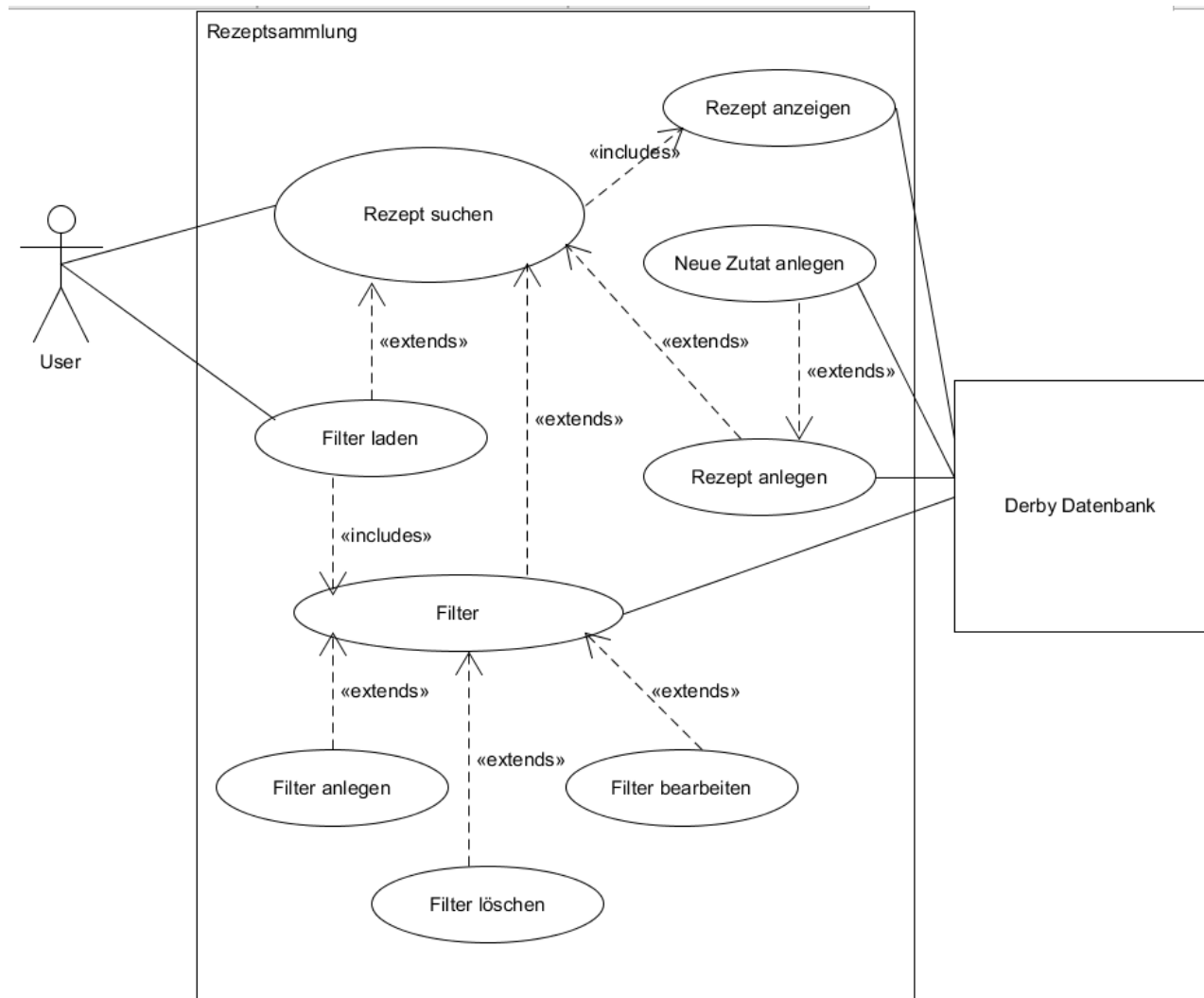


Abbildung 1 Usecase Diagramm Rezeptsammlung

3.2.1 Namenssuche für Rezepte

Titel	<i>Namenssuche für Rezepte</i>
Kurzbeschreibung	<i>Im Suchfeld gibt der User ein, für welches Gericht er ein Rezept sucht.</i>
Aktoren	<i>User, System, DB</i>
Vorbedingungen (Eingabe)	<i>User ist auf Startseite</i>
Beschreibung des Ablaufs (Verlauf)	<i>Im Suchfeld gibt der User zum Beispiel „Gulasch“ ein. Und bestätigt das mit anklicken des Suchen Buttons</i>
Auswirkungen (Ausgabe)	<i>Daraufhin wird eine Liste aller Rezepte ausgegeben, auf die die Bezeichnung Gulasch zutrifft. Zum Beispiel Rindsgulasch und Schweinsgulasch.</i>
Anmerkungen	

Version	Autor	Dateiname	Letzte Änderung	Druckdatum	Seite
1.0		Wifi_Projektdokumentation_Tobolka_2605	26.05.2015	26.05.2015	5 / 19

3.2.2 Manuelle Filterung

Titel	Manuelle Filterung
Kurzbeschreibung	<i>User wählt aus verschiedenen Kriterien optional aus und kann nach Gerichten mit bestimmten Zutaten suchen oder nach Gerichten ohne diese Zutaten.</i>
Aktoren	<i>User, System, DB</i>
Vorbedingungen (Eingabe)	<i>User ist auf Start</i>
Beschreibung des Ablaufs (Verlauf)	<i>User klickt auf Startseite auf „Suche verfeinern“</i>
Auswirkungen (Ausgabe)	<i>Ein GUI mit verschiedenen Kriterien „Low-Carb“, „Kohlehydrat-reich“ und verschiedenen Zutaten „Rind“, „Huhn“, „Paprika“ erscheint, die per Klick inkludiert oder ausgeschlossen werden könnten.</i>
Anmerkungen	

3.2.3 Verwendung von Custom-Filtern

3.2.3.1 Filter erstellen

Titel	<i>Filter erstellen</i>
Kurzbeschreibung	<i>Wie manuelle Filterung. Aber es können bestimmte Vorlieben berücksichtigt werden, indem eine bestimmte Kombination aus Kriterien und Zutaten gespeichert wird.</i>
Aktoren	<i>User, System, DB</i>
Vorbedingungen (Eingabe)	<i>User klickt auf Filter erstellen</i>
Beschreibung des Ablaufs (Verlauf)	<i>User wählt Kriterien analog manueller Filterung. Klickt auf Filter speichern. Dialog erscheint, der User auffordert Filter zu speichern.</i>
Auswirkungen (Ausgabe)	<i>Gespeicherte Filter können direkt von Startseite aus per Klick abgerufen werden.</i>
Anmerkungen	

3.2.3.2 Filter bearbeiten

Titel	<i>Filter bearbeiten</i>
Kurzbeschreibung	<i>Analog Filter erstellen</i>
Aktoren	<i>User, System, DB</i>
Vorbedingungen (Eingabe)	<i>User wählt aus Liste Filter zur Bearbeitung.</i>
Beschreibung des Ablaufs (Verlauf)	<i>GUI wie bei manueller Filterung erscheint, bestimmte Kriterien bereits aktiv. Diese können deaktiviert, andere hinzugefügt werden.</i>
Auswirkungen (Ausgabe)	<i>Danach wird User gefragt, ob er die Änderungen speichern will. Filter mit Änderung wird ggf. gespeichert</i>
Anmerkungen	

Version	Autor	Dateiname	Letzte Änderung	Druckdatum	Seite
1.0		Wifi_Projektdokumentation _Tobolka_2605	26.05.2015	26.05.2015	6 / 19

3.2.3.3 Filter löschen

Titel	<i>Filter löschen</i>
Kurzbeschreibung	<i>Gespeicherte Filter werden wieder gelöscht</i>
Aktoren	<i>User, System, DB</i>
Vorbedingungen (Eingabe)	
Beschreibung des Ablaufs (Verlauf)	<i>Analog Filter laden, nur erst auf Filter Bearbeitungsseite. Kunde wählt Filter aus Liste aus. Klickt auf löschen.</i>
Auswirkungen (Ausgabe)	<i>Dialog fragt Kunden „Sind sie sicher?“.</i>
Anmerkungen	

3.2.4 Neue Zutaten anlegen

Titel	<i>Neue Zutat</i>
Kurzbeschreibung	<i>Vermisst der User eine bestimmte Zutat, kann er sie neu anlegen</i>
Aktoren	<i>User System DB</i>
Vorbedingungen (Eingabe)	<i>User ist auf der Seite „Zutaten Selektor“</i>
Beschreibung des Ablaufs (Verlauf)	<i>Über einen Button „neue Zutat“ kann er eine solche Anlegen</i>
Auswirkungen (Ausgabe)	<i>Die Zutatenliste im Zutaten-Selektor wird größer</i>
Anmerkungen	

3.2.5 Liste gefundener Rezepte anzeigen

Titel	<i>Liste gefundener Rezepte anzeigen</i>
Kurzbeschreibung	<i>Die gefundenen Rezepte werden angezeigt als Liste</i>
Aktoren	<i>User System DB</i>
Vorbedingungen (Eingabe)	<i>User hat entweder mit Namens-Suche oder per Filter nach Rezepten gesucht</i>
Beschreibung des Ablaufs (Verlauf)	<i>Nachdem der User mit oder ohne Filter (manuell oder gespeicherter) auf „Rezept suchen“ klickt,</i>
Auswirkungen (Ausgabe)	<i>wird eine Liste mit Rezepten angezeigt, auf die die Suchkriterien zutreffen</i>
Anmerkungen	

3.2.6 Einzelnes Rezept anzeigen

Titel	<i>Anzeige Einzel-Rezept</i>
Kurzbeschreibung	<i>Details zu einem einzelnen Rezept werden angezeigt</i>
Aktoren	<i>User, System, Datenbank</i>
Vorbedingungen (Eingabe)	<i>User hat Rezept gesucht, bekommt Such-Ergebnis angezeigt</i>
Beschreibung des Ablaufs (Verlauf)	<i>User klickt ein Rezept auf der Liste gefundener Rezepte an,</i>
Auswirkungen (Ausgabe)	<i>Details dazu werden angezeigt. Zutaten, Kochinstruktionen</i>

Version	Autor	Dateiname	Letzte Änderung	Druckdatum	Seite
1.0		Wifi_Projektdokumentation _Tobolka_2605	26.05.2015	26.05.2015	7 / 19

3.2.7 Neues Rezept anlegen

Der User kann auch ein neues Rezept anlegen. Über den Button neues Rezept anlegen, der unterhalb der Liste gefundener Rezepte angezeigt wird. Dieser Button ist dort platziert, weil primär nach vorhandenen Rezepten gesucht werden soll.

Titel	<i>Neues Rezept anlegen</i>
Kurzbeschreibung	<i>User kann ein neues Rezept eintragen</i>
Aktoren	<i>User, System, Datenbank</i>
Beschreibung des Ablaufs (Verlauf)	<i>User klickt Button „neues Rezept anlegen“. Zutaten-Selektor- Ansicht erscheint. Nachdem der User alle Zutaten gewählt hat bzw. wenn nötig neu angelegt hat, gibt er noch die Koch Instruktionen in ein Textfeld ein. Danach wählt er zwischen speichern und abbrechen.</i>
Auswirkungen (Ausgabe)	<i>Rezept-Sammlung erweitert sich. Neues Rezept wird angezeigt</i>
Anmerkungen	<i>GUI sieht genauso aus wie die Rezept-Anzeige nur mit Button zum Speichern, daher kein eigenes Bild hier</i>

3.3 Schnittstellen

3.3.1 Benutzerschnittstellen (GUI)

Die grafischen Benutzerschnittstellen (GUI) sind hier mehr oder weniger Deckungsgleich mit den in Abschnitt 2 beschriebenen Funktionen der Software. Nachdem sie damit zentraler Bestandteil sind, werden sie hier detailliert in Form von Mockups (Prototypen ohne Funktion, die vom Aussehen her aber dem späteren Original entsprechen sollen) abgebildet.

3.3.1.1 Start-Bildschirm mit Namens-Suche

Hermanns Rezeptsammlung mit Spezialfilter

Willst du einfach schnell ein Rezept suchen?

Wir helfen dir, das passende Rezept zu finden.
Ob low-carb oder zur Auffüllung des KH Speichers, vegetarisch, oder mit Fleisch
Wenn du heute einfach Lust auf bestimmte Zutaten hast (oder andere vermeiden willst)

Hier wird dir geholfen:

Für Stammgäste: Deine Vorlieben sind schon gespeichert? Lade hier deinen gespeicherten Spezial Filter

▼

Abbildung 2 Startseite

Über die schnelle Namenssuche kann der User zum Beispiel direkt nach "Gulasch" suchen und erhält eine Liste gefundener Rezepte, in dem Fall verschiedene Gulasch Varianten. Über Filter laden kommt der User schnell zu vorgespeicherten Filtern. Über "Suche verfeinern" geht es weiter zur manuellen Filterung und Erstellung persönlicher Filter

Version	Autor	Dateiname	Letzte Änderung	Druckdatum	Seite
1.0		Wifi_Projektdokumentation _Tobolka_2605	26.05.2015	26.05.2015	8 / 19

3.3.1.2 Suche verfeinern

Suche verfeinern

Welche Zutaten dürfen es heute sein? Auf der nächsten Seite kannst du wählen

Zum Zutaten Selektor

Vorlieben können sich auch ändern. Hier kannst du deshalb deine Spezialfilter verwalten

Filter ▼

Filter bearbeiten

Filter erstellen

Filter löschen

Zurück

Exit

Abbildung 3 Suche verfeinern

Version	Autor	Dateiname	Letzte Änderung	Druckdatum	Seite
1.0		Wifi_Projektdokumentation _Tobolka_2605	26.05.2015	26.05.2015	9 / 19

3.3.1.3 Zutaten-Selektor

Zutaten Selektion

Hier kannst du optional eine Kategorie wählen (oder mehrere)

☐ low Carb ☐ Kohlehydrat-reich ☐ vegetarisch ☐ fettarm

Es geht auch noch genauer. Such dir aus, auf welche Zutaten du so richtig Gusto hast, und welche du weglassen willst

Ja bitte! Immer her damit Nein danke, bitte weglassen!

☐ Rind ☐ Zwiebel
☒ Huhn ☒ Paprika
☐ Paprika ☐ Broccoli
☐ Pfeffer -[] Tofu
-x Chili -x Zucchini
-[] Knoblauch

Auswahl als Filter speichern **Zeig mir Rezepte** **Zurück**

Abbildung 4 Zutaten-Selektor

Hier kann nach Kategorien gefiltert werden, wie „low Carb“, Kohlehydrat-reich oder vegetarisch. Zutaten können ausgeschlossen werden und man kann nach Rezepten mit bestimmten Zutaten suchen. Die Selektion kann man auch gleich (auch ohne den Filter erstellen Button geklickt zu haben) als individuellen Filter speichern. Über einen weiteren Button werden die Ergebnisse angezeigt. Die Anforderungen für manuelle Filterung und Erstellung eines neuen permanenten Filters sind weitgehend gleich. Daher reicht dafür eine Benutzeroberfläche aus. Über die man auch noch neue Zutaten anlegen kann.

3.3.1.4 Liste gefundener Rezepte

In einer Liste werden alle Rezepte angezeigt, die den Suchkriterien entsprechen. Wählt man eines aus, kann man es sich detailliert anzeigen lassen. Über „neues Rezept anlegen“ erstellt man ein neues Rezept.

Version	Autor	Dateiname	Letzte Änderung	Druckdatum	Seite
1.0		Wifi_Projektdokumentation _Tobolka_2605	26.05.2015	26.05.2015	10 / 19

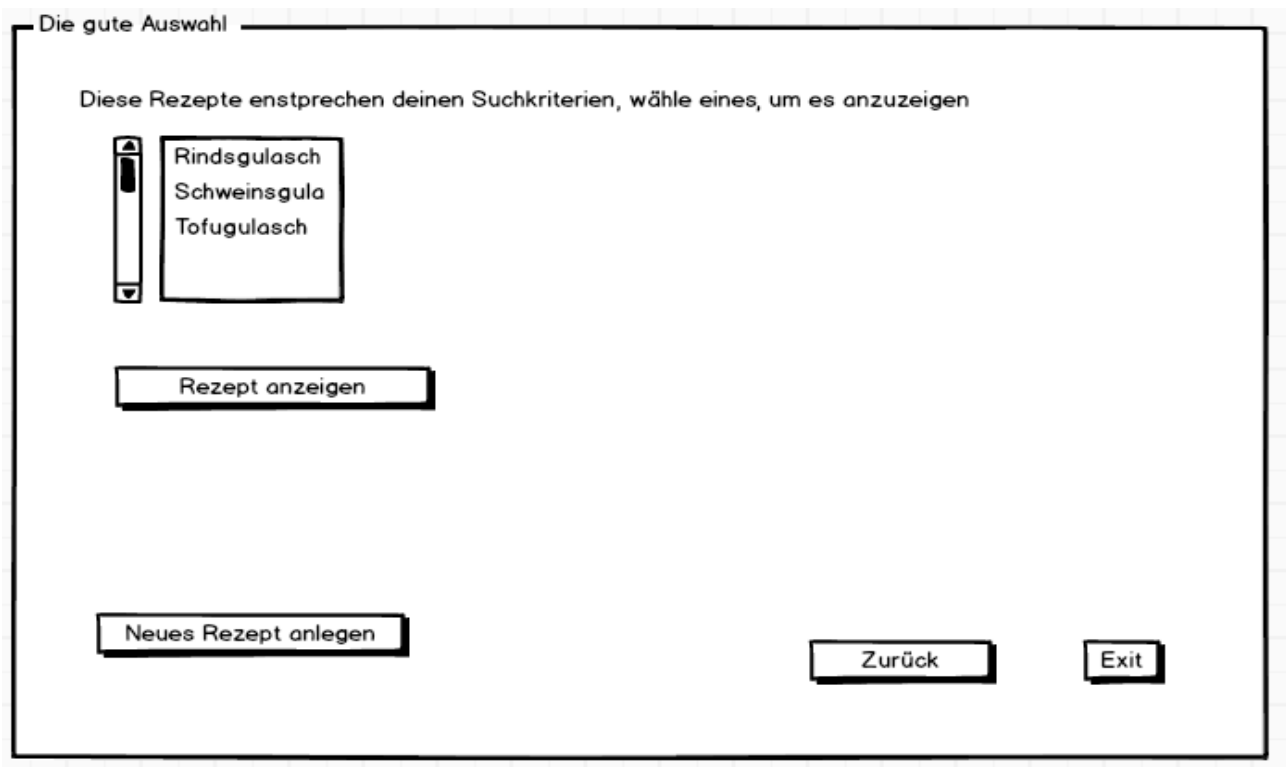


Abbildung 5 Liste gefundener Rezepte

3.3.1.5 Rezeptanzeige

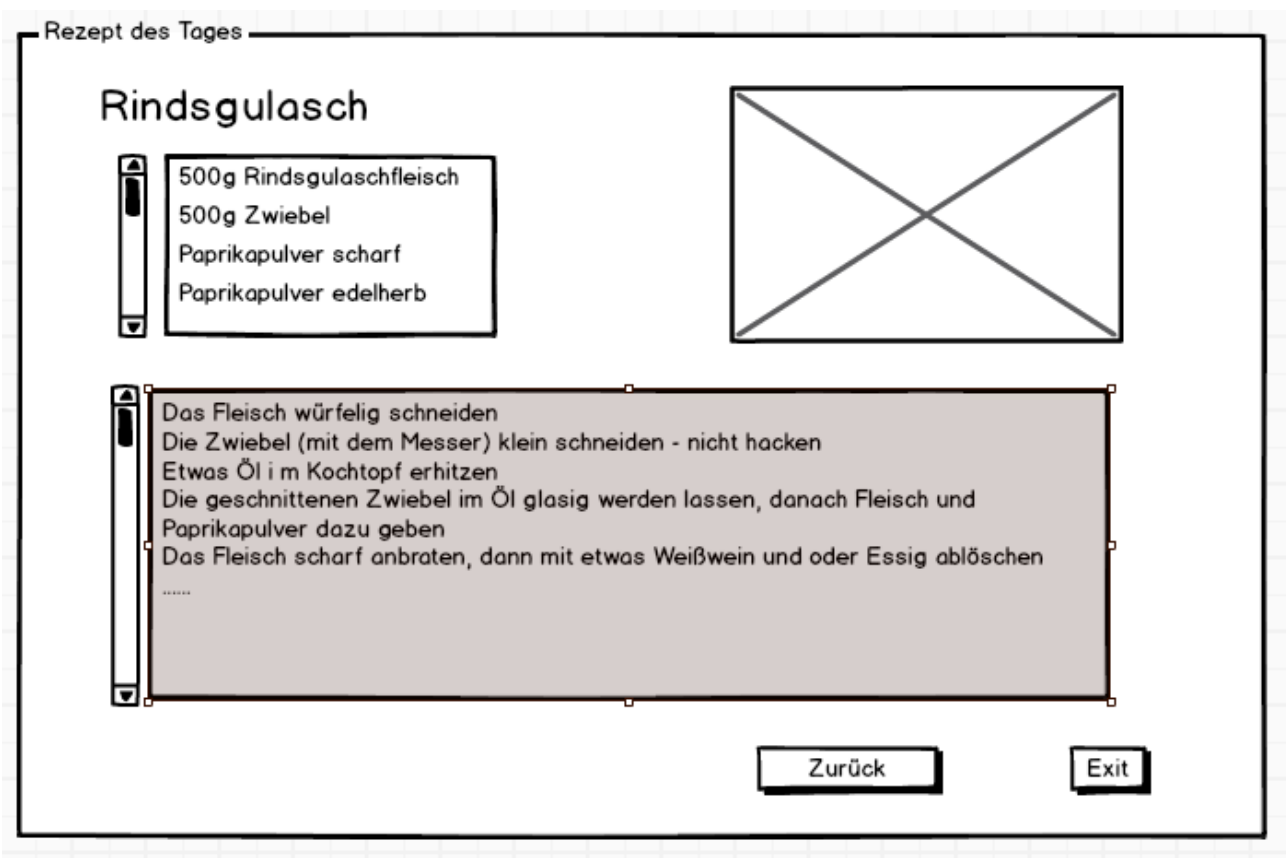


Abbildung 6 Rezept-Anzeige

Version	Autor	Dateiname	Letzte Änderung	Druckdatum	Seite
1.0		Wifi_Projektdokumentation _Tobolka_2605	26.05.2015	26.05.2015	11 / 19

3.3.2 Systemschnittstellen

Schnittstellen zu anderen Systemen sind in der ersten Version keine vorgesehen.

3.4 Nicht-Funktionale Anforderungen

3.4.1 Vorgaben zu Hardware und Software

Das Programm wird auf einem Rechner mit Windows 8.1 64 Bit entwickelt, mit Java JDK 1.8. Es soll auf jedem Rechner lauffähig sein, auf dem Java 1.8 läuft (auf dem JRE 1.8 installiert ist).

3.4.2 Performance

An die Performance werden keine besonderen Anforderungen gestellt.

3.4.3 Resources und Anforderungen an die Hardware

Die Datenmenge insgesamt fällt recht gering aus. Die verwendeten Grafiken sind nicht aufwendig und benötigen keine besonderen Ressourcen.

Die verwendete Derby Datenbank stellt ebenfalls keine besonderen Anforderungen an die Hardware.

3.4.4 Security & Safety

Hinsichtlich Security und Safety bestehen in der hier geschilderten ersten Version keine besonderen Anforderungen.

3.4.5 Reliability

Die Software soll natürlich zuverlässig funktionieren und dem User unnötigen Ärger ersparen. Es ist aber kein kritisches System – es bestehen also keine besonderen Anforderungen im Bereich Reliability.

3.4.6 Maintenance

3.4.7 Portability / Skalierbarkeit / Wiederverwendbarkeit

Keine besonderen Anforderungen. Wartbarkeit ist durch eine genaue Dokumentation gewährleistet und durch Objekt-orientiertes Design. Mit logischen und sinnvollen Klassen, sowie Methoden mit aussagekräftiger Benennung. Der Source Code wird kommentiert, was die Wartung erleichtert.

3.4.8 Useability

Keine besonderen Anforderungen. Die einzelnen Funktionen sollen intuitiv verständlich sein. Für den User gibt es die Möglichkeit, schnell und direkt über die Namenssuche ein Rezept zu suchen, oder gleich einen gespeicherten Filter zu laden. Mit Zutaten-Listen wird er nur konfrontiert, wenn er auf "Suche verfeinern" klickt.

Version	Autor	Dateiname	Letzte Änderung	Druckdatum	Seite
1.0		Wifi_Projektdokumentation _Tobolka_2605	26.05.2015	26.05.2015	12 / 19

3.5 Systemabgrenzung, Systemarchitektur und Datenhaltung

Die Rezeptsammlung ist als stand-alone Software konzipiert und damit nicht in andere Systeme eingebettet oder von diesen abhängig.

Die System-Architektur besteht grob aus 3 Schichten (Datenbank, Logik, Präsentation)

- Präsentation: Swing (Java GUI)
- Logik: mit Java JDK unter Verwendung von Eclipse programmiert
- Datenbank: Derby Datenbank mit JDBC Zugriff

Für die Implementierung ist es hilfreich, diese Funktionen auch grafisch genau zu beschreiben. Dafür folgt zunächst ein Klassendiagramm, das eine grobe Struktur der Klassen vorgibt, die zu implementieren sind, sowie deren Funktionen (Methoden), Eigenschaften (Attribute) und wie sie zueinander in Beziehung stehen (Assoziationen). Anschließend werden die einzelnen Funktionen mittels Sequenz-Diagrammen genauer dargestellt. Eine zusätzliche Beschreibung in Textform enthalten die Sequenz-Diagramme nicht, dazu bitte Abschnitt 3.2.1-3.2.7 konsultieren.

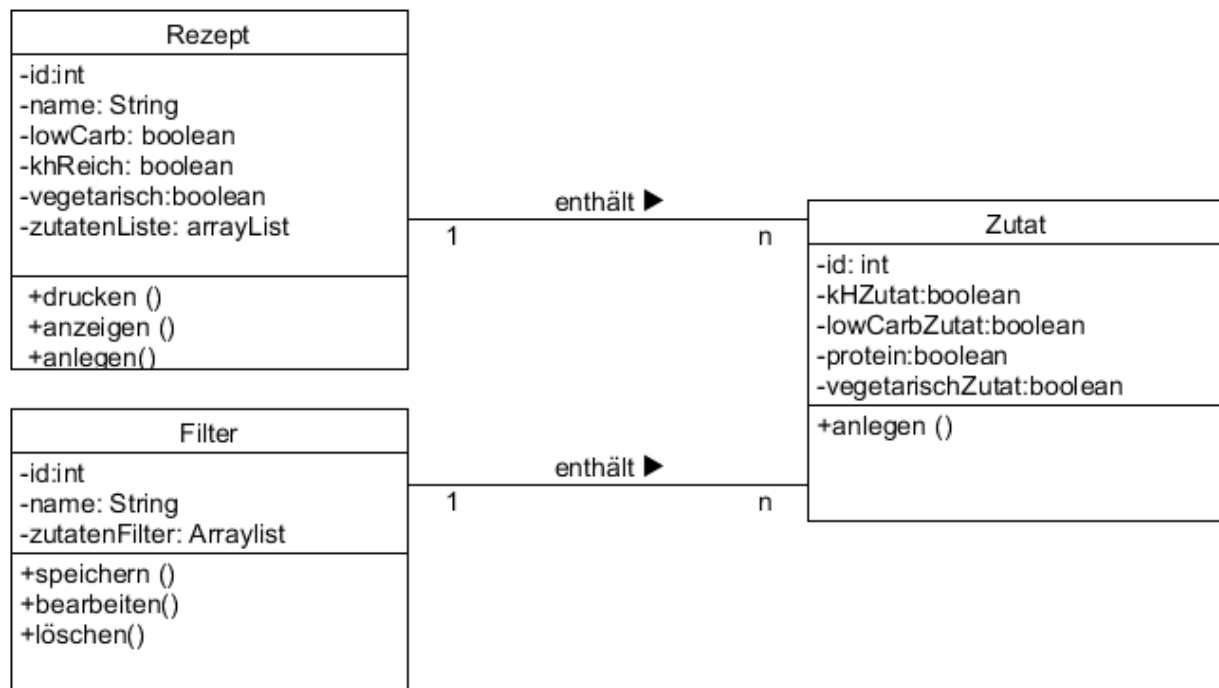


Abbildung 7 Klassendiagramm Rezeptsammlung

Die grafische Benutzeroberfläche (GUI) ist eher einfach strukturiert, wie in den Abbildungen unter 3.3.1 zu sehen ist.

In der Programm-Logik werden die Funktionen implementiert, die im Usecase-Diagramm (Abschnitt 3.2) angeführt sind. Was diese Funktionen jeweils leisten ist in den Tabellen 3.2.1 bis 3.2.7 bereits beschrieben. Aus diesem Grund befinden sich bei den Diagrammen keine zusätzlichen ausführlichen Schilderungen in Textform

Version	Autor	Dateiname	Letzte Änderung	Druckdatum	Seite
1.0		Wifi_Projektdokumentation_Tobolka_2605	26.05.2015	26.05.2015	13 / 19

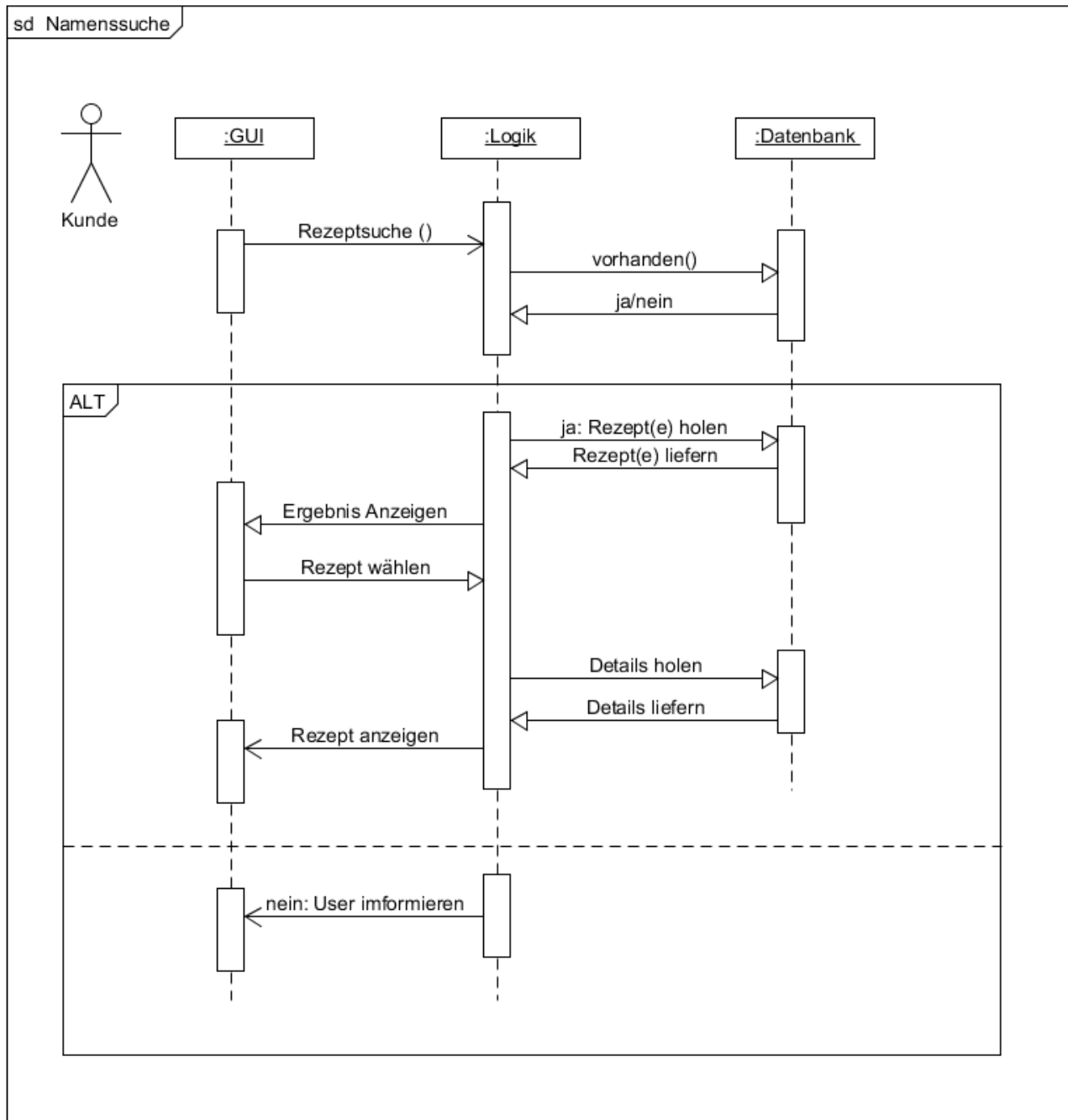


Abbildung 8 Sequenzdiagramm Namens-Suche

Version	Autor	Dateiname	Letzte Änderung	Druckdatum	Seite
1.0		Wifi_Projektdokumentation _Tobolka_2605	26.05.2015	26.05.2015	14 / 19

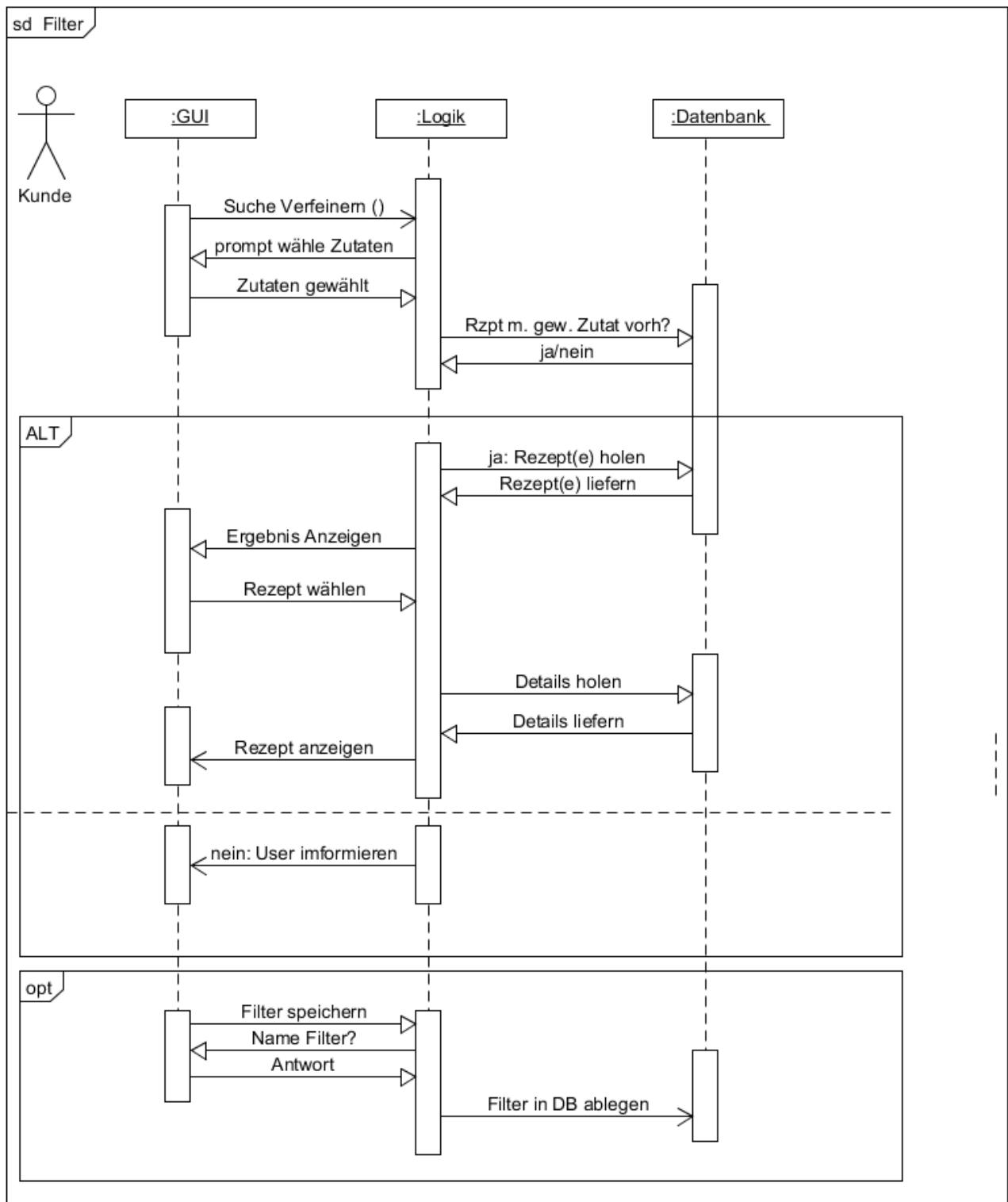


Abbildung 9 Sequenzdiagramm Filter erstellen

Die Sequenzdiagramme zur Funktion Filter bearbeiten und Filter löschen stellen an die Programmlogik keine wesentlichen anderen Anforderungen und sehen vom Ablauf her praktisch ident aus. Aus diesem Grund werden sie nicht gesondert mit eigenen Sequenzdiagrammen dargestellt.

Version	Autor	Dateiname	Letzte Änderung	Druckdatum	Seite
1.0		Wifi_Projektdokumentation _Tobolka_2605	26.05.2015	26.05.2015	15 / 19

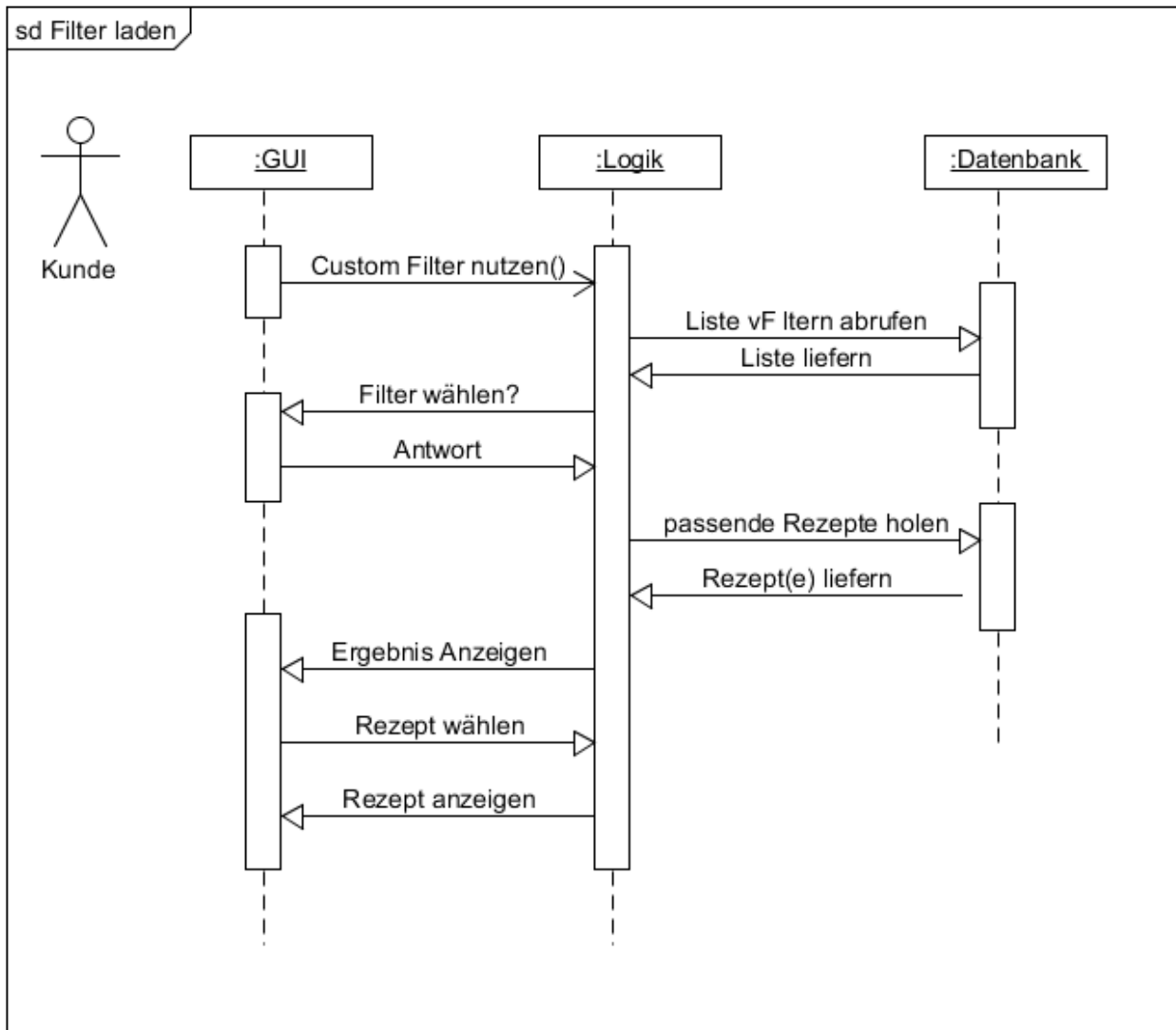


Abbildung 10 Sequenzdiagramm Filter laden

Nachdem Filter ein wesentlicher Bestandteil der zu entwickelnden Software sind wird auf die Funktion „Filter laden“ Wert gelegt. Im Sinne der Usability kann der „Stammgast“ (um im zu Rezepten und Essen passenden Jargon zu bleiben) direkt von der Startseite aus seinen vorher gespeicherten Filter laden. Ohne sich durch allzu viele Informationen wählen zu müssen und ohne sich durch mehrere Seiten „durchklicken“ zu müssen. Das Sequenzdiagramm dazu veranschaulicht, dass der Ablauf auch auf technischer Ebene unkomplizierter und schneller ist. Bei Datenbank-Zugriffen ist das nicht ganz unwesentlich.

Version	Autor	Dateiname	Letzte Änderung	Druckdatum	Seite
1.0		Wifi_Projektdokumentation _Tobolka_2605	26.05.2015	26.05.2015	16 / 19

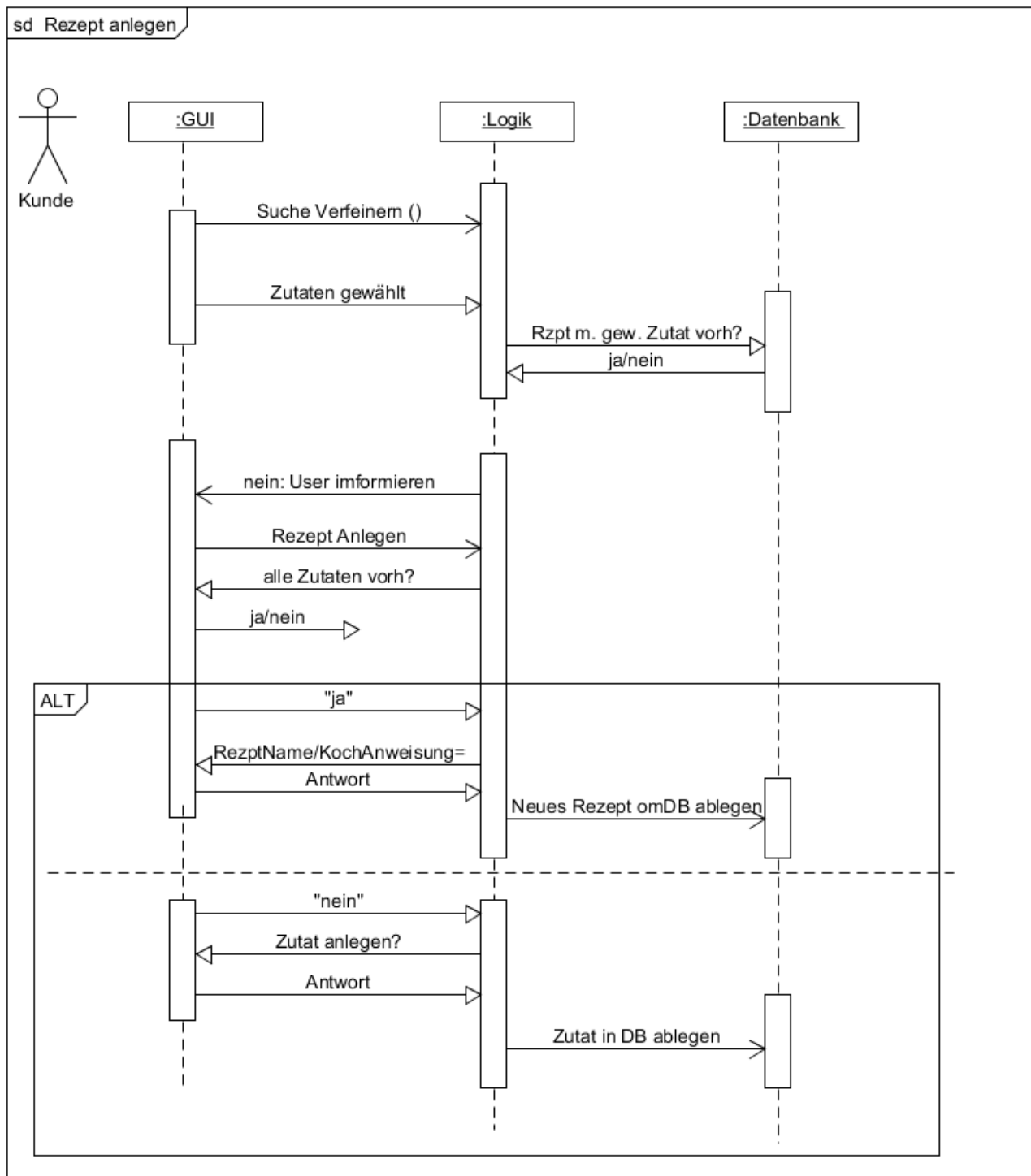


Abbildung 11 Sequenzdiagramm Rezept anlegen

Der Anwendungsfall (Usecase) „neues Rezept anlegen“ ist durch die vorhergehenden Sequenzdiagramme noch nicht abgebildet. Für Architektur und Design dabei interessant ist auch, dass zum Rezept anlegen optional auch der Punkt neue Zutat anlegen gehört.

Version	Autor	Dateiname	Letzte Änderung	Druckdatum	Seite
1.0		Wifi_Projektdokumentation _Tobolka_2605	26.05.2015	26.05.2015	17 / 19

Last but not least wollen wir hier noch etwas ausführlicher auf das Thema Datenhaltung eingehen. Also darauf, wie die Daten der Rezepte und alles, was dazu gehört abgelegt und organisiert sind.

Wie bereits erwähnt benutzen wir eine Derby Datenbank. Der Treiber zu dieser relationalen Datenbank ist in Java enthalten und genügt für unsere Zwecke vollauf. Die Datenbank wird von Java mittels JDBC angesprochen.

Die technischen Anforderungen dafür, ein Rezept in einer Datenbank zu speichern, sind nicht sehr hoch. Das Rezept hat einen Namen, nachdem gesucht werden kann. Es hat bestimmte Zutaten, kann in verschiedene Kategorien eingeordnet werden und dann steht noch etwas Text dabei, der einem sagt, was man mit den Zutaten machen soll. Nachdem man individuelle Filter speichern kann, müssen diese auch in der Datenbank abgebildet werden können. Und da man nach einzelnen Zutaten filtern kann benötigen wir auch eine separate Erfassung jeder einzelnen Zutat.

In der Rezeptsammlungs-Datenbank gibt es 3 Tabellen, je eine für Rezepte, Zutaten und Filter. In allen 3 Tabellen bekommt jeder Eintrag eine ID, in dem Fall eine Zahl, die eindeutig nur diesem Eintrag zugeordnet ist (primary Key). Rezepte, Zutaten und Filter stehen zueinander in Beziehung. Zum Beispiel enthält ein Rezept mehrere Zutaten. Es besteht also unter anderem aus einer RezeptID und mehreren ZutatenIDs. Ähnlich verhält es sich mit Filtern, diese bestehen aus einer Kategorie und oder einer oder mehreren Zutat(en).

Wie Rezepte, Zutaten und Filter, sowie deren Komponenten zueinander in Beziehung stehen wird in dem unten stehenden Entity Relationship Diagramm dargestellt

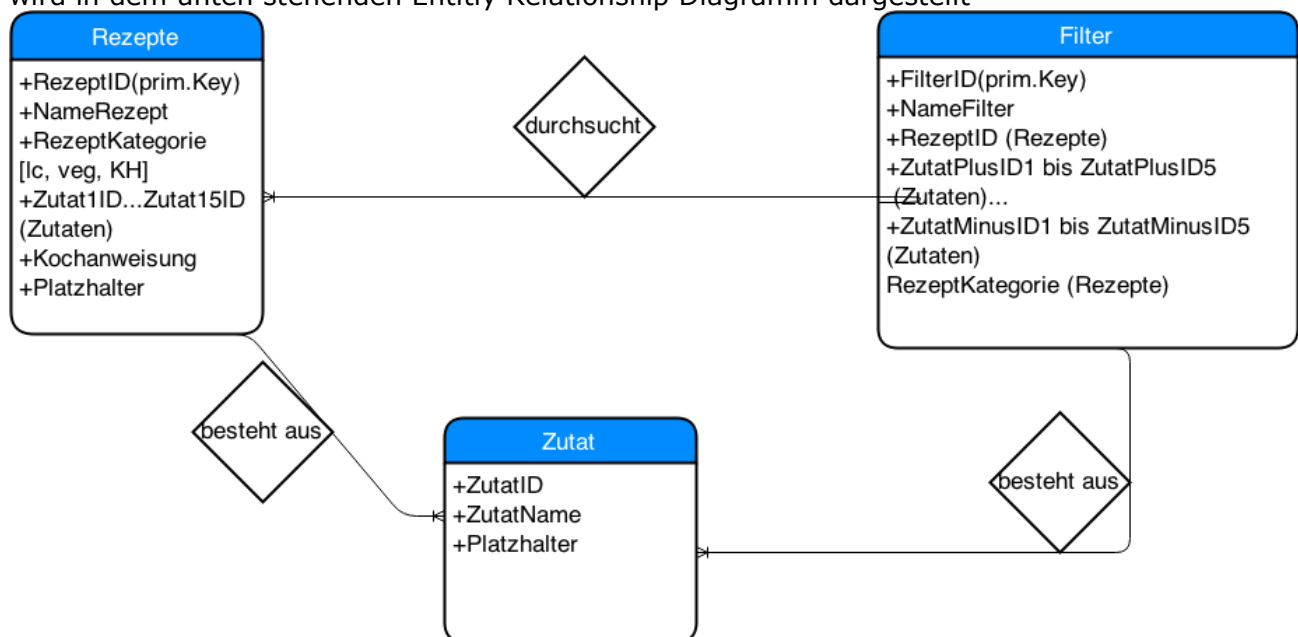


Abbildung 12 Entity Relationship Diagramm der Rezeptsammlungs-Datenbank

Rezepte, Zutaten und Filter werden hier als Entitäten dargestellt. In den Rhomben zwischen zwei Entitäten sieht man die Art der Beziehung in der die Entitäten zueinander stehen. Zusätzlich erkennt man die Kardinalitäten, anhand der Verbindungslinien zwischen den Entitäten. Ein gewöhnlicher Strich signalisiert das hier 1 Exemplar einer Entität in Beziehung steht, der gabelförmige Abschluss der Verbindungslinie bedeutet, hier sind n Exemplare einer Entität mit dem anderen Ende der Verbindungslinie in Beziehung. Zum Beispiel sind in einem Rezept n Zutaten enthalten. Ein Filter durchsucht n Rezepte nach Zutaten.

Zur besseren Übersichtlichkeit sind die Entitäten als Kästchen dargestellt und blau unterlegt. Im unteren Teil des Kästchens sind die Attribute der Entität aufgelistet. Diese entsprechen den

Version	Autor	Dateiname	Letzte Änderung	Druckdatum	Seite
1.0		Wifi_Projektdokumentation _Tobolka_2605	26.05.2015	26.05.2015	18 / 19

Spalten der jeweiligen Tabelle. Die ID an erster Stelle ist der Primary Key, mit dem jeder Eintrag später eindeutig identifiziert und angesprochen werden kann. In einem Rezept befinden sich mehrere Zutaten. Diese werden mittels ZutatID eingetragen, welche ident ist mit der ID einer Zutat in der Zutatentabelle. Wenn in einer Tabelle mit Attributen bzw. dem Primary Key einer anderen Tabelle gearbeitet wird spricht man auch von „Fremdschlüsseln“. Solche sind im Diagramm in Abbildung 12 oben an den runden Klammern erkennbar, die dem Attribut hintangestellt sind und in denen zu lesen ist, aus welcher Tabelle der Fremdschlüssel kommt.

Für ein Rezept können in der Datenbank bis zu 15 Zutaten angegeben werden. Bei Filtern ist es möglich, die Kategorie anzugeben, aus der das Rezept kommen soll, sowie bis zu 5 Zutaten zu inkludieren (ZutatPlusID) und bis zu 5 Zutaten auszuschließen (ZutatMinusID).

3.6 Rahmenbedingungen

<<Offen lassen – wenn sich Punkte, die weiter oben sind, nicht ausgehen: Hier schreiben, dass in späterer Version implementiert werden. >>

< Hier werden Rahmenbedingungen festgelegt die das Produkt / Projekt erfüllen muss oder definitiv nicht erfüllen wird (auch die Nicht-Ziele können angegeben werden um Missverständnisse zu vermeiden).

- *Explizit zu erfüllende Anforderungen des Auftraggebers, z.B. technologische Vorgaben, Kompatibilität (Browser!), Standards, rechtliche Vorgaben, Zeitliche Vorgaben, organisatorische Vorgaben, Vorgaben zur Geheimhaltung, etc.*
- *Wichtige Produktmerkmale zur Absicherung beider Seiten, um Fehlinterpretationen und falsche Hoffnungen zu vermeiden*
- *Dinge die zum Zeitpunkt dieser SRS nicht festgelegt sind und daher später Zusatzaufwand bedeuten können (nicht im Fixpreis / Lieferumfang enthalten, etc.).*
- *Geforderte Meilensteine und Liefertermine wenn direkt mit dem Produkt verbunden / dadurch bedingt (Bankomat-Software Euro Umstellung muss am 1.1. fertig sein!>*

4 Begriffsbestimmungen und Abkürzungen

5 Benutzerhandbuch

Version	Autor	Dateiname	Letzte Änderung	Druckdatum	Seite
1.0		Wifi_Projektdokumentation _Tobolka_2605	26.05.2015	26.05.2015	19 / 19