|  |
| --- |
| A-Team |
| EasyDiet |
| Pflichtenheft |

|  |
| --- |
| Daniel Kuschny;Simon Moosbrugger;Mathias Joschika;Matthias Amann  25.03.2011 |

EasyDiet  
Pflichtenheft (SRS - Software Requirements Specification)   
Version <1.1>

# Revisionsverlauf

|  |  |  |  |
| --- | --- | --- | --- |
| Datum | Version | Beschreibung | Author |
| 18.03.2011 | 1.0 | Einfügen Stakeholder, Einführung-System | Simon Moosbrugger |
| 21.03.2011 | 1.1 | Einführung, Stakeholder und Teil der Produkt Überblick | Mathias Joschika |
| 25.03.2011 | 1.2 | Übernahme von Google-Docs, Neuformattierung | Daniel Kuschny |

Inhaltsverzeichnis

[1 Revisionsverlauf 1](#_Toc288822389)

[2 Einführung 6](#_Toc288822390)

[2.1 System 6](#_Toc288822391)

[2.2 Zweck 6](#_Toc288822392)

[2.3 Umfang 6](#_Toc288822393)

[2.4 Referenzen 7](#_Toc288822394)

[2.5 Abkürzungen 7](#_Toc288822395)

[2.6 Überblick 7](#_Toc288822396)

[3 Stakeholder- und Benutzerbeschreibung 7](#_Toc288822397)

[3.1 Stakeholder/Benutzer Zusammenfassung 8](#_Toc288822398)

[3.2 Benutzerumgebung 9](#_Toc288822399)

[4 Produkt Überblick 10](#_Toc288822400)

[4.1 Zusammenfassung Produktfähigkeiten/Eigenschaften 10](#_Toc288822401)

[4.2 Produkt Fähigkeiten/Eigenschaften 11](#_Toc288822402)

[4.2.1 Systembenutzerverwaltung 11](#_Toc288822403)

[4.2.2 Patientenstamm anlegen 11](#_Toc288822404)

[4.2.3 Diät anlegen 11](#_Toc288822405)

[4.2.4 Diätplan erstellen 11](#_Toc288822406)

[4.2.5 Ernährungsprotokoll anlegen 12](#_Toc288822407)

[4.2.6 Rezepte anlegen 12](#_Toc288822408)

[4.2.7 Patientenstatus anlegen 12](#_Toc288822409)

[4.3 Annahmen und Abhängigkeiten 13](#_Toc288822410)

[5 Domänenmodell 14](#_Toc288822411)

[4.1 Überblick 14](#_Toc288822412)

[4.2 Detailliertes Modell 15](#_Toc288822413)

[5.1.1 Patient 15](#_Toc288822414)

[5.1.2 FamilienAnamnese 15](#_Toc288822415)

[5.1.3 Diätbehandlung 16](#_Toc288822416)

[5.1.4 Behandlungsvorlage 16](#_Toc288822417)

[5.1.5 Behandlungsstatus 16](#_Toc288822418)

[5.1.6 Patientenstatus 17](#_Toc288822419)

[5.1.7 PatientenstatusTyp 18](#_Toc288822420)

[5.1.8 Laborbefund 18](#_Toc288822421)

[5.1.9 Kontaktjournal 18](#_Toc288822422)

[5.1.10 KontaktTyp 18](#_Toc288822423)

[5.1.11 Ernährungsprotokoll 19](#_Toc288822424)

[5.1.12 ErnährungsprotokollTyp 19](#_Toc288822425)

[5.1.13 Plan 19](#_Toc288822426)

[5.1.14 PlanTyp 20](#_Toc288822427)

[5.1.15 Zeitraum 20](#_Toc288822428)

[5.1.16 Mahlzeit 20](#_Toc288822429)

[5.1.17 MahlzeitenZeile 21](#_Toc288822430)

[5.1.18 Rezept / Nahrungsmittel 21](#_Toc288822431)

[5.1.19 Nahrungsmittelparameter 22](#_Toc288822432)

[5.1.20 Parameterdefinition 22](#_Toc288822433)

[5.1.21 Parameterset 22](#_Toc288822434)

[5.1.22 Diätparameter 23](#_Toc288822435)

[5.1.23 DiätparameterTyp 23](#_Toc288822436)

[5.1.24 DiätparameterSet 23](#_Toc288822437)

[5.1.25 Systembenutzer 24](#_Toc288822438)

[5.1.26 Rolle 24](#_Toc288822439)

[5.2 Einschränkungen (Constraints) 25](#_Toc288822440)

[5.2.1 Patient 🡪 Diätbehandlung 25](#_Toc288822441)

[5.2.2 Diätbehandlung 🡪Plan 25](#_Toc288822442)

[5.2.3 Mahlzeit 🡪 Diätparameter / Mahlzeitenzeile 25](#_Toc288822443)

[6 Dynamisches Modell 26](#_Toc288822444)

[5.1 Detaillierte Benutzungsfälle (UseCases) 26](#_Toc288822445)

[6.1.1 <ein detaillierter Benutzungsfall> 26](#_Toc288822446)

[6.1.2 detaillierte Benutzungsfallbeschreibungen 26](#_Toc288822447)

[6.1.3 Sequenz Diagramm 26](#_Toc288822448)

[5.1.1.3 Kontrakte 26](#_Toc288822449)

[6.2 Objekt  Lifecycles 26](#_Toc288822450)

[6.2.1 Lebenszyklus einer Diätbehandlung 26](#_Toc288822451)

[7 Nichtfunktionale Anforderungen 27](#_Toc288822452)

[7.1 Regeln 27](#_Toc288822453)

[7.2 Usability 27](#_Toc288822454)

[7.3 Zuverlässigkeit 27](#_Toc288822455)

[7.4 Performanz 27](#_Toc288822456)

[7.5 Unterstützbarkeit 27](#_Toc288822457)

[7.6 Online Benutzerdokumentation und Help System 27](#_Toc288822458)

[7.7 zugekaufte Komponenten 28](#_Toc288822459)

[7.8 Schnittstellen 28](#_Toc288822460)

[7.8.1 Benutzerschnittstellen 28](#_Toc288822461)

[7.8.2 Software Schnittstellen 28](#_Toc288822462)

[7.8.3 Kommunikationsschnittstellen 28](#_Toc288822463)

[7.9 zusätzliche Lizenzierungen 28](#_Toc288822464)

[7.10 Copyright und andere rechtliche Anforderungen 28](#_Toc288822465)

[7.11 Anzuwendende Standards 28](#_Toc288822466)

[8 Iterationenplan (Timeboxes) 29](#_Toc288822467)

[8.1 Überblick 30](#_Toc288822468)

[8.2 Timebox 1 30](#_Toc288822469)

[8.2.1 Benutzungsfall/fälle (UseCase(s)) 30](#_Toc288822470)

[8.2.2 Architektur 30](#_Toc288822471)

[8.2.3 Deliverables 30](#_Toc288822472)

[8.2.4 Abhängigkeiten 30](#_Toc288822473)

[8.3 Timebox 2 30](#_Toc288822474)

[8.3.1 Benutzungsfall/fälle (UseCase(s)) 30](#_Toc288822475)

[8.3.2 Architektur 30](#_Toc288822476)

[8.3.3 Deliverables 30](#_Toc288822477)

[8.3.4 Abhängigkeiten 30](#_Toc288822478)

[8.4 Timebox 3 30](#_Toc288822479)

[8.4.1 Benutzungsfall/fälle (UseCase(s)) 30](#_Toc288822480)

[8.4.2 Architektur 30](#_Toc288822481)

[8.4.3 Deliverables 30](#_Toc288822482)

[8.4.4 Abhängigkeiten 30](#_Toc288822483)

[9 Glossar 31](#_Toc288822484)

# Einführung

## System

Das System wird von DA’s zur Unterstützung bei der Erstellung von Diätplänen für Patienten im AKH Wien verwendet. Es offeriert nicht nur diese Funktion sondern bietet zusätzlich noch die Möglichkeit EP’s und Laborbefunde zu erfassen, Statistiken anhand dieser zu erzeugen, Rezepte zur Verfügung zu stellen und bietet die Möglichkeit alle Ereignisse die im Zusammenhang mit dem Patienten stehen zu protokollieren.

## Zweck

Der Zweck des Pflichtenheftes/ SRS ist die detaillierte und strukturierte Analyse und Beschreibung der Kundenanforderungen, um die Vollständigkeit des zu entwickelnden Systems sicherzustellen.

Die Zielgruppe ist der Kunde (Definition der Anforderungen), der zuständige technische Projektleiter (Analyse und Dokumentation der Anforderungen), und die Qualitätssicherung (Review)

## Umfang

Dieses Dokument ist nur für das Projekt EasyDiet gültig.

Der Umfang des Projektes umfasst:

* Das Produkt soll als Java-Applikation sowie als Java-Applet angeboten werden.
* Durch die Verwendung von Java ist das Produkt Plattformunabhängig.

Dieses Projekt ist vollkommen unabhängig von anderen Projekten und daher als eigenständiges Projekt zu sehen.

## Referenzen

* Requirements-Workshop
* Systembeschreibung
* Bundeslebensmittelschlüssel

## Abkürzungen

Definition und Beschreibung aller in diesem Dokument verwendeten Abkürzungen.

|  |  |
| --- | --- |
| Abkürzung | Beschreibung |
| DA | Der/die DiätassistentIn |
| BLS | Bundeslebensmittelschlüssel |
| EP | Ernährungsprotkoll |

## Überblick

Das Dokument ist in zwei Teile gegliedert:

* technisches Wissen erforderlich
* technisches Wissen nicht erforderlich

Nur bei Punkt 5 ist technisches Wissen erforderlich und daher nicht für unerfahrene Leser geeignet. In Punkt 5 wird das Domänenmodell besprochen, bei dem Design und Analytisches Wissen vorausgesetzt wird. Bei den anderen Punkten muss kein technisches Wissen vorhanden sein.

# Stakeholder- und Benutzerbeschreibung

Das Erstellen eines Diätplanes birgt viele Möglichkeiten Fehler zu machen. Fehler wie:

* Vorlieben und Abneigungen vergessen oder vertauschen
* Allergien übersehen
* keine Kenntnis über die exakten Inhaltsstoffe eines Rezeptes

Da diese Fehler die Motivation und die Zufriedenheit des Patienten erheblich senken, ist es wichtig, dass diese so gut wie möglich vermieden werden. Durch die Unterstützung des Systems ist die Minimierung dieser Fehler möglich

Ebenso bei der Patientenverwaltung können wichtige Patientendaten vergessen oder vertauscht werden. Wenn dies der Fall ist, muss der Patient extra kontaktiert werden und dies beansprucht extra Zeit und Aufwand.

## Stakeholder/Benutzer Zusammenfassung

| Name | Rolle/Funktion | interessiert an |
| --- | --- | --- |
| Arzt/Ärztin | erstellt die Zuweisungsdiagnose und leitet den Patienten an den Diatässistenten weiter | schnellen Genesung des Patienten,  dass der DA seine Vorgaben umsetzt, |
| DiätassistentIn | Erstellung von Diätplänen, Erhebung des Patientenstatus, entscheidet und überwacht die Diät | dass der Patient die Vorgaben einhält und dadurch den größtmöglichen Nutzen aus Diätbehandlung erhält, an einer guten Unterstützung bei der Diätplanerstellung durch das System, an der Auswertungen der verschiedenen Pläne und Protokolle |
| DA-AssistentIn (SekretärIn, PraktikantIn, StudentIn) | eintragen von EP’s;  Stammdatenpflege; Rezeptverwaltung;  Drucken von Plänen, Vorlagen, etc | einer schnellen Eingabe von EP’s,  der einfachen Stammdatenpflege,  einer unkomplizierten Rezeptverwaltung |
| PatientIn | bekommt eine Diät verschrieben | einer möglichst effizienten Behandlung,  dass auf seine Vorlieben und Abneigungen eingegangen wird, |
| Koch/Köchin | kocht nach den Diätplänen die die DA’s erstellt haben | übersichtliche Darstellung der Rezepte und genaue Auflistung der Zutaten |
| IT-MitarbeiterIn | Wartungsarbeit | einem stablien System,  einer einfachen Wartbarkeit dessen,  einfache Integration in vorhandene Systeme |
| KrankenpflegerIn | eintragen von EP’s | der einfachen Eingabe von Ernährungsprotokollen |
| VersicherungsmitarbeiterIn | Ansehen von Statistiken | einem einfachen Zugang und übersichtlicher Darstellung der Daten |
| Krankenkasse | Bezahlung der anfallenden Behandlungskosten | einer exakten Auflistung der in Anspruch genommenen Leistungen, |
| SoftwareentwicklerIn/TesterIn | Weiterentwicklung des Systemes | der Funktionalität des Programmes und der Weiterentwicklung des Systems |
| Bundesgesundheitsamt | liefert BLS | der Verwendung ihres BLS zur Sicherstellung der Vereinheitlichung |
| Systeminhaber (Stadt Wien) | Finanziert die Hard- und Software | möglichst niedrigen Projektkosten und kurzer -dauer |

## Benutzerumgebung

Durchschnittlich sind bei einem gesamten Durchlauf aller Usecases 2 verschiedene Personen integriert. Als Erstes die Sekretärin, welche die Stammdaten des Patienten in das dazugehörige Formular einträgt.

Als zweites der DA, der die Diätpläne zusammenstellt. Diese 2 sind fixiert und keine anderen Personen bedienen das System. Die Dauer eines solchen Durchlaufes variiert von wenigen Minuten bis hin zu mehreren Wochen, je nachdem welcher Usecase durchgeführt wird. Die Abläufe innerhalb dieses Durchlaufes haben jedoch konstante Zeitaufwände.

Bis jetzt besitzt jeder DA einen PC auf dem Windows XP installiert ist. Jeder PC ist mit dem Netzwerk des AKH Wien verbunden. Bisher wurde zur Erfassung der Daten nur Microsoft Word und Excel verwendet. Es wird darum gebeten, dass auch die zukünftige Applikation auf dieser Architektur laufen muss.

# Produkt Überblick

Das Programm EasyDiet soll DA’s bei ihrer Arbeit unterstützen. Es können die Stammdaten des Patienten eingepflegt werden und es kann im Kontaktjournal mitprotokolliert werden.

Das Hauptfeature des Systems ist das Erstellen von Diätplänen. Hier kann der/die DA für den jeweiligen Patienten aus einer vorgefilterten Liste Rezepte und Nahrungsmittel für die verschiedenen Mahlzeiten auswählen.

Im Nachhinein können diverse Auswertungen gemacht werden. Grundlage für diese Auswertungen sind die EP’s, die Diätparameter des Diätplanes und die Laborbefunde.

Im Krankenhaus befindet sich bereits die benötigte Hardware, da die DA’s bisher den Computer aber nur zum Nachsehen von Patientendaten verwendet haben, soll nun ein unterstützendes Programm eingeführt werden.

## Zusammenfassung Produktfähigkeiten/Eigenschaften

|  |  |
| --- | --- |
| Produktfähigkeit/-eigenschaft | Stakeholder Nutzen/Gewinn |
| Diätplan erstellen | DA: Die Rezepte können nach Abneigungen und Allergien des Patienten gefiltert werden. Außerdem kann der DA Zielparameter hinzufügen, welche er später bei der Soll-Ist Analyse vergleichen kann. Durch das Hinzufügen der verschiedenen Rezepte wird automatisch berechnet, ob sich die aktuellen Werte noch im Bereich der angegebenen Zielparameter oder diese bereits überschritten haben. |
| Soll - Ist Vergleich | DA: Durch den Soll - Ist Vergleich kann der DA die Entwicklung des Patienten anhand der Zielparameter verfolgen. Falls die Werte zu stark abweichen, kann der DA reagieren und Maßnahmen ergreifen. Ebenso kann kontrolliert werden ob der DA bis jetzt die richtige Behandlungsform gewählt hat. |
| Patientenverwaltung | DA: Durch die Patientenverwaltung können Stammdaten sowie medizinische Daten eines Patienten zusammen zentral abgelegt werden und dort abgefragt werden. Der Patient kann eindeutig identifiziert werden. Der Arzt sowie der DA haben Zugang zu den Daten und es müssen nicht Daten mehrfach eingegeben werden und gar redundant gespeichert werden. |
| Rezeptverwaltung | DA: Durch die Rezeptverwaltung kann der DA die verschiedenen Rezepte immer wieder verwenden, ohne sie jedes Mal anlegen zu müssen. Nicht nur die Wiederverwendbarkeit, sondern auch das einfache hinzufügen stellt sicher, dass die Rezeptdatenbank immer auf dem akutellsten Stand ist. |
| Patientenstatus | DA: Durch den Patientenstatus kann der DA genau feststellen wie die momentane psychische sowie physische Verfassung des Patienten ist. Dank dieser Information kann der Da nicht nur die Diätpläne genau auf die Bedürfnisse des Patienten anpassen sondern weiß auch wie er dem Patienten am besten umgehen soll. |
| Diätplanvorlage anlegen | DA: Da Diätpläne oft ähnlich sind, kann der DA Diätplanvorlagen erstellen. Wenn nun ein neuer Diätplan erstellt wird, muss er nur noch die sich ändernden Daten eingeben und erspart sich somit Zeit und Aufwand. |

## Produkt Fähigkeiten/Eigenschaften

### Systembenutzerverwaltung

Um einen neuen Benutzer anzulegen muss ein Loginname, -passwort, Vorname, Nachname, E-Mail Adresse, Durchwahl, Abteilung, Position eingegeben werden. Außerdem kann der Administrator noch Felder für beliebige Daten anlegen. Ebenso können Systembenutzer bearbeitet, gelöscht und angezeigt werden.

### Patientenstamm anlegen

Das System EasyDiet unterstützt eine Patientenstammdatenverwaltung. Die Stammdaten beinhalten allgemeine Informationen wie Name, Adresse, Geburtsdatum, Geschlecht sowie Informationen über den Familienstatus, Religion, Sozialanamnese sowie Familienanamnese. Ebenso werden die medizinischen Daten des Patienten erfasst. Zu diesen gehören frühere Krankheiten und Operationen des Patienten. Zugleich können eventuelle Ausschlusskriterien für Nahrungsmittel/Inhaltsstoffe definiert werden, die später bei der Diätplanerstellung vom System berücksichtig werden.

### Diät anlegen

Für jeden Patienten wird zuerst eine Diät angelegt, welche dann alle Diätpläne beinhaltet. Die Diät enthält den Zeitraum, eine Kurzbeschreibung sowie einen optionalen Beschreibungstext bei dem zusätzliche Informationen eingegeben werden können. An die Diät werden dann die Zuweisungsdiagnose und das Ernährungsprotokoll vom Typ 24-Stundenprotokoll angehängt. Als letztes werden noch die Parameter definiert, die mit dieser Diät überwacht bzw. erreicht werden möchten.

### Diätplan erstellen

In Diätplanen herrscht folgende Struktur:

• Diätpläne beinhalten Zeiträume (mit Parametern)

• Zeiträume beinhalten Mahlzeiten (mit Parametern)

• Mahlzeiten beinhalten Rezepte und/oder Nahrungsmittel

Der Diätassistent wählt die Rezepte nach seiner Erfahrung und Ausbildung aus, wobei die Ausschlusskriterien, die der Patient beim Beratungsgespräch angegeben hat, bereits gefiltert werden. Es gibt 3 verschiedene Typen von Diätplänen: Ernährungsempfehlung, Diätplan, Speiseplan. Wenn der Plan eine Ernährungsempfehlung ist, sind die Parameter bei der Mahlzeit optional sonst nicht.

### Ernährungsprotokoll anlegen

Ernährungsprotokolle enthalten dieselben Informationen wie ein Diätplan. Es können Zeiträume mit Mahlzeiten hinzugefügt werden. In den Mahlzeiten könne Rezepte und Nahrungsmittel angehängt werden. Der eigentliche Unterschied zum Diätplan ist, dass hier die wirklich verzehrte Menge angegeben wird und diese als Grundlage für die Auswertungen dient.

### Rezepte anlegen

Beim Anlegen eines neuen Rezeptes müssen folgende Informationen eingegeben werden:

* Name (Pflichtfeld)
* Schwierigkeitsgrad (Mit 1-5 Kochlöffeln)
* Zubereitungszeit (Pflichtfeld)
* Nutzungsbeschreibung (Pflichtfeld) -„Was hat dieses Rezept für eine Auswirkung“
* Kurzbeschreibung (optional)

Anschließend müssen die Parameter des Rezeptes festgestellt und eingetragen werden. Des Weiteren werden die Nahrungsmittel aus denen das Rezept besteht hinzugefügt bis das Rezept vollständig ist.

### Patientenstatus anlegen

Der Patientenstatus besteht aus folgenden Informationen:

* Datum  
  *Zeitpunkt der Aufnahme*
* Anamnese
* Gewicht
* Größe
* Perzentilen
* Compliance
* Motivation

Zusätzlich zu den oben angeführten Informationen kann ein oder mehrere Laborbefunde angehängt werden. Die Anamnese beinhaltet die momentane Einschätzung des DA’s über den aktuellen Zustand des Patient.

Parameterset anlegen:

Ein Parameterset dient zur Gruppierung von Parametern. Beim Erstellen eines Parametersets wird zuerst der Namen des Sets definiert und anschließend die gewünschten Parameter mit den entsprechenden Kriterien hinzugefügt.

Diätplanvorlage erstellen:

Es können auch Diätplanvorlagen erstellt werden. Der Ablauf ist genau derselbe wie beim Diätplan erstellen, jedoch wird die Vorlage keiner Diät zugeordnet, sondern in einem Vorlagenpool abgelegt.

Diätvorlage erstellen

Das Erstellen einer Diätvorlage ist identisch mit dem Erstellen einer Diät. Der Hintergedanke hierbei ist, dass es standardisierte Diäten gibt, die für alle Menschen gleich sind.

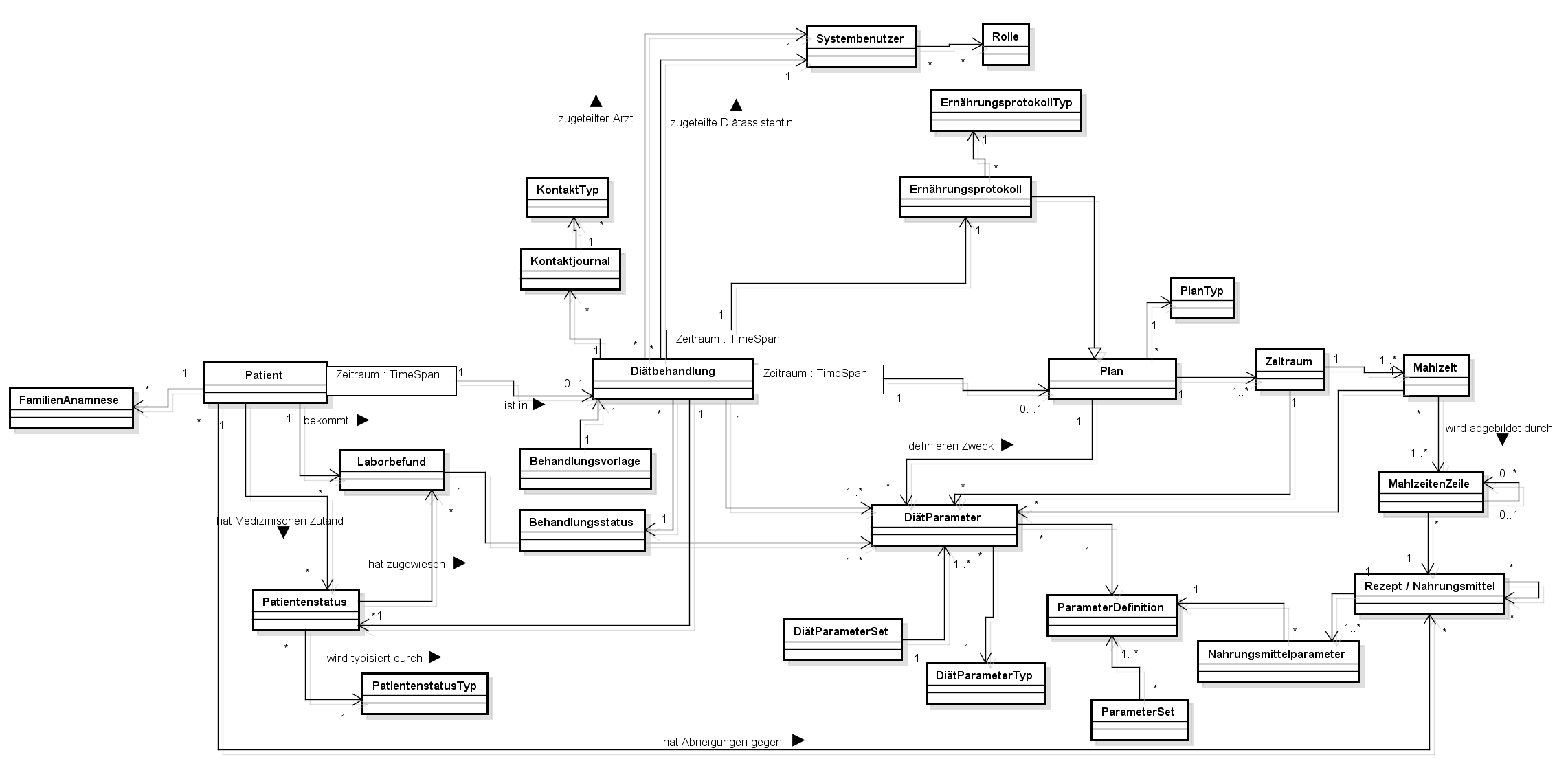
## Annahmen und Abhängigkeiten

[TODO]

# Domänenmodell

Das Domänenmodell gibt einen Überblick über die im System vorhandenen Elemente und deren Beziehung miteinander.

## Überblick



## Detailliertes Modell

In den folgenden Sektionen werden die einzelnen Entitätstypen des Domänenmodells genauer erläutert und deren Aufbau beschrieben.

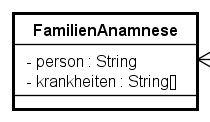
### Patient



Ein *Patient* ist die Repräsentation der Stammdaten eines Patienten. Diese Stammdaten umfassen neben Sozialversicherungsnummer, Name und Adresse auch die Sozialanamnese, Familienanamnese sowie die bisherige Krankengeschichte. Über eine Verknüpfung mit hinterlegten Nahrungsmitteln, können Abneigungen oder Medizinische Kriterien zu Nahrungsmitteln hinterlegt werden. Beispiel:

* Patient mag kein Käse
* Patient ist Allergisch gegen Nüsse

### FamilienAnamnese



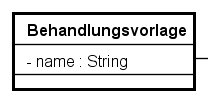
Eine *Familienanamnese* beinhaltet die Krankheiten für eine einzelne Person im Umfeld des Patienten. Die Krankheiten werden als Liste von beliebigen Texten abgelegt.

### Diätbehandlung



Eine *Diätbehandlung* gruppiert eine Reihe von Diätplänen mit dem Ziel einen über Diätparameter definierten Nutzen über einen definierten Zeitraum zu erreichen. Diese Behandlung kann mit der Umgangssprachlichen „Diät“ verglichen werden. Zu dieser Behandlung werden Protokolle unterschiedlicher Art erfasst. Ein Patient hat zu einem Zeitpunkt/-raum immer nur eine gültige Diätbehandlung. Es können nicht zwei oder mehr Diätbehandlungen zeitgleich verfügbar sein.

### Behandlungsvorlage



Oft benötigte Diätbehandlungen wie etwa eine „5 Wochen Diät“ können als *Vorlagen* hinterlegt werden. Diese Vorlagen können bei Bedarf als Basis für neue Diätbehandlungen dienen. Dieses Vorlagensystem ermöglicht es zusätzlich Vorlagen für einzelne Diätpläne zu hinterlegen.

### Behandlungsstatus



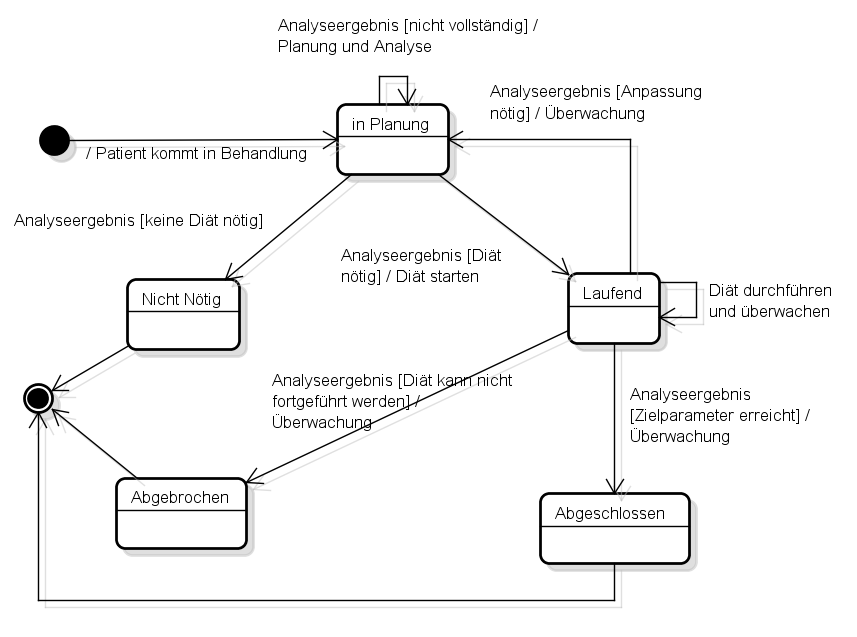
Der Behandlungsstatus gibt an in welcher Phase sich die Behandlung zurzeit befindet.

* In Planung  
  *Die Behandlung befindet sich im Planungsstadium. Es wurden noch keine Pläne zugeteilt. Der Patient ist evtl. mit dem ersten 24-Recall Protokoll beschäftigt.*
* Nicht Nötig

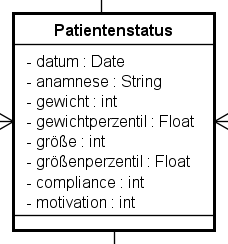
*Die Planungsphase hat ergeben dass der Patient keine Diät benötigt, die Diät wird nicht fortgeführt.*

* Laufend  
  *Die Diät läuft gerade. Der Patient ist mit der Umsetzung der Pläne und Erfassung von EPs beschäftigt.*
* Abgeschlossen  
  *Die Diät ist zeitlich abgeschlossen und wurde bis zum Ende durchgeführt.*
* Abgebrochen   
  *Die Diät wurde während der Durchführung abgebrochen.*

Das nachfolgende Diagramm beschreibt die Übergänge zwischen diesen einzelnen Status:

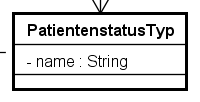


### Patientenstatus



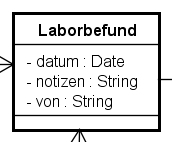
Im *Patientenstatus* wird der gesundheitliche und psychische Zustand des Patienten für einen Zeitpunkt als Fließtext erfasst. Zum Patientenstatus können Laborbefunde angehängt werden welche für diesen Status relevant sind. Der Patientenstatus wird über einen *PatientenstatusTyp* genauer typisiert.

### PatientenstatusTyp



Der *PatientenstatusTyp* definiert für einen Patientenstatus welche Art von Status erfasst wird. Dies kann neben einer *Zuweisungsdiagnose* auch eine *laufende Erfassung* sein.

### Laborbefund



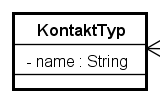
In einem Laborbefund werden die medizinischen Eigenschaften wie Blutwerte oder Körperfettanteil des Patienten erfasst. Der Laborbefund selbst beinhaltet dabei lediglich den *Kopf*  des Befundes. Die Werte zum Befund werden über eine Verknüpfung auf *Diätparameter* angelegt da die Struktur der Diätparameter die Erfassung von Werten für *Parameterdefinitionen* erlaubt.

### Kontaktjournal



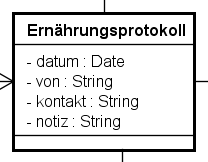
Im Kontaktjournal wird erfasst wann in welcher Form mit dem Patienten Kontakt aufgenommen wurde und auch warum. Die Kontakte werden zu einer Diätbehandlung erfasst.

### KontaktTyp



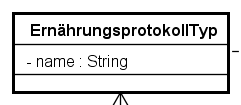
Der KontaktTyp legt für ein Kontaktjournal fest, um welche Art von Kontakt es sich gehandelt hat: Untersuchung, Telefonische Absprache, Diätplanüberarbeitung

### Ernährungsprotokoll



In einem *Ernährungsprotokoll* ist eine Repräsentation eines *Plans* welche dazu verwendet wird, verzehrte Mengen von Nahrungsmitteln zu erfassen. Über dieselbe Struktur eines normalen Plans kann über die Hierarchie von *Zeiträumen* und *Mahlzeiten* genau erfasst werden von welchem Nahrungsmittel/Rezept wie viel verzehrt wurde.

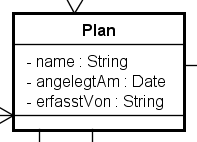
### ErnährungsprotokollTyp



Ein *ErnährungsprotokollTyp* definiert für ein Ernährungsprotokoll um welche Art von EP es sich handelt: Normales Ernährungsprotokoll, Wäge-Protokoll,…

Dieser Typ legt den Bindungsgrad fest, in welchem Detailgrad Protokolle erfasst werden müssen.

### Plan



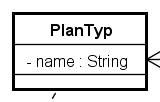
Ein *Plan* wird dazu verwendet um über eine Liste von Zeiträumen einen Diätplan, eine Ernährungsempfehlung oder ähnliches abzubilden. Über Diätparameter können Kriterien für diesen Plan festgelegt werden welche einzuhalten oder zu überwachen sind. Der zeitliche Gültigkeitsbereich des Plans wird über sogenannte *Zeiträume* festgelegt.

Pläne werden in Folge über diese Hierarchie befüllt:

* Pläne beinhalten Zeiträume
  + Zeiträume beinhalten Mahlzeiten
    - Mahlzeiten beinhalten Mahlzeitenzeilen
      * Mahlzeitenzeilen haben Nahrungsmittel/Rezepte zugeordnet

Um welche Art Plan es sich handelt, wird über einen Zugordneten Typ festgelegt.

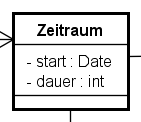
### PlanTyp



Der *PlanTyp* legt fest um welche Art von Plan es sich handelt. Je nach *PlanTyp* hat dies Auswirkung auf die Validierung und den Bindungsgrad des Planes:

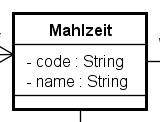
* Ernährungsempfehlung  
  *Der Patient hat sich lediglich an Diätparameter zu halten und bekommt keine genauen Nahrungsmittel vorgeschrieben*
* Diätplan  
  *Beim Diätplan kann der Patient sich zwischen den vorgegebenen Nahrungsmitteln und Rezepten aussuchen welche er zu sich nehmen möchte.*
* Speiseplan  
  *Der Speiseplan gibt dem Patienten exakt vor welche Nahrungsmittel er in welcher Menge verzehren muss. Der Patient hat sich daran zu halten.*

### Zeitraum



Ein *Zeitraum* bildet die erste Hierarchieebene innerhalb eines *Plans*. Ein Zeitraum definiert einen Startzeitpunkt und eine Dauer in Tagen für welche der Zeitraum gültig ist. Zeiträume dürfen sich nicht überschneiden (auch nicht über Pläne hinweg). Zu jedem Zeitpunkt darf nur maximal ein Zeitraum gültig sein. Für Zeiträume können, wie bei Plänen, optional Diätparameter festgelegt werden welche für diesen Zeitraum gültig sein müssen.

### Mahlzeit



Eine *Mahlzeit* ist eine für einen Zeitraum spezifische Ansammlung von Rezepten welche über *Mahlzeitenzeilen* hinzugefügt werden. Eine Mahlzeit hat neben einem Namen und Identifikationscode ebenfalls die Möglichkeit über Diätparameter den Nutzen der Mahlzeit genauer eingrenzen zu können.

### MahlzeitenZeile



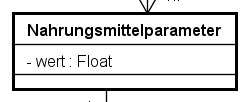
Eine *MahlzeitenZeile* ist ein einzelner Eintrag einer Mahlzeit in welcher für ein Rezept zusätzliche Angaben gemacht werden können. So kann neben der Verzehrmenge und einem Zusatztext auch festgelegt werden ob ein Rezept/Nahrungsmittel hinzugefügt oder entfernt werden soll. MahlzeitenZeilen können hierarchisch angeordnet sein was es erlaubt Varianten von Nahrungsmitteln zu erstellen:  
+ 1 Tasse Kaffee   
            - Zucker  
            + Süßstoff  
+ 1 Teller Gemischter Salat   
            - Rote Bete

### Rezept / Nahrungsmittel



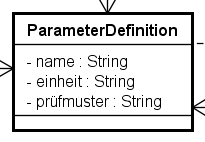
Ein *Rezept* beschreibt ein zu verzehrendes Produkt welches aus beliebig vielen weiteren Rezepten oder Nahrungsmitteln zubereitet und verzehrt werden kann. Rezepte und Nahrungsmittel werden größtenteils vom Bundeslebensmittelschlüssel vorgegeben und haben somit einen eindeutigen Code genannt BLS-Code zugeordnet. Der Schwierigkeitsgrad wird als 1-5 angegeben wobei 1 Leicht und 5 Schwierig ist. Der Nutzen beschreibt den Medizinischen Nutzen des Nahrungsmittels/Rezeptes als Fließtext während die Beschreibung eine inhaltliche Beschreibung des Rezeptes ist.

### Nahrungsmittelparameter



Ein *Nahrungsmittelparameter* beschreibt die Zuordnung einer Parameterdefinition inklusive Wert zu einem Rezept. Somit kann bspw. einem Nahrungsmittel Ei über einen Nahrungsmittelparameter ein Brennwert von 93 kcal zugewiesen werden.

### Parameterdefinition

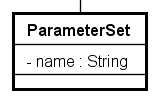


Eine Parameterdefinition beschreibt Nährwerte, Inhaltsstoffe,… deren Einheit und gültiger Wertebereich. Beispiele:

|  |  |  |
| --- | --- | --- |
| Name | Einheit | Gültige Werte |
| Brennwert | kcal | Zahlen |
| Vitamin B | g | Zahlen |
| Laktose |  | Frei, Haltig |

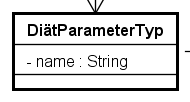
Die gültigen Werte werden über als Prüfmuster angegeben.

### Parameterset



Über ein *ParameterSet* können Parameterdefinitionen für die Wiederverwendung gruppiert werden. Während des Anlegens eines Rezeptes oder eines Diätparametersets können auf diese Parametersets zurückgegriffen werden um schnell alle nötigen Parameterdefinitionen hinzuzufügen.

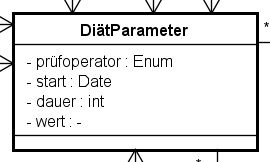
### DiätparameterTyp



Ein Diätparametertyp spezifiziert für einen Diätparameter welche Art von Diätparameter ist. Beispiele:

* Überwachungsparameter
* Zielparameter

### Diätparameter

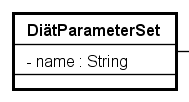


Ein *Diätparameter* definiert eine Prüfbedingung für eine Parameterdefinition. Ein Diätparameter kann somit festlegen welche Werte für welche Parameterdefinition in welchem Zeitraum einzuhalten sind. Beispiel:

* Von 01.01.2011 bis 05.01.2011 muss der Brennwert kleiner als 5000kcal sein.

Diese Parameter können in unterschiedlichen Bereichen des Systems verwendet werden um bspw. die Grenzwerte für Überwachungsparameter oder Zielparameter festzulegen. Die Datumsangabe solcher Parameter ist optional.

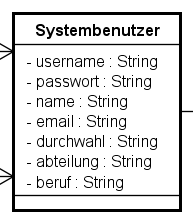
### DiätparameterSet



Ein Diätparameterset gruppiert eine Reihe von verwendbaren Diätparametern. So können Gruppen für Diätparameter angelegt werden die in Kombination oft benötigt werden. Die Angabe der Werte für Diätparameter in Diätparametersets ist optional. Beim Zuweisen eines *DiätParameterSets* werden alle Diätparameter dupliziert sodass Werte angepasst werden können.  
Beispiel:  
**Fettrelevante Parameter**

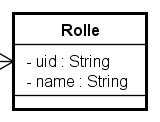
|  |  |  |
| --- | --- | --- |
| Parameterdefinition | Prüfoperator | Wert |
| Brennwert (kcal) | > |  |
| Brennwert (kcal) | < | 1500 |
| Fettgehalt (g) | < | 50 |

### Systembenutzer



Ein *Systembenutzer* hat die Möglichkeit sich bei EasyDiet mit seinen Anmeldedaten einzuloggen und Aktionen im System zu tätigen. Das Passwort wird im System verschlüsselt abgelegt.

### Rolle



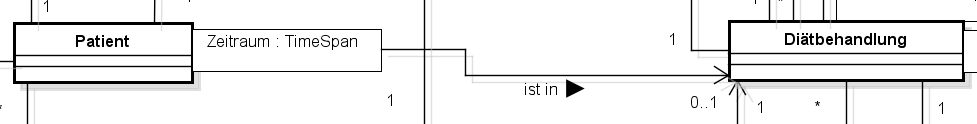
Eine Rolle räumt einem Systembenutzer Rechte innerhalb von EasyDiet ein. Diese Rollen werden auf Anwendungsfallebene abgebildet

Beispiele

|  |  |
| --- | --- |
| UID-Rolle | Beschreibung |
| CREATE\_PATIENT – Patientenstammdaten anlegen | Der Benutzer hat die Berechtigungen neue Patienten im System anzulegen |
| ADD\_RECIPE – Rezept anlegen | Der Benutzer hat die Berechtigungen neue Rezepte im System anzulegen. |

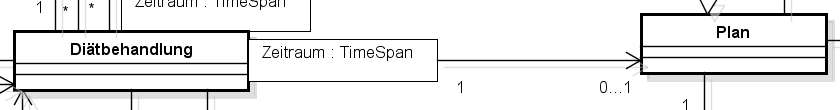
## Einschränkungen (Constraints)

### Patient 🡪 Diätbehandlung



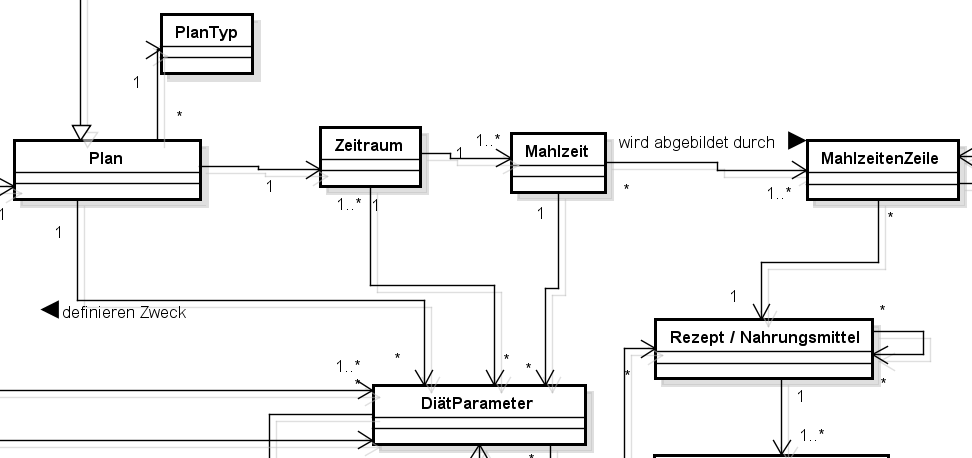
Der Patient hat innerhalb eines beliebigen Zeitraumes maximal eine Diätbehandlung zugeordnet. Diätbehandlungen dürfen Sich nicht zeitlich überschneiden.

### Diätbehandlung 🡪Plan



Innerhalb einer Diätbehandlung darf für jeden Zeitraum maximal ein Plan gültig sein.

### Mahlzeit 🡪 Diätparameter / Mahlzeitenzeile



Ein Spezialfall von Mahlzeiten tritt ein wenn die Mahlzeit innerhalb eines Plans vom Typ Ernährungsempfehlung angelegt wird. Ist dies der Fall, können zur Mahlzeit keine Mahlzeitenzeilen hinzugefügt werden. Stattdessen müssen nun zur Mahlzeit Diätparameter definiert werden. Im Spezialfall der Ernährungsempfehlung werden somit keine Nahrungsmittel vorgeschrieben, sondern es werden lediglich Kriterien für diese Mahlzeit festgelegt.

Ein weiterer Spezialfall hierbei ist: Wurden bereits im Zeitraum der Mahlzeit Diätparameter festgelegt ist die Angabe weiterer Diätparameter für die Mahlzeit keine Pflicht.

# Dynamisches Modell

## Detaillierte Benutzungsfälle (UseCases)

### <ein detaillierter Benutzungsfall>

### detaillierte Benutzungsfallbeschreibungen

### Sequenz Diagramm

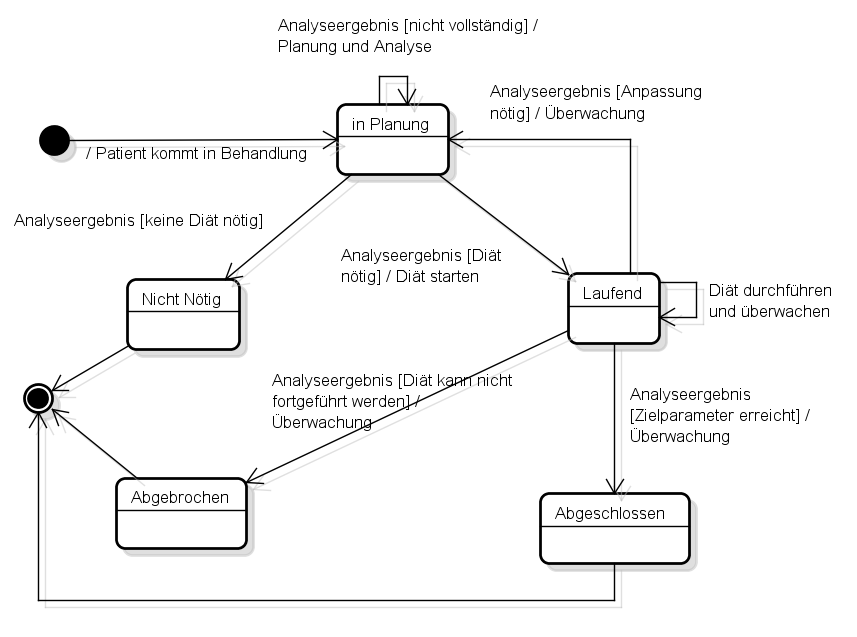
[optional für solche Benutzungsfälle, deren Schritte mit Kontrakten beschrieben werden]

### Kontrakte

[für den UseCase als Ganzes, und für einzelne Arbeitsschritte darin. Ist optional, für Usecases bei denen wichtige, oder nicht selbstverständliche Voraussetzungen oder Nachbedingungen auf dem Domänenmodell zu definieren sind]

## Objekt Lifecycles

### Lebenszyklus einer Diätbehandlung



# Nichtfunktionale Anforderungen

[nach den FURPS+ Kategorien, siehe Vorlesungsfolien Kapitel 2: Anforderungestypen. Nur das auswählen, was wir für den Roomanizer wirklich als Anforderung wissen]

## Regeln

Patienten werden nur behandelt wenn ihre Identität festgestellt wurde oder ein akuter Notfall ein umgehen dieser Regel erfordert.

Nur ein ausgebildeter Diätassistent darf Diätpläne verwalten, da es sonst zu rechtlichen Problemen für das Krankenhaus kommen kann.

## Usability

Das Programm wird eine intuitive Oberfläche besitzen, das heißt man muss sich nicht zuerst mit der Dokumentation befassen sondern kann gleich mit der Arbeit beginnen.

Das Programm unterstützt den User wo immer es geht mit Hilfe von Tooltips, Erklärungen, Validierungsfunktionen und Hinweisen.

Falsche oder nicht eingegeben Daten in einem Formular werden, sofern überprüfbar, auch als Falsch markiert und dem Nutzer wird eine Meldung dazu ausgegeben.

## Zuverlässigkeit

Benutzereingaben werden vom Benutzer manuell gespeichert.

Stürzt das Programm ab, kann mit den zuletzt gespeicherten Daten weitergearbeitet werden.

Grundsätzlich sollte das System stabil und ohne Abstürze funktionieren.

## Performanz

Das Programm soll dem/der DA die Möglichkeit bieten Diätpläne schneller zu erstellen.

Es werden dadurch mehr Patienten an einem Tag behandelt werden können.

## Unterstützbarkeit

EasyDiet ist für das AKH Wien ausgelegt, das heißt es wird nur in Deutsch angeboten.

Die Einträge der Tabellen in der Datenbank werden vom Benutzer mit Hilfe des Programms selbst administriert.

## Online Benutzerdokumentation und Help System

[TODO(machen wir online dokumentation? oder ist das das hier?)]

## zugekaufte Komponenten

Für die Entwicklung ist der BLS des Bundeministeriums für Ernährung, Landwirtschaft und Verbraucherschutz gratis.

Für die Verwendung im AKH muss dieser zugekauft werden. Hierfür bietet das Bundesministerium verschiedene Lizenzmodelle an. Nachzusehen sind diese auf der Homepage des Bundesministeriums (http://www.bls.nvs2.de/index.php?id=134).

## Schnittstellen

### Benutzerschnittstellen

[TODO:Wird hier die GUI gemeint?         ]

### Software Schnittstellen

Die Software bietet Schnittstellen zur Krankenhaussoftware. Diese ermöglicht den Austausch von Patientendaten, damit nicht jeder Patient neu im EasyDiet-System angelegt werden muss.

### Kommunikationsschnittstellen

[TODO: Was ist hier gemeint?]

## zusätzliche Lizenzierungen

## Copyright und andere rechtliche Anforderungen

Das Copyright der Software teilen sich die Entwickler und die Stadt Wien die als Systeminhaber das Projekt auch finanziert.

Die Software bleibt geistiges Eigentum der Entwickler wobei der Stadt Wien jegliche Rechte zum Vertrieb und Einsatz der Software zugestanden werden.

Das Krankenhaus muss selbst sicherstellen, dass nur berechtigte Personen die Zugangsdaten zum System erhalten um somit den gesetzlichen Bestimmungen Folge zu leisten.

Die Entwickler distanzieren sich von jeglicher Verantwortung die im Zusammenhang mit der Verwendung des Systems.

## Anzuwendende Standards

[TODO: Was für Standards verwenden wir?]

# Iterationenplan (Timeboxes)

TODO

[eine Beschreibung der 3 nächsten Realsierungsphasen (Iterationen) in Form von Timeboxen. Verlangt eine vorherige Berwertung von UseCases – das Ranking.

Ranking:

UseCases werden geordnet nach ihrer Bewertung. Die drei höchstbewerteten werden in den drei ersten Timeboxes realisiert. Die Berwertung geschieht nach folgenden Kriterien:

1. Risiko
   * Komplexität des UseCses
   * Anforderungen (inkl. an die Usability): haben/bekommen wir die richtigen.Anforderungen?  wie stabil sind sie? haben wir Ansprechpartner?
   * Team KnowHow: der dazu notwendigen Technologie, und des Domänenverständnisses
   * Technologie: ist die einzusetzendeTechnologie gut definiert, stabil, erprobt, einfach?
   * politisch
2. Architekturrelevanz gibt die Realisierung des UseCases wichtige Einblicke in die SW-Architektur des Systems? Hilft er bei den wichtigsten Entscheidungen zur Architektur?
3. Benutzerrelevanz ist der Benutzer an einer frühen Realiserung dieses UseCases interessiert?
4. Lern-und Entwicklungsrelevanz ist der UseCase geeignet, um mangelndes teaminternes KnowHow bzgl. GUI-Entwurf, Klassenentwurf, Datenbankanbindung etc. aufzubauen?

Tip: Ranking zunächst nach 1-3 , dann 4 miteinbeziehen. Nur 1-3 in diesem Dokument erwähnen.

Da in unserem Zeitplan die Zeiten der drei ersten Timeboxes feststehen, muss für jede Timebox der zu realsierende UseCase (Main Success Scenario/Extensions) und die Architektur festgelegt werden in der er realsiert wird.

die Architektur betrifft Einschränkungen bezüglich:

* GUI
* Domänenmodell
* DB-Anbindung
* Reports, Ausdrucke, Networking, Security, HW-Plattform
* Fehlerbehandlung

und muss eventuell berücksichtigen, dass gewisse Systemteile, die der UseCase voraussetzt, bisher noch fehlen und in irgendeiner Weise zunächst bereitgestellt werden müssen, bis spätere Iterationen diese teile liefern werden.]

## Überblick

[Reihenfolge der Timeboxen, welche UseCses, in welchen Zeiten, mit welchen Abhängigkeiten]

## Timebox 1

[eine Iterationsbeschreibung, in der ein Systemteil realisiert wird, in einer fest vorgegebenen Zeit]

### Benutzungsfall/fälle (UseCase(s))

[der zu realsierende UseCase, oder mehrere. Und der Umfang davon: Main Success Scenario, wenn das in früheren Timeboxes noch nicht realisert wurde, dann welche Extensions?]

### Architektur

[siehe oben, Info zu 7. Immer aus der Perspektive von Einschränkungen]

### Deliverables

[Code, Manuale, Spezifikationen, Testdaten/-ergebnisse]

### Abhängigkeiten

[als Vorbedingungen: von Systemfähigkeiten, von UseCases anderer Timeboxes, Testdaten, Benutzerinput/-einbezug]

## Timebox 2

### Benutzungsfall/fälle (UseCase(s))

### Architektur

### Deliverables

### Abhängigkeiten

## Timebox 3

### Benutzungsfall/fälle (UseCase(s))

### Architektur

### Deliverables

### Abhängigkeiten

# Glossar

[wichtige Begriffe des Domänenbereichs in kurzer Textform  definiert]