

# **Pflichtenheft**

## Zusammenfassung

Pflichtenheft der Gruppe 1 „JRMDS“ für das externe Softwareprojekt jQAssistant der Firma Buschmais GbR , durchgeführt vom Softwaretechnologielehrstuhl der TU Dresden.

## Historie

Version	Status	Datum	Erläuterungen
0.1	Grobentwurf	29.10.14	Erster Entwurf
0.2	Version nach Gespräch mit dem Kunden	05.11.14	Verbesserung Analyseklassendiagramm, Szenarien Beschreibungen, Sequenzdiagramme, Akteure hinzugefügt, fachlichen Überblick erweitert, Glossar hinzugefügt
0.3	Abschluss Analysephase	07.11.14	Use-Cases verfeinert und Sequenzdiagramme vervollständigt

## Inhaltsverzeichnis

Zusammenfassung.....	2
Historie .....	2
1. Vorangegangene Entwicklung und Zielsetzung.....	5
2. Anforderungen .....	5
2.1. Muss-Kriterien .....	5
2.2. Kann-Kriterien.....	6
3. Fachlicher Überblick .....	7
3.1. Technische Umgebung .....	7
3.2. Zielgruppe.....	7
4. Diagramme .....	8
4.1. Anwendungsfälle gekoppelt mit Sequenzdiagrammen .....	8
User registrieren.....	9
User einloggen.....	11
Userprofil bearbeiten .....	12
Suche nach Rules und deren Filterung.....	17
Rules/Groups hinzufügen .....	18
Projektbearbeitung .....	20
Rule/Group bearbeiten .....	24
Rule/Group löschen.....	25
Referenzen .....	27
Externer Server/Client ruft Gruppe aus einem Projekt ab .....	29
Einbinden externer Referenzen .....	31
Passwort vergessen .....	33
4.1. Akteure .....	35

4.2. Top-Level-Architektur.....	35
4.3. Analysediagramm .....	36
4.4. Komponentendiagramm .....	36
5. GUI Prototyp.....	37
6. Entwurfsdatenmodell.....	39
7. Testfälle .....	39
8. Glossar .....	39

## 1. Vorangegangene Entwicklung und Zielsetzung

Die buschmais GbR bietet mit dem Open-Source-Werkzeug "jQAssistant" eine flexible Lösung an, um bei der Implementierung von Softwareprojekten die Definition von Regeln, u.a. einheitliche Namenskonventionen und Strukturierung von Modulen/Programmteilen, festzulegen und deren Einhaltung während der Umsetzung kontinuierlich zu überwachen. Dafür werden die Strukturen eines gesamten Softwareprojektes eingelesen und in einer Graphendatenbank (Neo4j) abgespeichert, um diese dann mithilfe der Abfragesprache "Cypher" zu analysieren und auszuwerten. Die dafür nötigen Regeln sind derzeit als Cypher-Queries in XML-Dateien gespeichert und werden zurzeit manuell gepflegt.

Um die Bearbeitung und Administration der Regeln zu vereinfachen, soll eine zentrale Verwaltung über Repositories ermöglicht werden.

## 2. Anforderungen

### 2.1. Muss-Kriterien

MK01	Server mit grafischer Weboberfläche
MK01.1	CRUD – Erstellen, Anzeigen, Bearbeiten, Löschen
MK01.2	Suche nach ID und Beschreibung
MK02	Zuordnung von Regelsätzen zu Projekten
MK03	Anlegen/Bearbeiten/Löschen von Regeln/Gruppen/Templates über die Oberfläche
MK04	Validierung von Regeln, z.B. Prüfung auf Zyklen und verletzte Abhängigkeiten
MK05	Abbildung der Beziehungen/Referenzen zwischen Objekten (Auswahlmenü, ggf. Drag&Drop)
MK06	Jedes Projekt kann weitere Regeln importieren
MK06.1	lokaler Server: um Regeln aus einem anderen Projekt zu übernehmen
MK06.2	entfernter Server: Import über XML-Schnittstelle
MK07	Lokale Regeln können entfernte Regeln überschreiben
MK07.1	Überschreiben muss in der UI sichtbar sein

MK08	Bereitstellung von Regeln als REST-Ressource
MK08.1	XML: Build-Clients (z.B. Maven-Plugin), Repositories und IDEs (z.B. zukünftiges Eclipse-Plugin)
MK08.2	JSON: UI-Client (Browser)
MK09	Speicherung der Regelsätze in einer Neo4j Datenbank

## 2.2. Kann-Kriterien

KK01	Import von Regeln aus bestehenden XML-Dateien
KK02	Abgleich von Regeln zwischen Repositories
KK02.1	Online vs. Offline
KK03	Benutzerverwaltung
KK03.1	Login und Rechteverwaltung auf Projektbasis
KK03.2	Rechte für Projekte: für alle lesbar (auch Gäste), bearbeiten nur für registrierte Benutzer
KK04	um Sortierung und Übersicht zu wahren, kann jede Regel/Gruppe/Template usw. mit mehreren Tags versehen werden

### 3. Fachlicher Überblick

Umzusetzen ist eine Serverlösung mit GUI zum Verwalten der Regeln bzw. Filter

#### 3.1. Technische Umgebung

Die Anforderungen sollen mit einem auf Java basierenden **Server** umgesetzt werden, der mit einer Webanwendung kommuniziert. Das Build-Management-Tool Maven wird benutzt, um Tests und den Build-Vorgang zu automatisieren sowie um Online-Dependencies zu verwalten. Diese wird mit HTML, JavaScript und JQuery umgesetzt, es wird ein Browser benötigt. Die Graphendatenbank neo4j wird verwendet, um die Strukturen der Software und die Regeln zu speichern. Die Webanwendung kommuniziert mit der Datenbank. Das mittels eines Spring-Frameworks erstellte REST-Interface bearbeitet die Anfragen und das Herausgeben der Regeln, wenn nötig werden sie über DocumentBuilder geparkt. Die Übertragung der Daten zwischen Client und Server werden mit JSON in Verbindung mit Ajax realisiert. Die Architektur lehnt sich an das MVC an mit Spring Web MVC.

UML - Visualisierungen, welche während der Analysephase entstehen, werden mit dem Eclipse-Plugin [Papyrus] und MagicDraw erstellt.

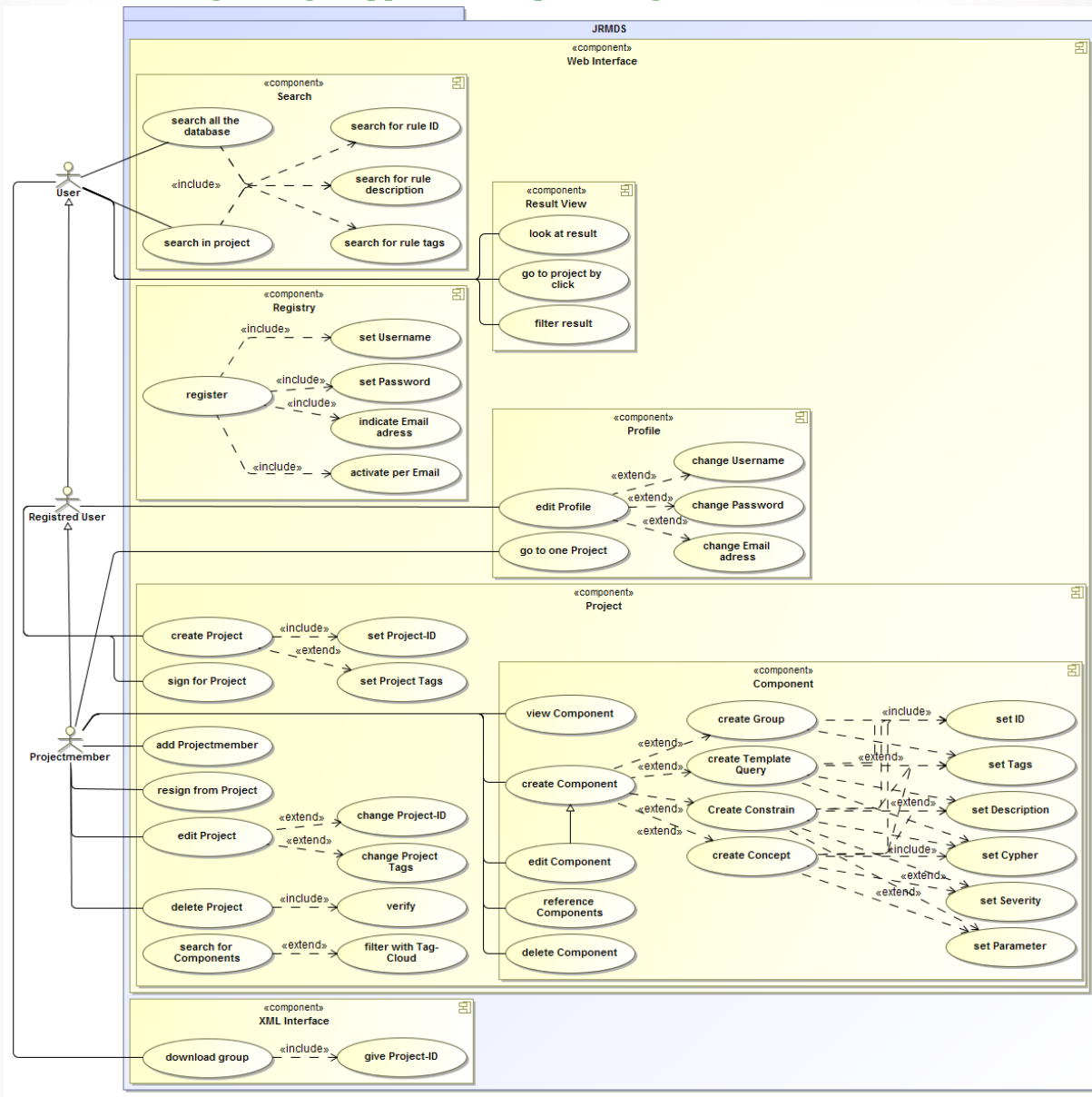
Die Benutzer sollten als **Client** über eine Internetverbindung verfügen und einen HTML5 fähigen Browser besitzen. Maus und Tastatur werden zur Bedienung der GUI empfohlen.

#### 3.2. Zielgruppe

Die Zielgruppe umfasst Softwareentwickler, die ihren Code auf Wartungsfähigkeit, Performance und Validität prüfen möchten.

## 4. Diagramme

### 4.1. Anwendungsfälle gekoppelt mit Sequenzdiagrammen





### User registrieren

Der User möchte sich registrieren. Dazu klickt er auf der Hauptseite im Login Bereich auf "Registrieren".

Ein Registrieren-View wird geöffnet, wo er seinen gewünschten Usernamen in ein Textfeld eingeben kann. Das JRMDS prüft in der User-Datenbank nach, ob der Username bereits vorhanden ist.

#### 1) Username ist vergeben

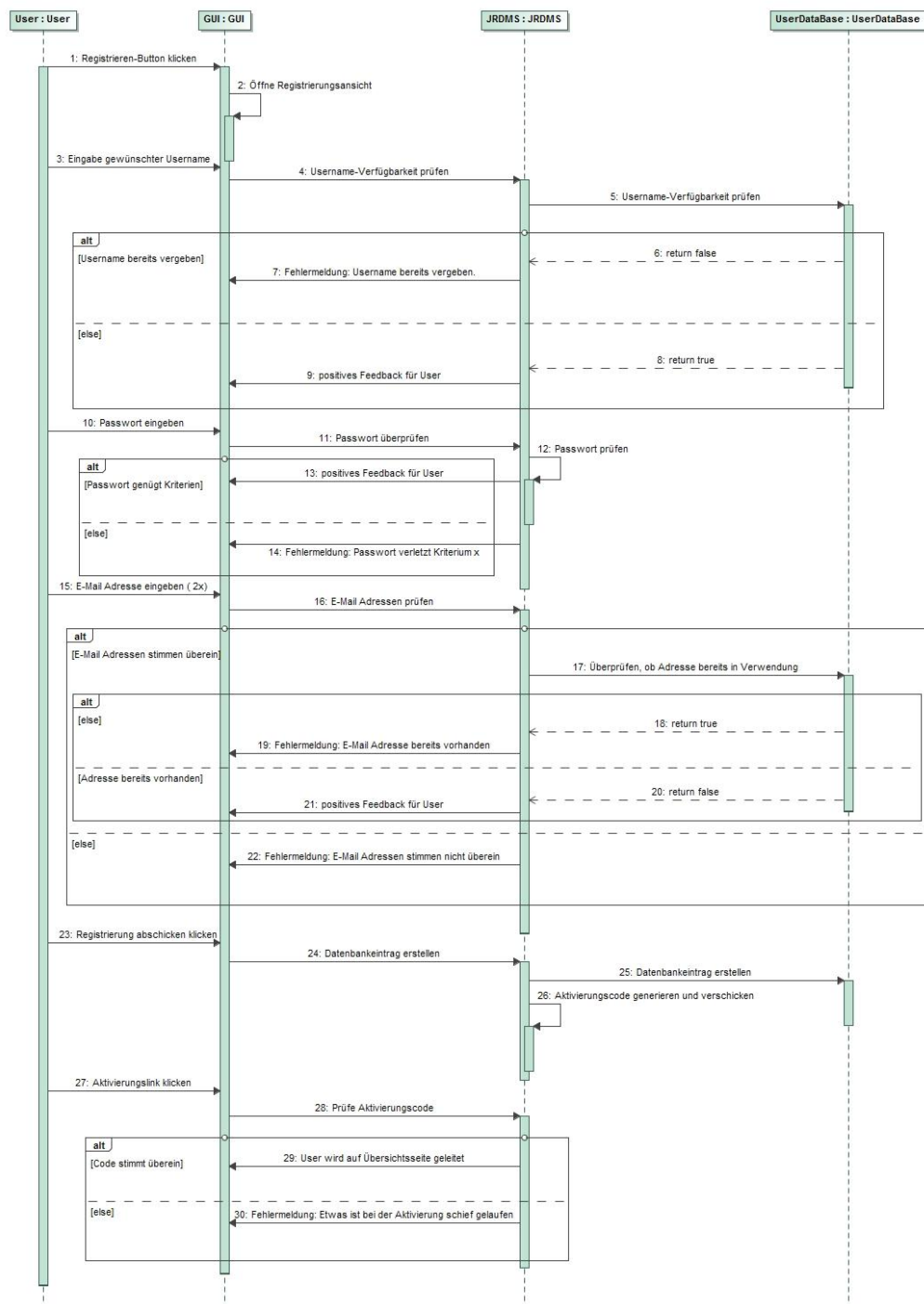
Das JRMDS macht den Nutzer auf den vergebenen Usernamen aufmerksam. Er muss einen alternativen Namen eingeben.

#### 2) Username ist nicht vergeben

Der User gibt ein Passwort ein. Das JRMDS prüft, ob das Passwort bestimmten Kriterien genügt.

Der User muss eine E-Mail-Adresse eingeben (2-Mal). Das JRMDS prüft, ob die beiden Eingaben übereinstimmen und ob die Adresse nicht bereits in Verwendung ist.

Wenn alles in Ordnung ist, so wird in der Datenbank ein Eintrag für den User erstellt mit einem zufällig generierten Aktivierungscode und eine Bestätigungsmail vom JRMDS versandt. Der User muss den Aktivierungslink anklicken. Das JRMDS prüft den Aktivierungscode auf Gültigkeit und schaltet den User bei Übereinstimmung frei. Er wird dann auf die Übersichtsseite weitergeleitet.



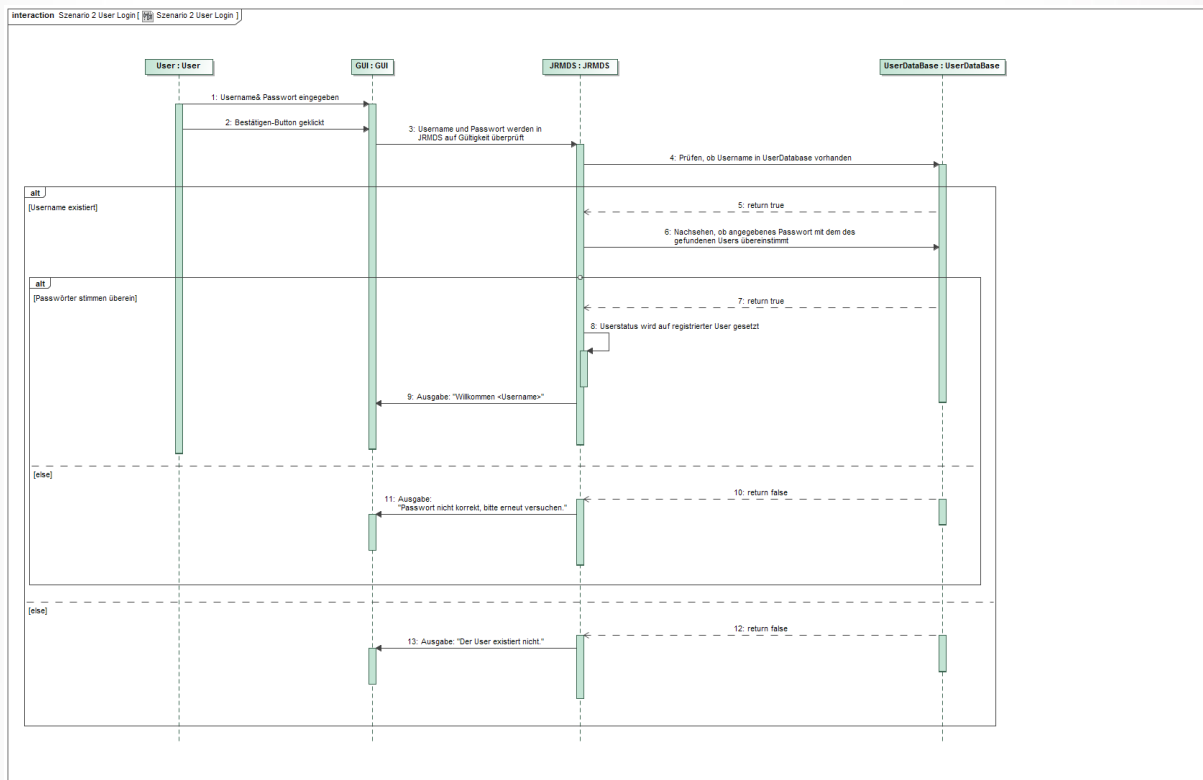
## User einloggen

Der User gibt auf der Startseite im Loginbereich seinen Usernamen und sein Passwort ein. Nach einem Klick auf den Bestätigen-Button wird der Username in der User Datenbank gesucht. Ist dieser nicht vorhanden, wird die Meldung "Der User existiert nicht" ausgegeben.

Wenn der Username existiert, wird das vom User eingegebene Passwort mit dem in der Datenbank hinterlegten abgeglichen.

Stimmen diese nicht überein wird die Meldung "Passwort nicht korrekt, bitte erneut versuchen." ausgegeben.

Ist der Abgleich korrekt bekommt der User den Status registrierter User und damit Zugriff auf sein Profil und die damit verknüpften Inhalte.



## Userprofil bearbeiten

Jeder registrierte User hat ein Profil, welches vom Admin und dem zugehörigen User bearbeitet werden kann. Einzusehen ist das Profil, mitsamt den Projektzugehörigkeiten, von jedem anderen User. Dazu muss man auf den Usernamen des Users klicken. Dort finden dann die User mit entsprechenden Berechtigungen den Bearbeiten- Button.

Der Admin kann Username, Passwort und Emailadresse anderer User unter denselben Bedingungen ändern. Wird der Bearbeiten-Button angeklickt, öffnet sich ein Bearbeiten-View, in der User ihr Passwort und ihre Emailadresse ändern können.

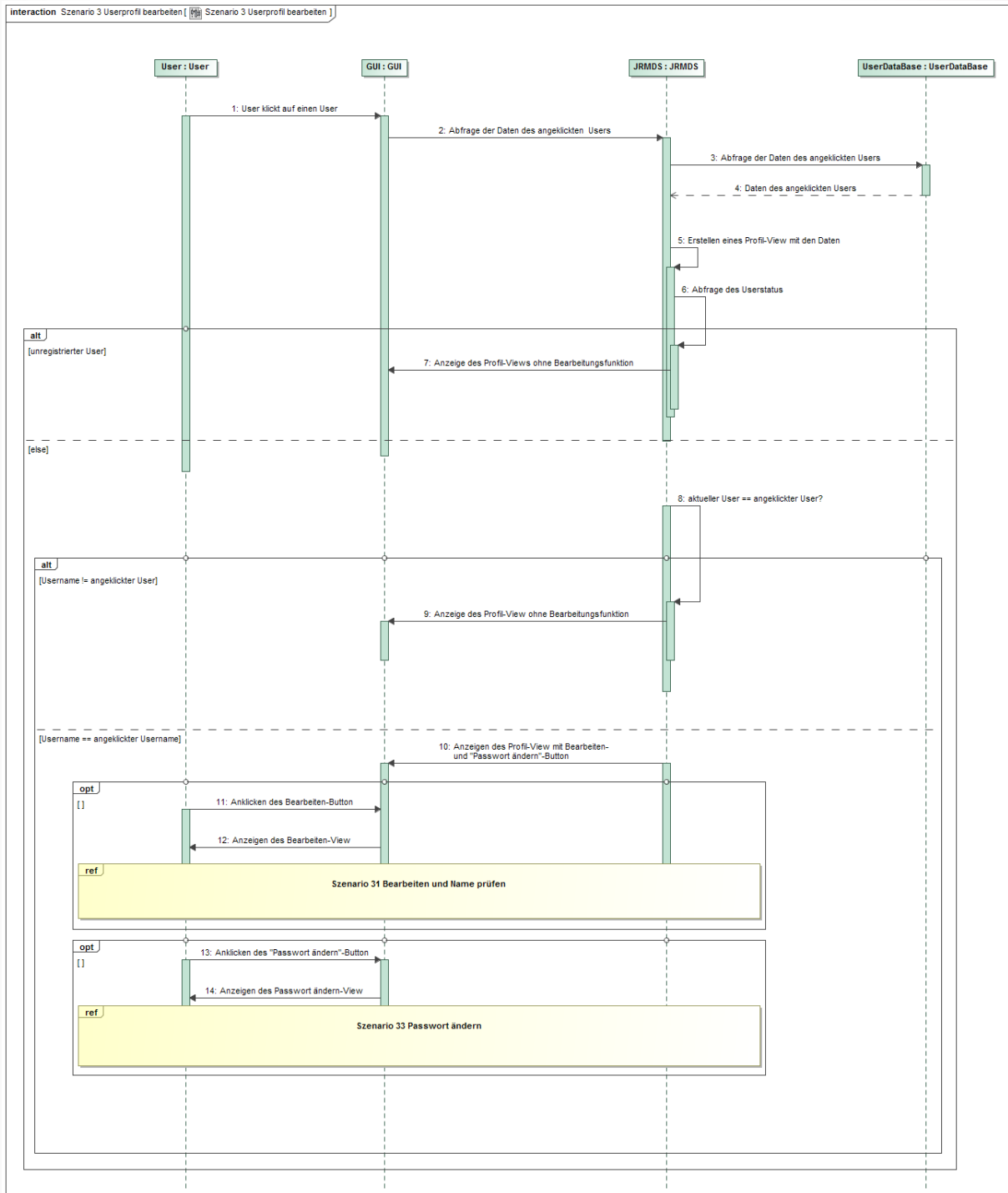
1) Möchten User ihr Passwort ändern, müssen sie ein neues doppelt eingeben. Mit einem Klick auf den Bestätigen-Button kontrolliert die JRMDs, ob die eingegebenen Passwörter übereinstimmen und bestimmte Voraussetzungen erfüllen.

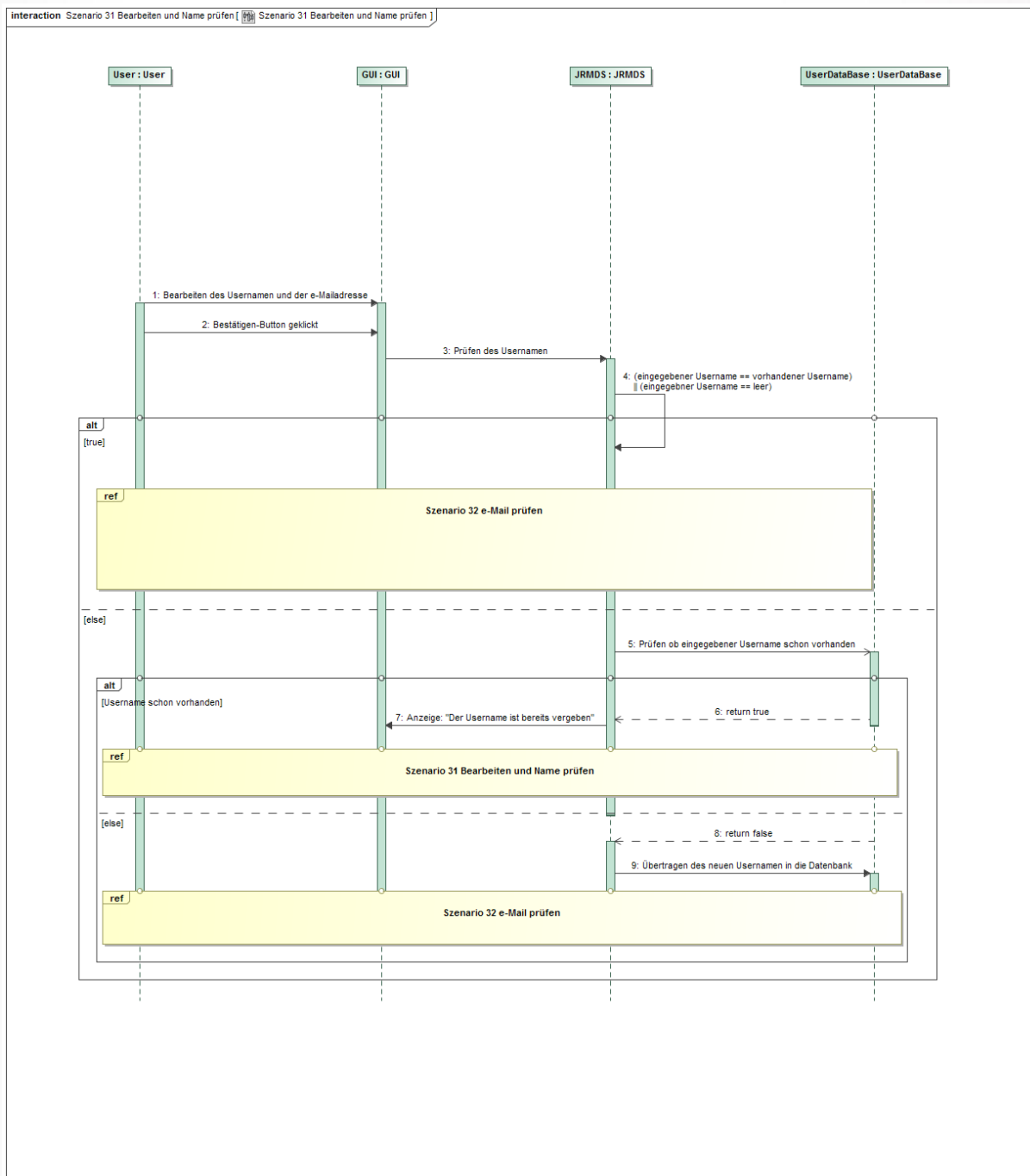
Entspricht das erste Passwort nicht den Voraussetzungen, wird die Nachricht "Das Passwort genügt nicht den Voraussetzungen." ausgegeben.

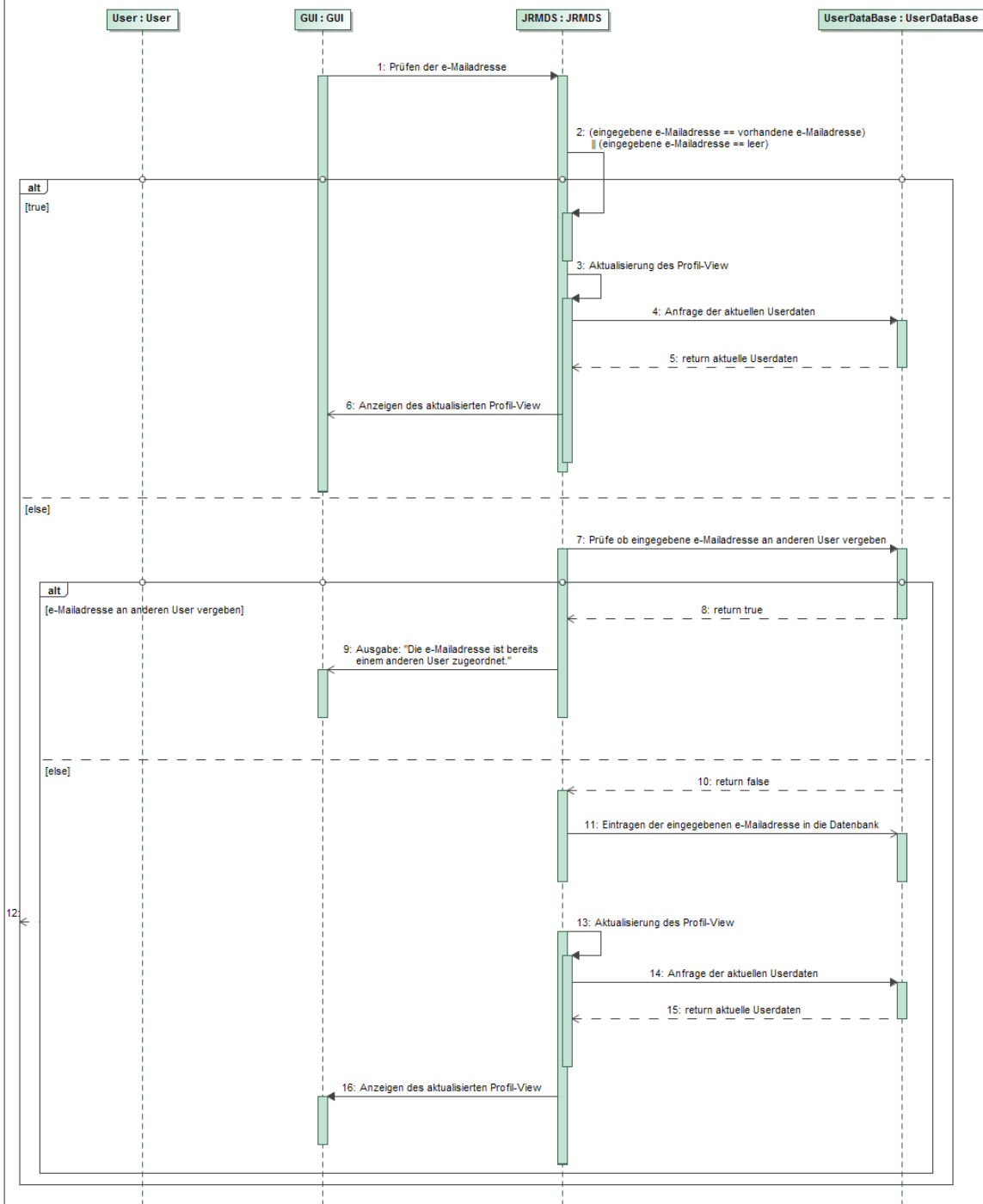
Stimmen die beiden Passwörter nicht überein, wird die Nachricht "Die Passwörter stimmen nicht überein." ausgegeben.

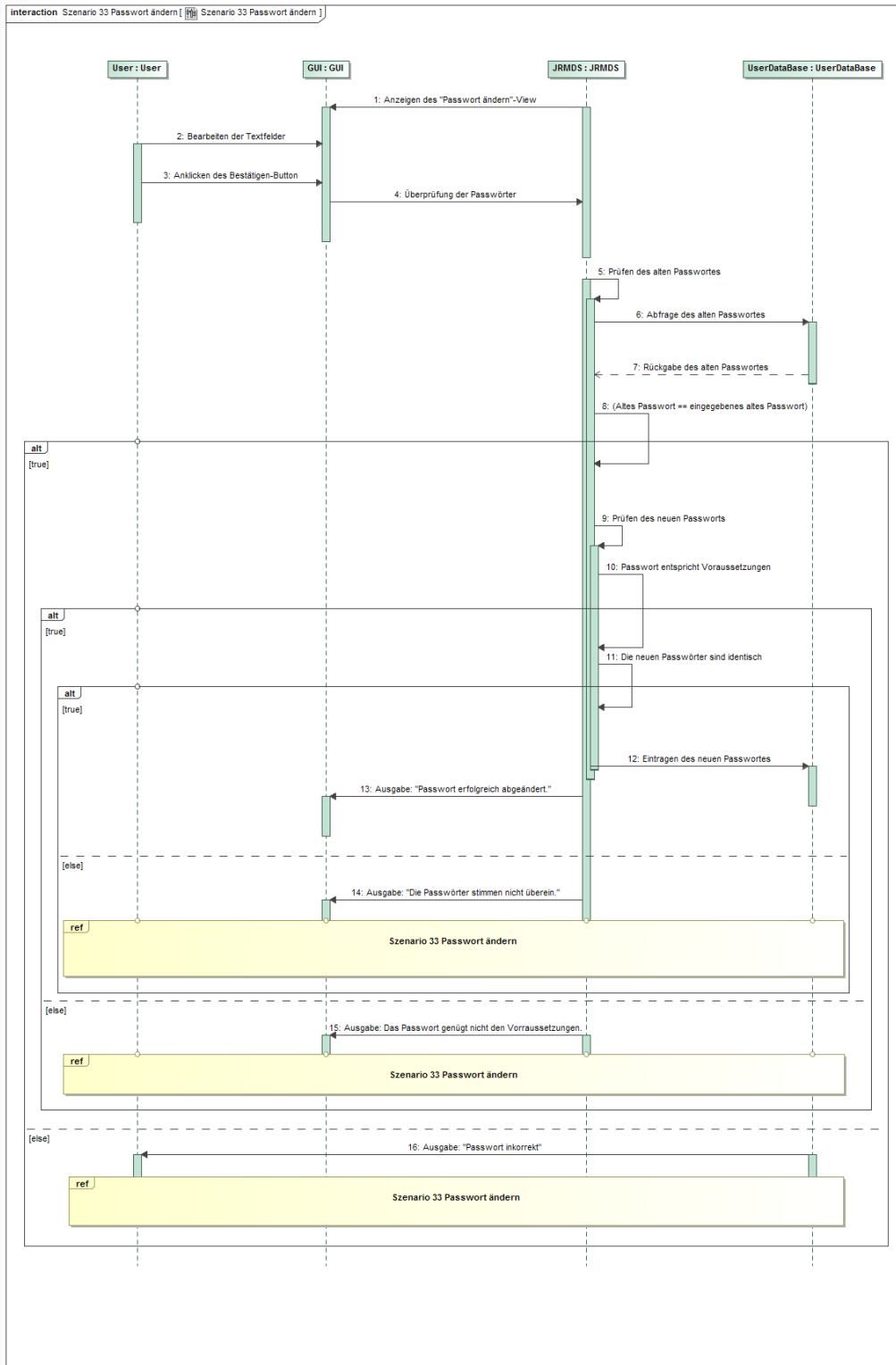
2) Möchte ein User seine e-Mailadresse ändern, muss er die alte e-Mailadresse mit der neuen ersetzen. Mit einem Klick auf den Bestätigen-Button überprüft JRMDs, ob die e-Mailadresse bereits in der User-Datenbank hinterlegt ist. Ist die e-Mailadresse bereits hinterlegt, wird eine Nachricht ausgegeben "Die e-Mailadresse ist bereits einem anderen User zugeordnet".

Gibt es nach dem Klick des Bestätigen-Button keine Benachrichtigungen, übernimmt JRMDs die Änderungen des Profils in die Userdatenbank. Der User wird zu seiner Profilseite weitergeleitet. Ein Abbrechen-Button ermöglicht jederzeit während der Bearbeitung des Profils eine Rückkehr zur Profilseite ohne Änderungen zu übernehmen.









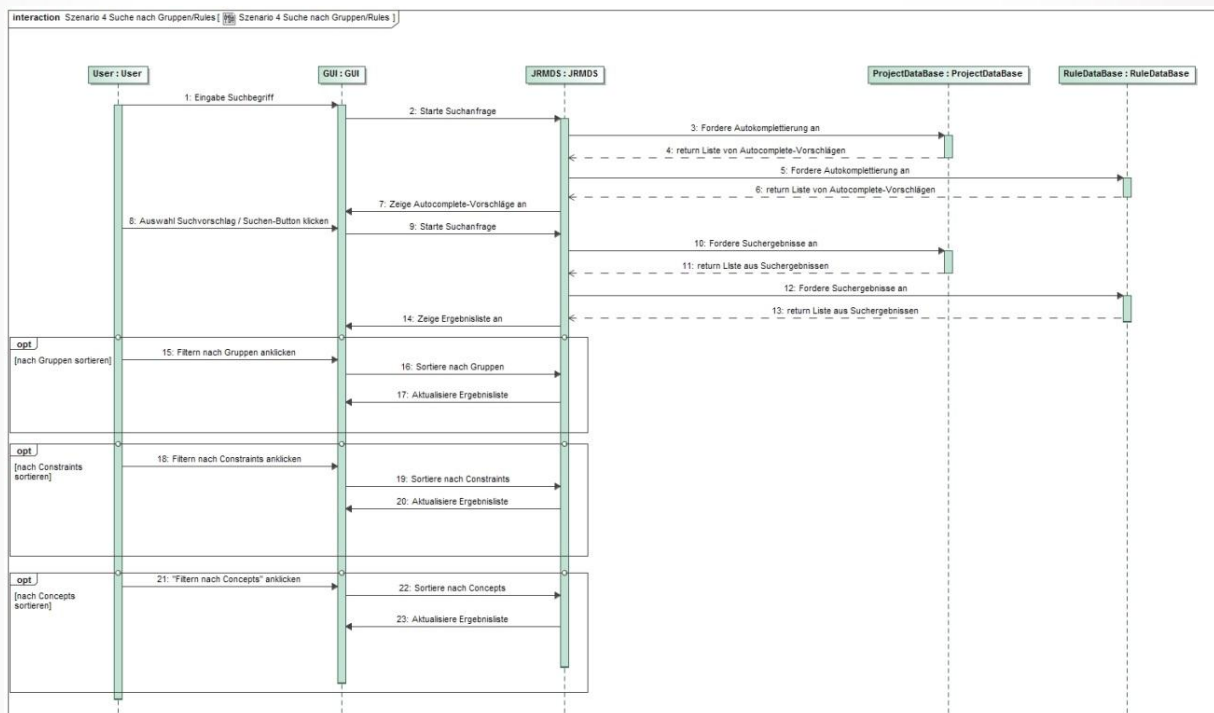


## Suche nach Rules und deren Filterung

Der User befindet sich auf der Seite und möchte nach einer Rule oder einer Group suchen. Dazu gibt er einen Suchbegriff in das Suchfeld ein. Das JRMDs reagiert auf die Tasteneingabe und gleicht diese mit der Datenbank ab. Dem User werden sofort sinnvolle automatische Komplettierungsvorschläge geliefert. Wählt der User einen der Vorschläge aus und bestätigt seine Auswahl mit der Enter-Taste bzw. durch Klicken des Suchen-Buttons, so stellt das JRMDs eine Liste der relevantesten Suchergebnisse zusammen und zeigt diese auch an. Der User hat nun optional die Möglichkeit, die Ergebnisliste nach Groups, Constraints, Concepts zu ordnen.

- 1) Ordnen nach Groups liefert nur Groups als Ergebnis
- 2) Ordnen nach Constraints liefert nur Constraints als Ergebnis
- 3) Ordnen nach Concepts liefert nur Concepts als Ergebnis

Alle Suchergebnisse werden mit zusätzlichen Informationen angefüllt (zum Beispiel Beschreibung, ID, etc....)



### Rules/Groups hinzufügen

Über die Seite navigiert der User zu seinem Projekt. Dort findet er die Buttons "Group erstellen" und "Rule erstellen".

#### 1) Group erstellen

Der User möchte eine Group erstellen. Dazu befindet er sich auf der Seite des Projektes und klickt auf den Button "Group erstellen". Dies öffnet den View zum Erstellen einer neuen Group. Hier wird der User aufgefordert, spezifische Angaben bzgl. der neuen Group zu machen. (ID, Tags, Referenz (siehe Szenario Referenzen definieren))

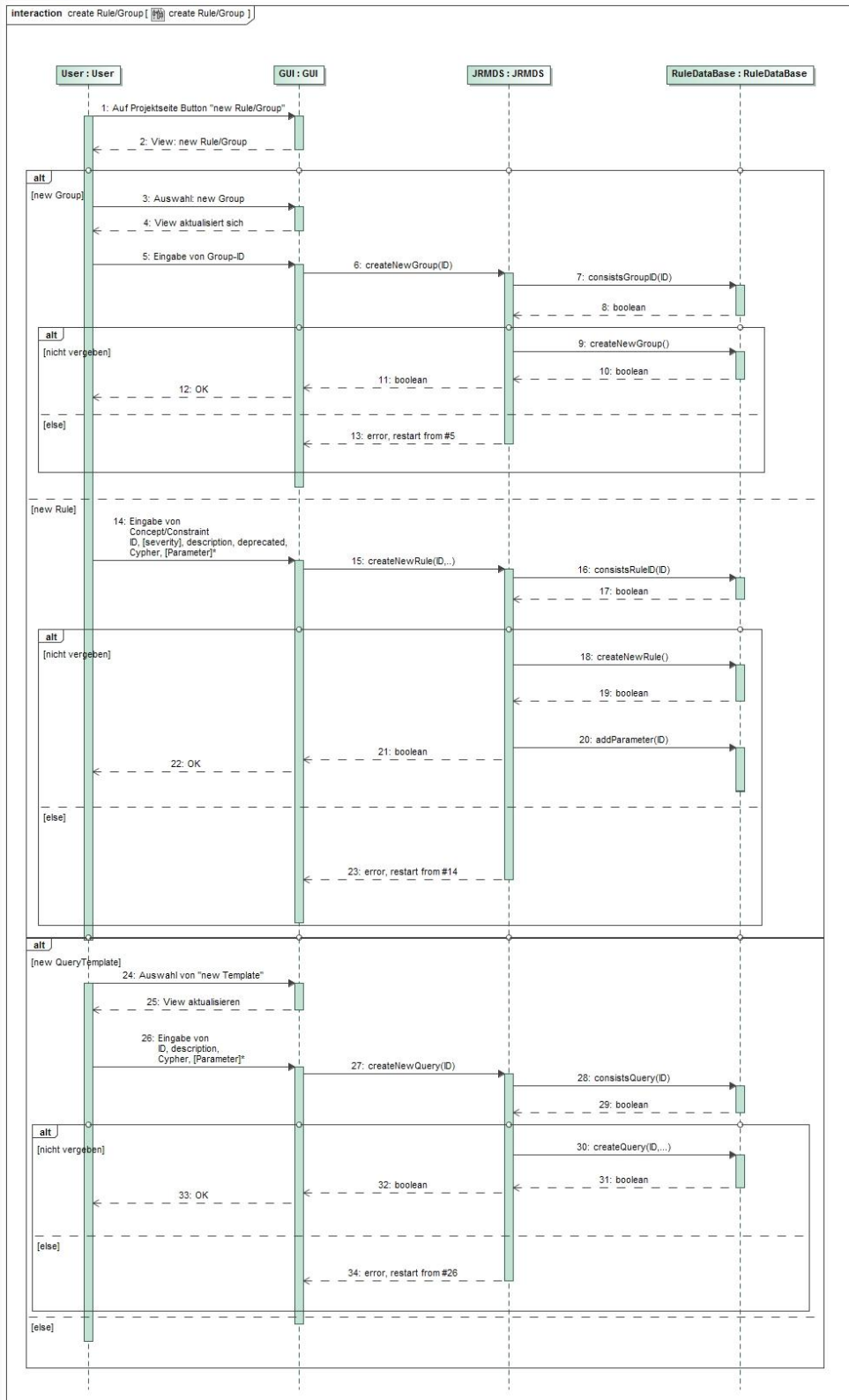
#### 2) Regel hinzufügen

Der User möchte eine Rule erstellen. Dazu befindet er sich auf der Seite des Projektes und klickt auf den Button "Rule erstellen". Dies öffnet den View zum Erstellen einer neuen Rule. Hier wird der User aufgefordert, spezifische Angaben bzgl. der neuen Rule zu machen. (ID, Tags, Cypher, Severity, Referenz (siehe Szenario Referenzen definieren))

Das JRMDs überprüft die Eingaben auf ihre Gültigkeit nach dem Anklicken des Bestätigen-Button. Ist die ID bereits in der Rule-Datenbank vergeben, so wird eine Nachricht ausgegeben "Die ID ist bereits vergeben".

Wenn die Eingaben valide sind, wird ein neuer Eintrag in der Project-Datenbank mit den Eingaben des Users getätigt.

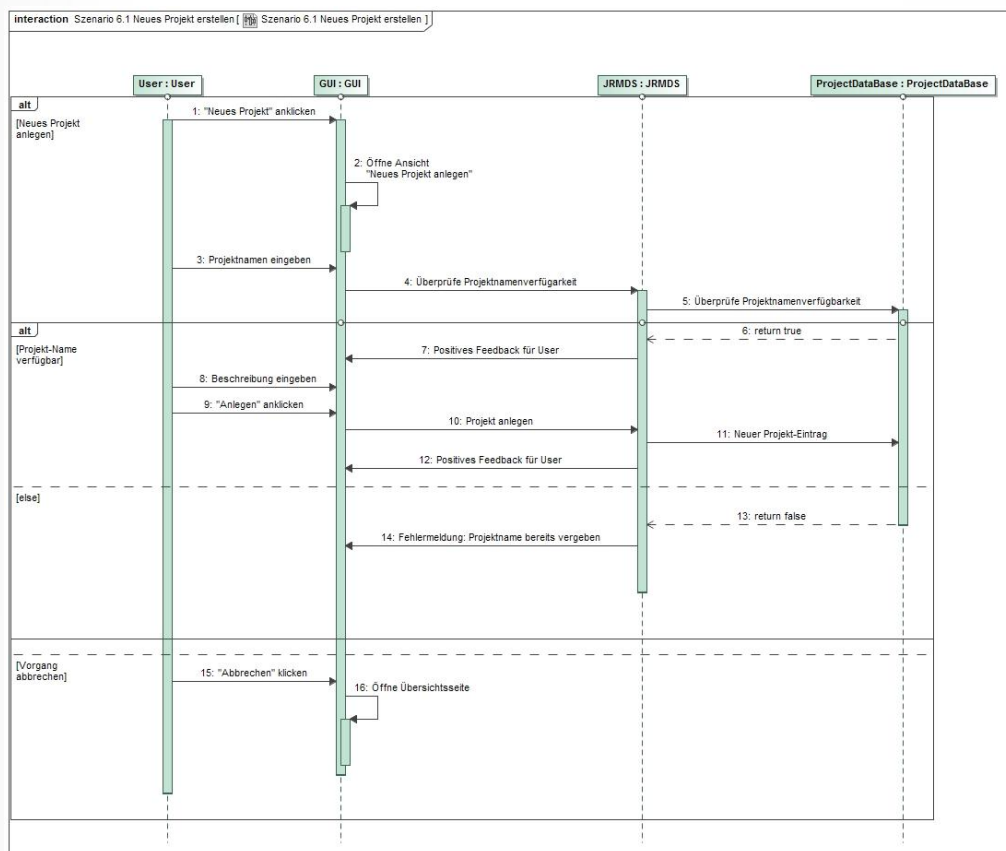
Der Vorgang kann jederzeit über einen Abbrechen-Button abgebrochen werden.



## Projektbearbeitung

### 1) anlegen

Ein registrierter User kann über die Schaltfläche "Erstelle neues Project" ein neues Project anlegen. Ein View zum Erstellen eines neuen Projects erscheint. Hier kann der User einen Project-Namen und optional eine Project-Beschreibung angeben. Nach dem Anklicken des Bestätigen-Button kontrolliert JRMDS, ob der Name bereits für ein Project in der Project-Datenbank vergeben ist. Sollte dieser schon existieren, erscheint die Meldung "Project-Name ist bereits vergeben". Ist der Name frei, wird für das Project in der Project-Datenbank ein Eintrag angelegt. Über einen Abbrechen-Button kann das Anlegen eines neuen Projects jederzeit abgebrochen werden. Die eingetragenen Daten werden beim Anklicken verworfen und der User gelangt zurück zur Hauptseite.



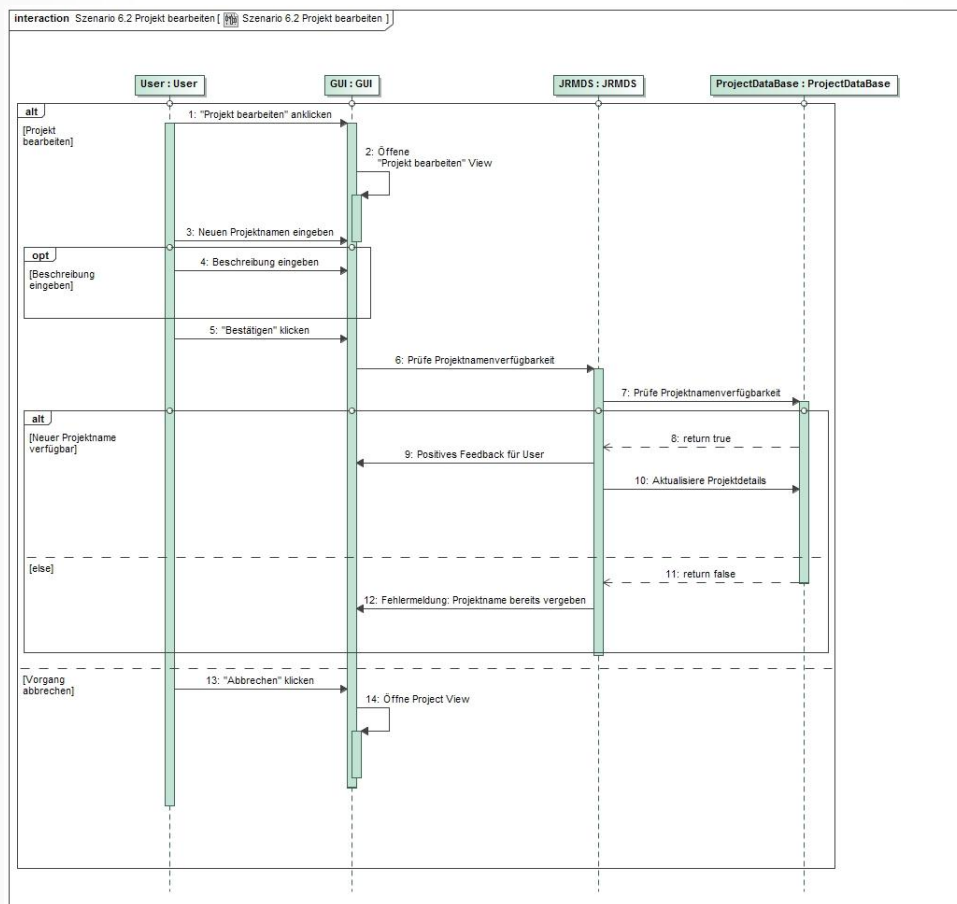
### 2) bearbeiten

Im Project View kann jeder User, der den Status Project-Member hat, über den Button "Project bearbeiten" das Project bearbeiten. Beim Anklicken des Buttons öffnet sich ein neuer View, in dem der Project-Name und die Beschreibung geändert werden können.

Wird der Bestätigen-Button angeklickt, kontrolliert JRMDS, ob der neue Project-Name bereits in der Project-Datenbank vorhanden ist. Ist dies der Fall, erscheint die Meldung "Project-Name ist bereits vergeben".

Wenn der Name noch frei ist, werden die Daten zu dem Project in der Project-Datenbank entsprechend abgeändert.

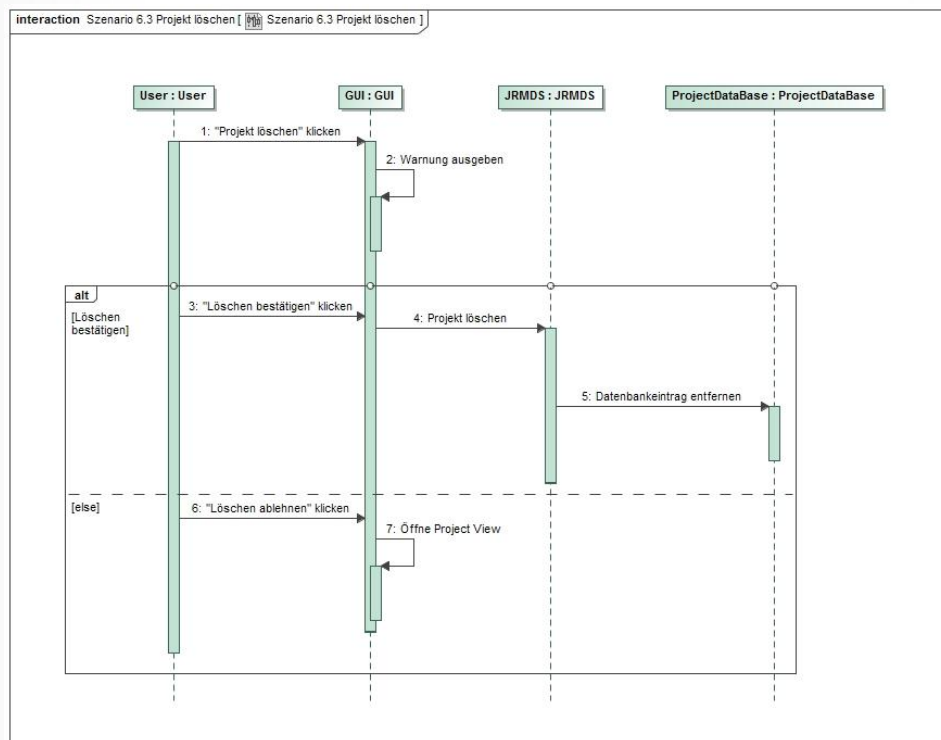
Über einen Abbrechen-Button kann das Bearbeiten des Projects jederzeit abgebrochen werden. Die eingetragenen Daten werden beim Anklicken verworfen und der User gelangt zurück zum Project View.



### 3) löschen

Im Project View kann mit dem Button "Project löschen" das gesamte Projekt gelöscht werden. Das Löschen eines Projekts muss in einem neuen View mit der Meldung "Sind Sie sicher?" bestätigt oder abgelehnt werden.

Wird abgelehnt, wird der Vorgang abgebrochen. Bei Bestätigung werden alle gespeicherten Inhalte zu dem Project aus der Project-Datenbank entfernt.



#### 4) Mitglieder hinzufügen

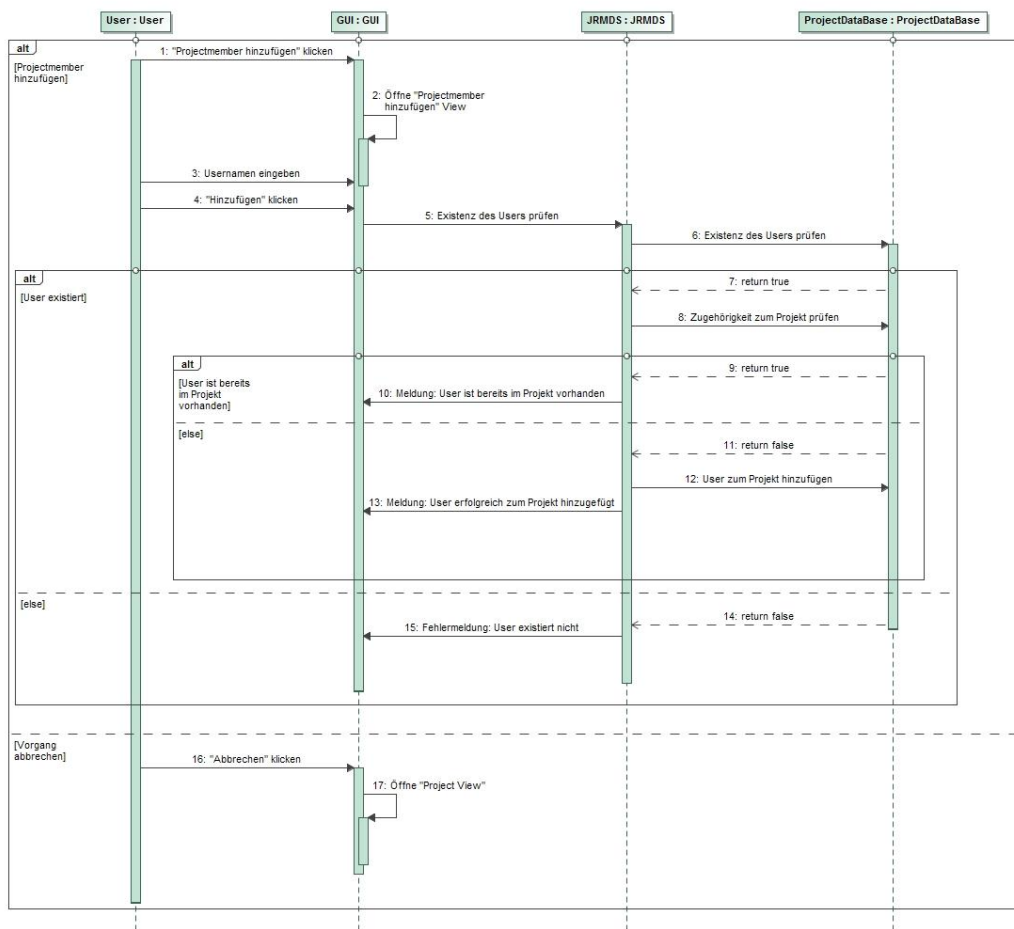
Mit dem Button "Projectmember hinzufügen" im Project View kann ein neues Projectmember hinzugefügt werden. Dazu muss in einem gesonderten "Projectmember hinzufügen" View der Username eingegeben werden. Nach dem Bestätigen über einen Bestätigen-Button kontrolliert JRMDS ob dieser in der User-Datenbank vorhanden ist.

Ist dieser dort nicht vorhanden, wird "User existiert nicht" ausgegeben.

Wenn der Username vorhanden ist, wird der Status des Users für dieses Project auf Projectmember gesetzt und die entsprechende Verlinkung in der Project-Datenbank gesetzt.

Sollte der User schon Projectmember für dieses Project sein, wird "User ist bereits ein Projectmember" ausgegeben.

Über einen Abbrechen-Button kann der Vorgang des Hinzufügens eines neuen Projectmembers jederzeit abgebrochen werden. Nach Anklicken dieses Buttons gelangt der User zurück zum Project View.





### Rule/Group bearbeiten

Möchte ein Projectmember eine Rule oder Group bearbeiten, muss er auf der Webseite korrekt eingeloggt sein und in seinem Project die Rule/Group anwählen. Der registrierte User kann den Bearbeiten-Button anklicken und gelangt somit in den Bearbeiten-View.

#### 1) Regelbearbeitung:

Das Projectmember kann ID, Tags, Severity, Description und Cypher direkt in einem Textfeld bearbeiten. Klickt man in das Textfeld von ID und Cypher, erscheint ein Warnungs-View, welcher auf die Verweise und möglichen Beeinträchtigung anderer Groups hinweist.

#### 2) Gruppenbearbeitung:

Das Projectmember kann ID, Tags und Referenz in einem Textfeld bearbeiten. Klickt man in das Textfeld von ID, erscheint ein Warnungs-View, welcher auf die Verweise und möglichen Beeinträchtigung anderer Groups hinweist.

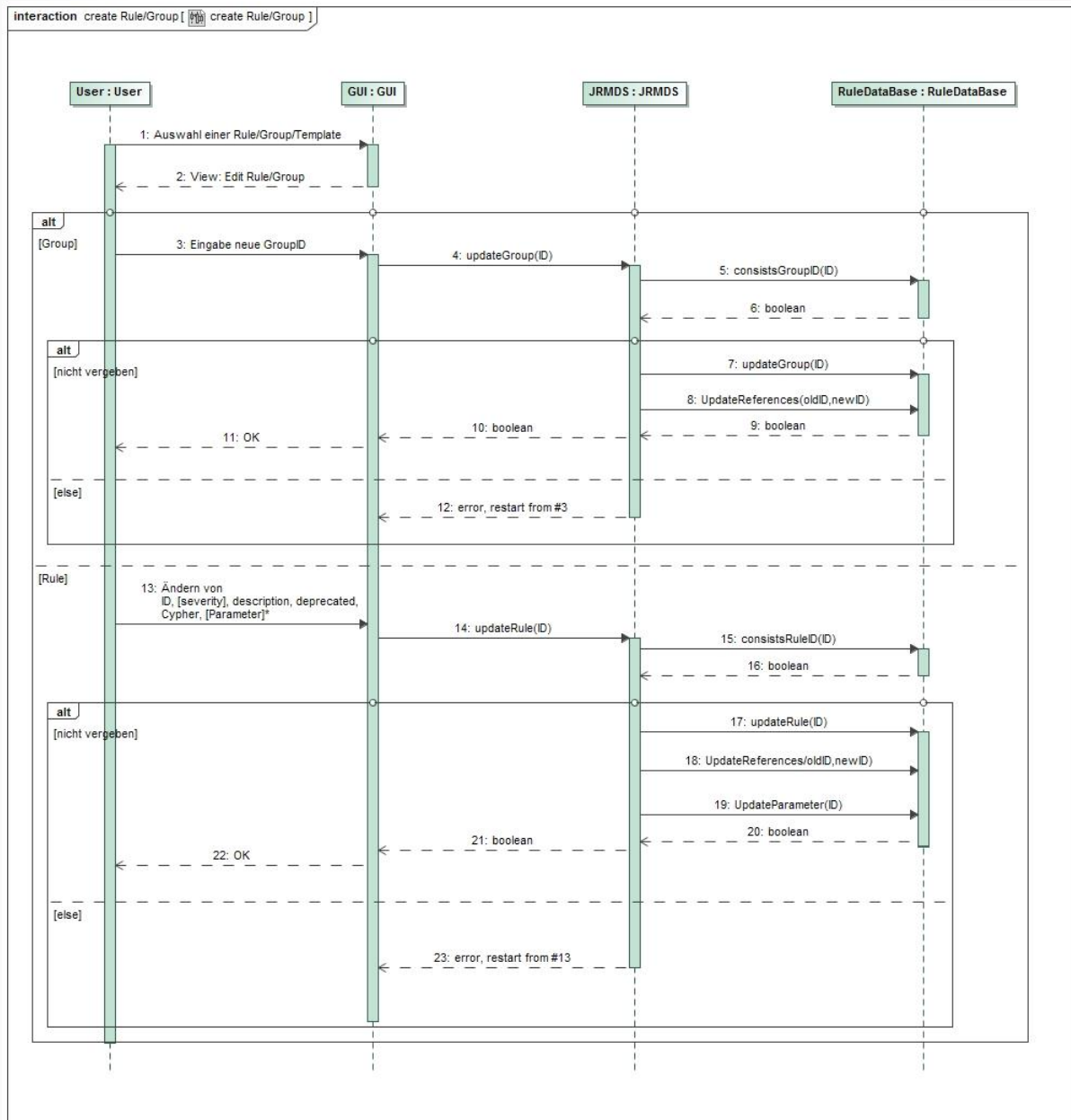
Nach dem Anklicken des Bestätigen-Button im Warnungs-View, wird dieser geschlossen und das Projectmember kann den Text editieren. Der Fokus bleibt auf dem Bestätigen-Button, bis er geklickt wurde.

Wird im Bearbeiten-View der Bestätigen-Button angeklickt überprüft das JRMDS die Eingaben auf ihre Gültigkeit.

Ist die ID bereits in der Rule-Datenbank vergeben, so wird eine Nachricht ausgegeben "Die ID ist bereits vergeben". Wenn die Eingaben valide sind, wird der entsprechende Rule Eintrag in der Rule-Datenbank mit den neuen Eingaben des Users versehen.

Der registrierte User kann den Bearbeitungsvorgang abbrechen, indem er auf den Abbrechen-Button klickt. Nach dem erfolgreichen Bestätigen oder Abbrechen gelangt er wieder zur Übersicht der Rules.





## Rule/Group löschen

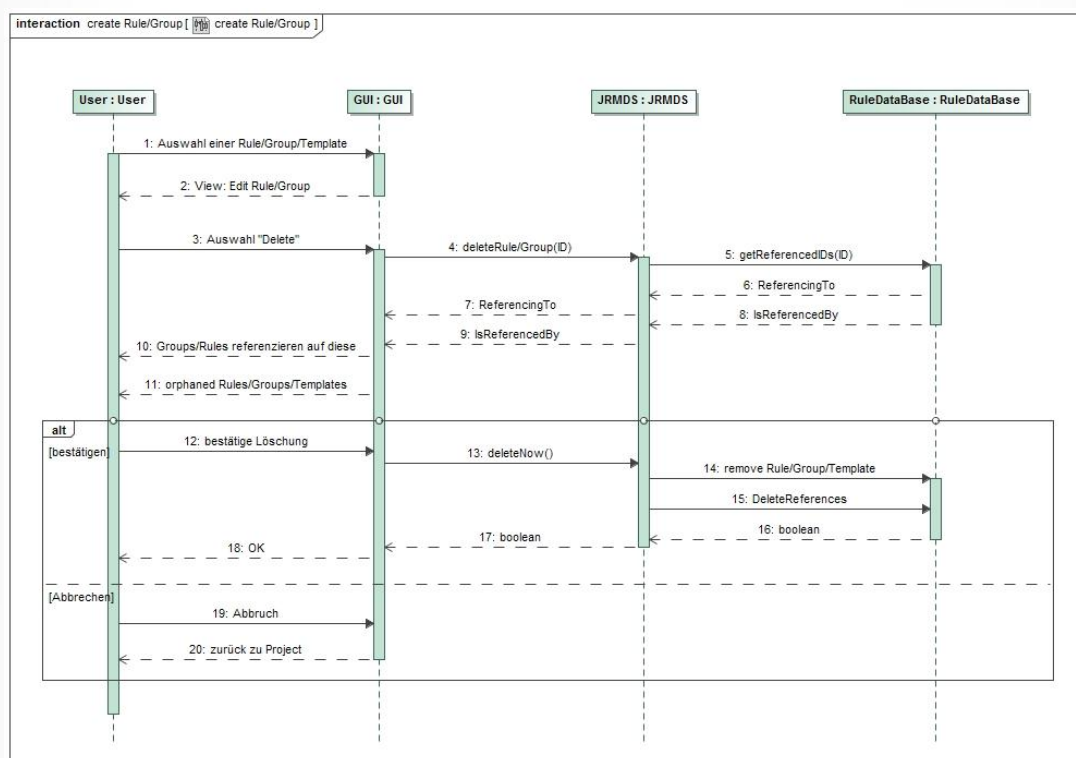
Möchte ein Projectmember eine Rule oder eine Group aus seinem Project löschen, muss er im View seines Project sein. Der Projectmember kann die Rule oder die Group seines Project anwählen und über einen Löschen-Button herauslöschen.

Eine Warnungs-View öffnet sich, sobald das Projectmember auf den Button klickt:

- 1) Rule: Der View zeigt alle Groups an, die diese Rule beinhalten und durch Abänderung beeinflusst werden.
- 2) Group: Der View zeigt alle Groups an, die der zu löschenden untergeordnet und übergeordnet sind und so durch das Löschen potentiell verwaisen werden.

Bestätigt das Projectmember in dem Warnungs-View das Löschen über einen Bestätigen-Button, wird die ausgewählte Group bzw. Rule gelöscht. Bricht das Projectmember den Löschvorgang ab mit dem Abbrechen-Button ab, schließt sich der neue View.

Betätigt das Projectmember im Löschen-View den Abbrechen-Button, kehrt es zurück zum Project-View.



## Referenzen

Ein Projectmember befindet sich innerhalb von einem Project und möchte für eine bestimmte Group oder Rule References hinzufügen. Dazu klickt er auf die entsprechende Group oder Rule, wodurch sich ein View mit den Eigenschaften öffnet.

Es gibt ein Eingabefeld für eine ID um die Reference zu anderen Groups oder Rules zu definieren. Dieses Eingabefeld bietet automatisch Vorschläge zur Vervollständigung an, die während der Eingabe angezeigt werden.

Alternativ kann über einen "Suchen" Button neben dem Eingabefeld auch nach Rules oder Groups gesucht werden. Diese Suche beschränkt sich auf alle Rules und Groups innerhalb des Projects. Die Ergebnisse können auf die gleiche Art gefiltert werden, wie bei der regulären Suche. Mithilfe von Checkboxen können mehrere Rules oder Groups ausgewählt werden, um eine Reference zu diesen zu erstellen.

Sollte eine Group geöffnet sein, dann wird die Autovervollständigung und die Suche weitere Groups, Concepts oder Constraints anzeigen.

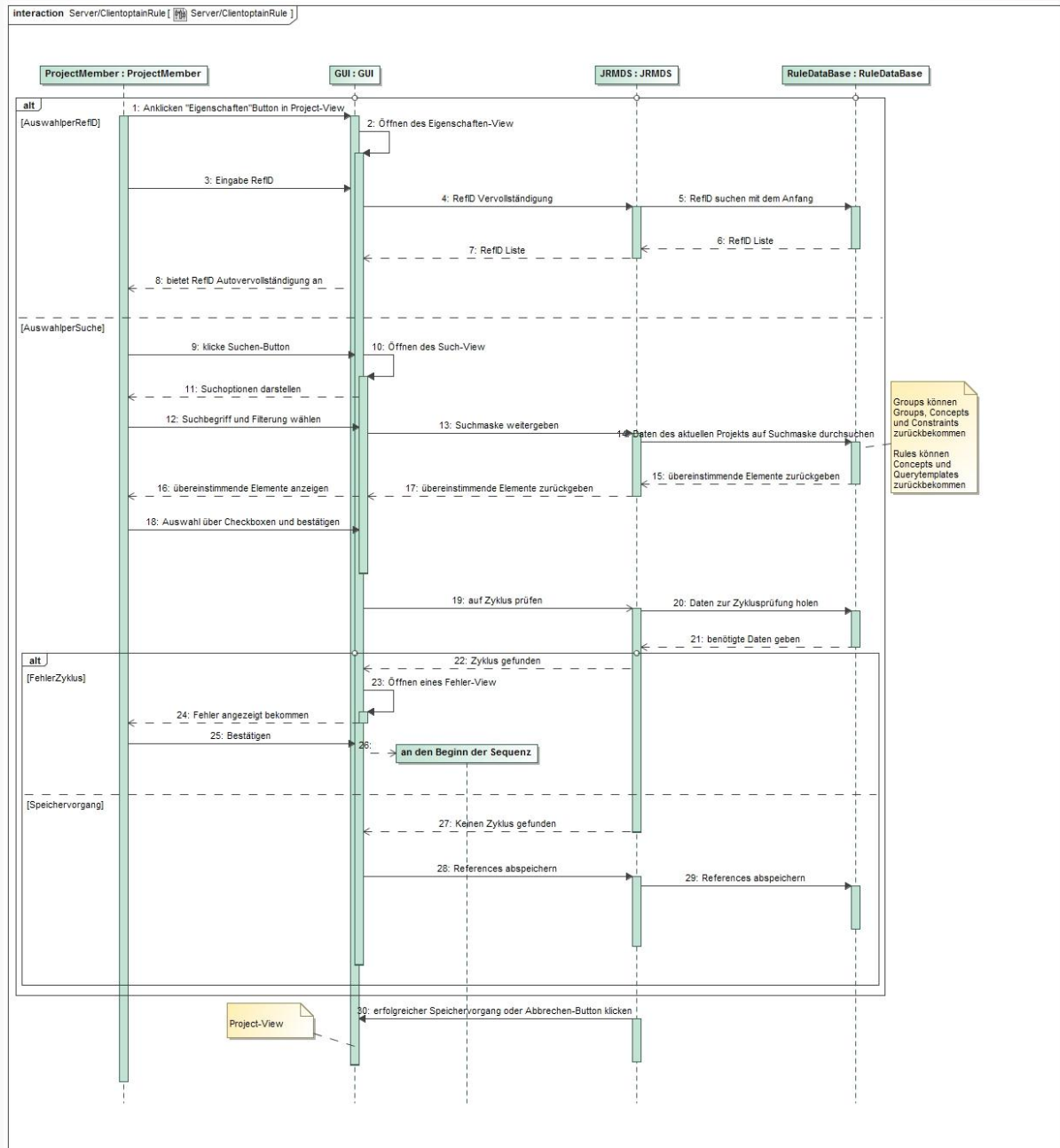
Sollte eine Rule geöffnet sein, dann werden nur Concepts und Querytemplates angezeigt.

Innerhalb des Suchfensters kann durch die Filter weiter eingegrenzt werden.

Wenn References durch Anklicken eines Bestätigen-Button hinzugefügt werden, wird von dem Server im Hintergrund geprüft, ob es einen Zyklus gibt und in diesem Falle eine Fehlermeldung ausgegeben, ohne die Reference zu speichern.

Andernfalls werden die References abgespeichert.

Das Projectmember kann jederzeit über einen Abbrechen-Button das Hinzufügen von Referenzen abbrechen und so zum Project-View zurückgelangen.



### Externer Server/Client ruft Gruppe aus einem Projekt ab

Externe Server / Client schickt eine Anfrage für eine bestimmte Group aus einem Project. Das JRMDs ruft Informationen zu dem Project ab und prüft, ob der Client für den Zugriff authentifiziert ist. Wenn dies nicht der Fall ist, wird ein HTTP-Statuscode (401) zurückgegeben.

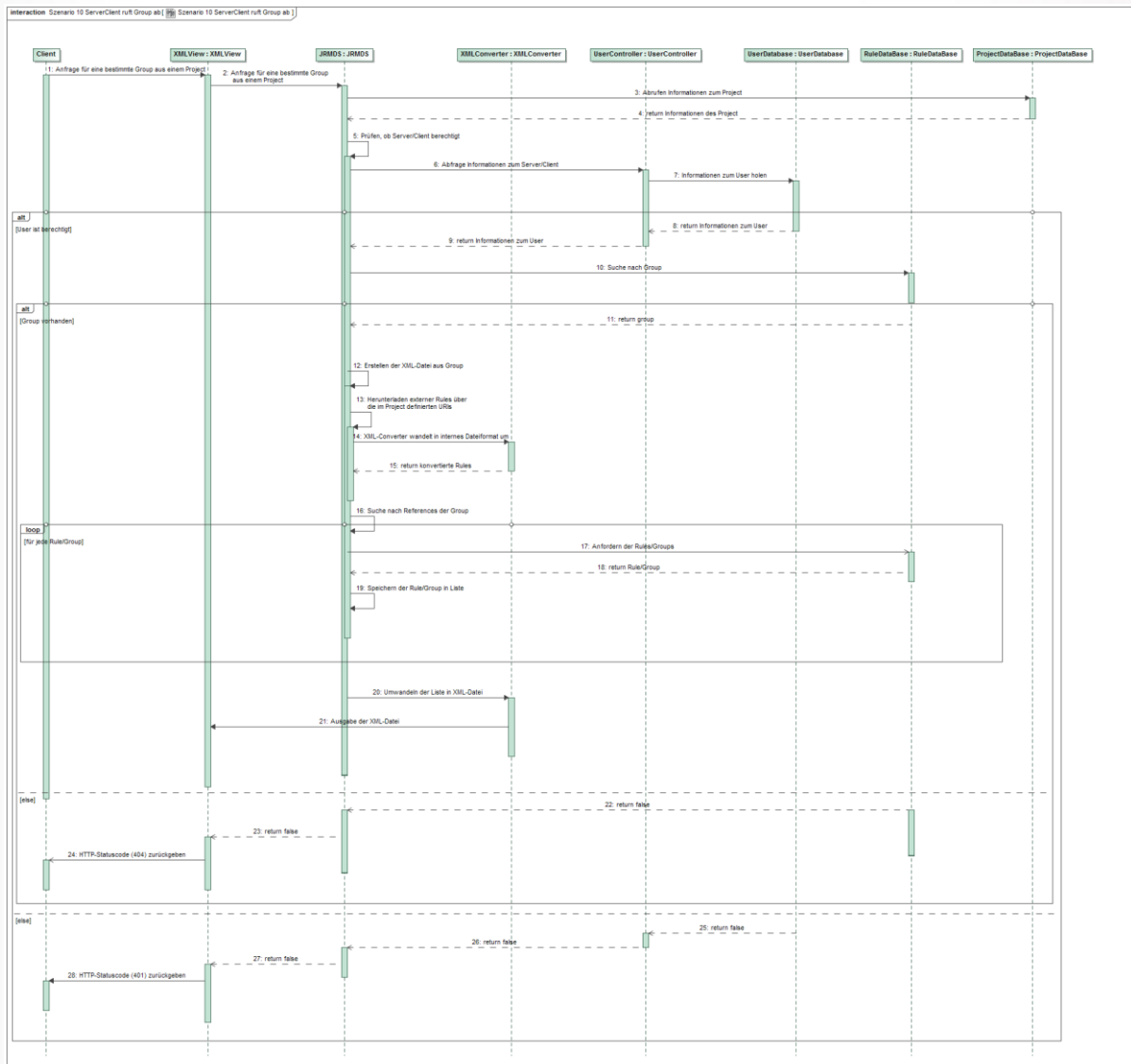
Wenn der Client für den Zugriff auf dieses Project berechtigt ist, wird die betreffende Group gesucht.

Existiert die Gruppe nicht, wird ein HTTP-Status 404 zurückgegeben.

Existiert die Group, beginnt das JRMDs mit der Generierung des XML-Codes. Dazu werden zuerst vom JRMDs alle externen Rules über die im Project definierten URIs heruntergeladen und vom XML-Converter in das interne Datenformat gewandelt. Das JRMDs sucht nun für die spezifizierte Group alle definierten References und speichert gefundene Groups oder Rules in einer listenförmigen Datenstruktur. Enthalten die gefundenen Rules/Groups weitere References, werden diese ebenfalls aufgelöst.

Werden für eine Reference Rules/Groups jeweils external wie auch internal gefunden, so überschreiben sich Rules/Groups mit gleicher ID zuerst in der Reihenfolge der spezifizierten URIs und schließlich die internal Rules/Groups.

Das JRMDs sendet diese Liste von Groups und Rules zum XML-Converter. Die resultierende XML-Datei bekommt der Client dann zum Download angeboten.

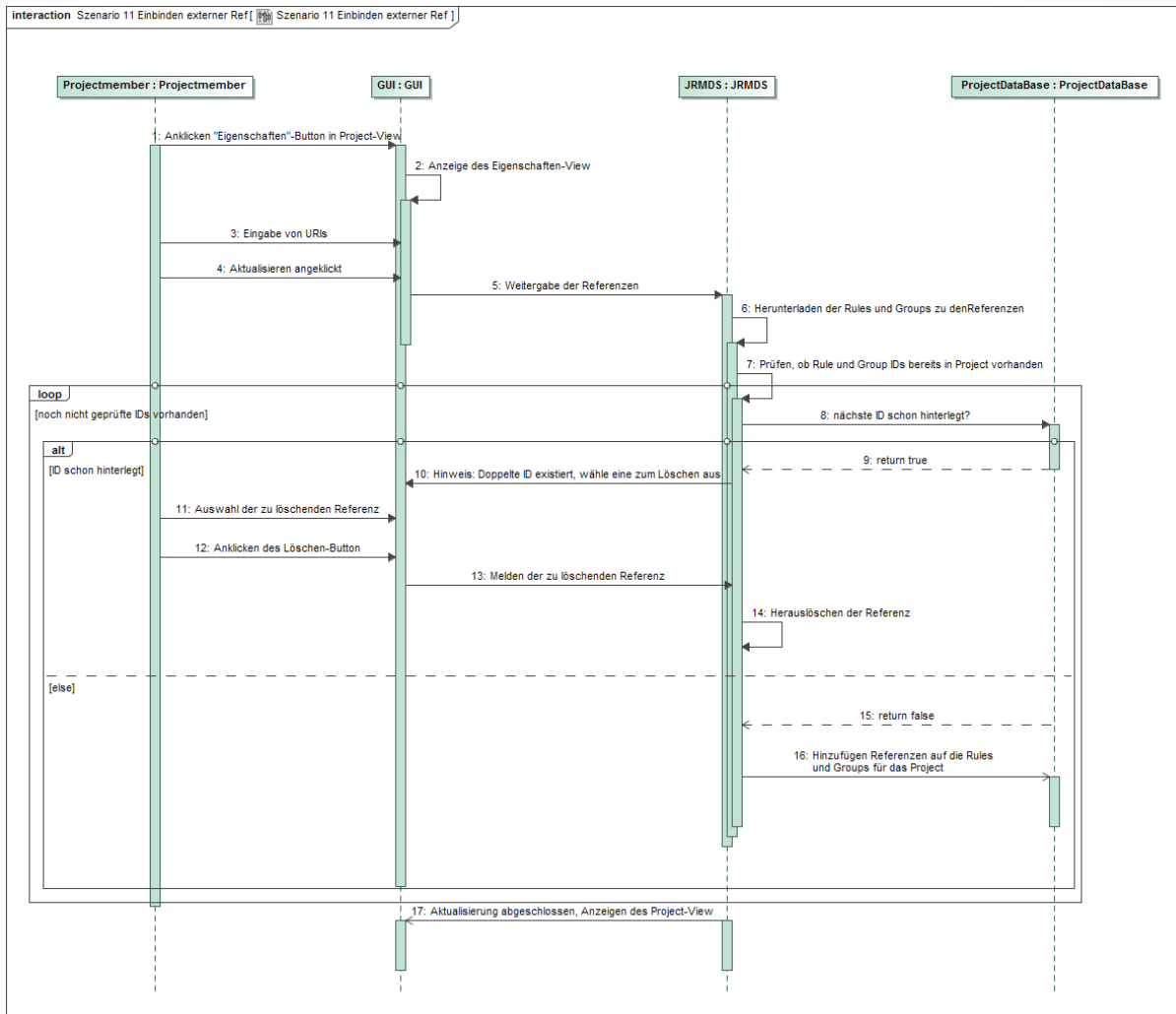


### Einbinden externer Referenzen

Ein Projectmember will das Project um external Rules aus einem Repository erweitern. Dazu öffnet er die Eigenschaftsseite von dem Project. Dort kann er in einer Liste URIs eintragen, die entsprechende Ressourcen angeben. Die Eingabe von optionalen Benutzernamen und Passwort erfolgt innerhalb der URI.

Wenn das Projectmember auf Aktualisieren klickt, dann werden von dem JRMDs die Rules und Groups heruntergeladen, aber nicht in der Datenbank gespeichert. Es wird nun geprüft, ob Rules und Groups, die bereits innerhalb des Projects spezifiziert wurden, auch in dem Repository existieren. Wenn dies der Fall ist, werden Rules oder Groups mit gleicher ID im external Repository ignoriert und in dem View wird ein Hinweis angezeigt.

Wenn der Fall eintritt, dass eine auf diese Art überschriebene Rule von einer anderen Rule in dem external Repository als Reference angegeben wurde (wodurch die external Rule nun auf die internal Rule eine Reference hat), dann wird zum einen darauf ebenfalls hingewiesen und es wird zusätzlich geprüft, ob es nicht auch einen Zyklus gibt (external Reference zeigt auf internal Rule, internal Rule hat aber die externe Rule ebenfalls als Reference definiert). In diesem Fall muss der Benutzer entscheiden, welche der beiden References gelöscht werden soll.





### Passwort vergessen

Befindet sich der User auf der Hauptseite und klickt er auf "Passwort vergessen" im Loginbereich, so öffnet sich ein neuer View. In diesem muss der User seine e-Mailadresse eingeben.

Das JRMDS prüft in der User-Datenbank, ob diese e-Mailadresse registriert ist.

1.) Liegt kein Usereintrag vor, der diese e-Mailadresse hinterlegt hat, wird "Es existiert kein User mit dieser e-Mailadresse" ausgegeben.

2.) Liegt ein Usereintrag in der User-Datenbank mit dieser e-Mailadresse vor, so wird an die Adresse eine e-Mail mit einem Link zum Passwort-Neusetzen geschickt. Wenn der User diese e-Mail öffnet und auf den Link klickt, öffnet sich ein Passwort-Neusetzen-View. Dort muss der User doppelt sein gewünschtes, neues Passwort eintragen.

Nach Anklicken des Bestätigen-Buttons prüft JRMDS ob die beiden Passwörter übereinstimmen und gewisse Voraussetzungen erfüllen.

Stimmen die Passwörter nicht überein, wird "Die Passwörter stimmen nicht überein" ausgegeben.

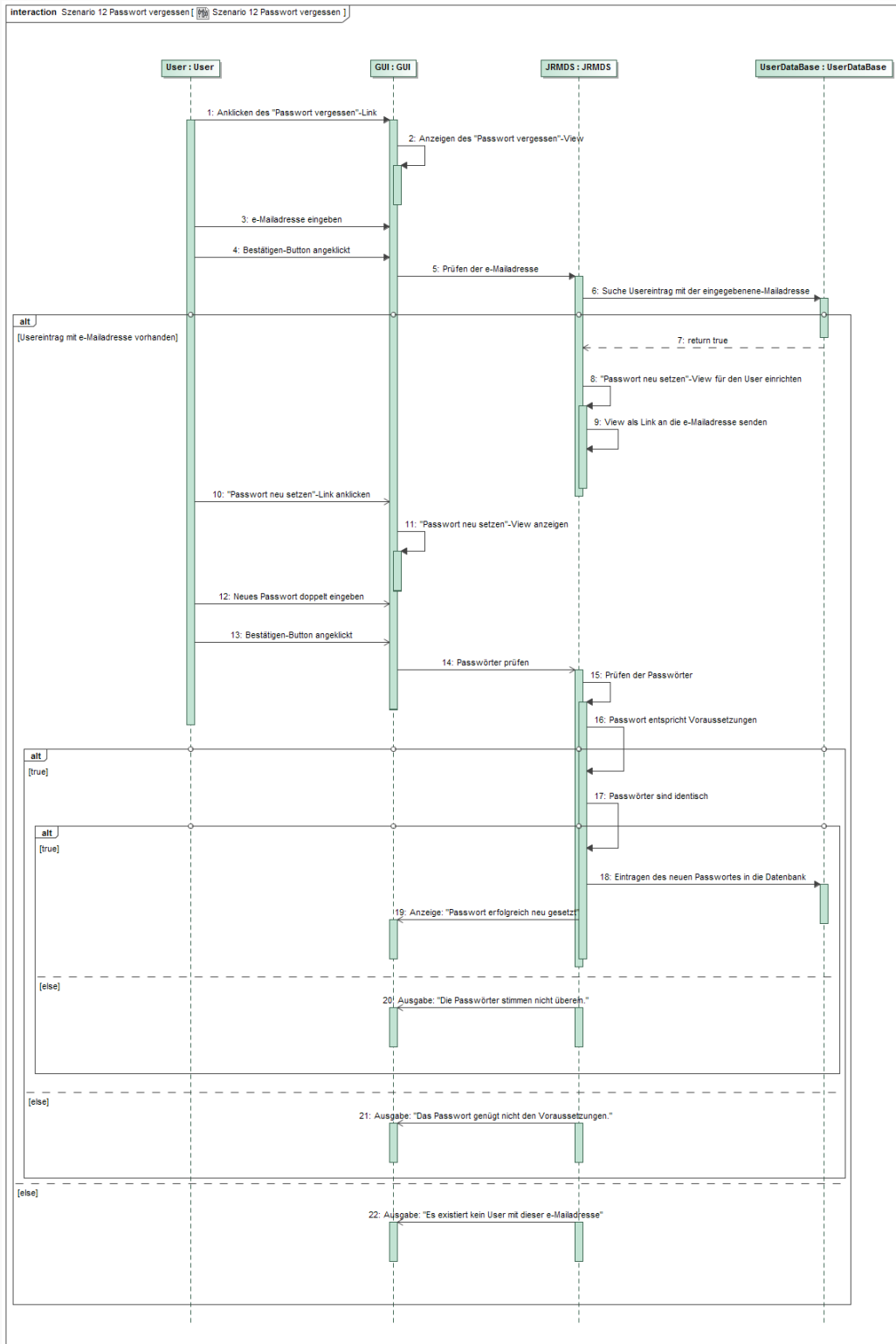
Genügen die Passwörter nicht den Voraussetzungen, wird "Die Passwörter genügen nicht den Voraussetzungen" ausgegeben.

Wurde das Passwort korrekt gewählt und richtig in beide Felder eingetragen, so sendet JRMDS dieses an die User-Datenbank als neues Passwort für den User.

Dem User wird eine Nachricht ausgegeben: "Passwort erfolgreich neu gesetzt."

Der User kann jederzeit auf dem Passwort-Vergessen-View sowie auf dem Passwort-Neusetzen-View auf Abbrechen klicken und seine bisherigen Einträge verwerfen.

Er wird dann direkt auf die Hauptseite zurückgebracht.

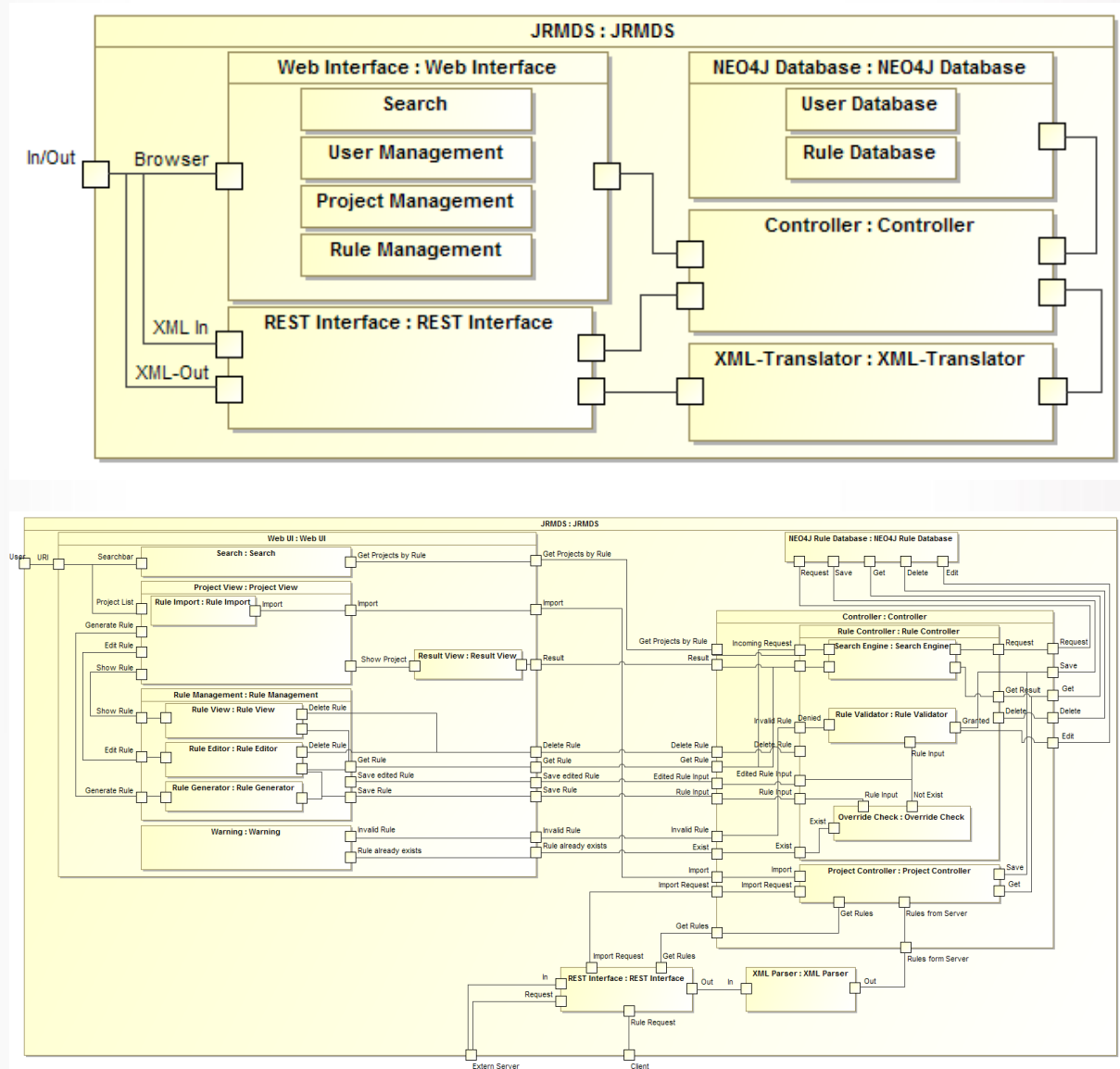


## Benutzer ohne Berechtigungszuweisung

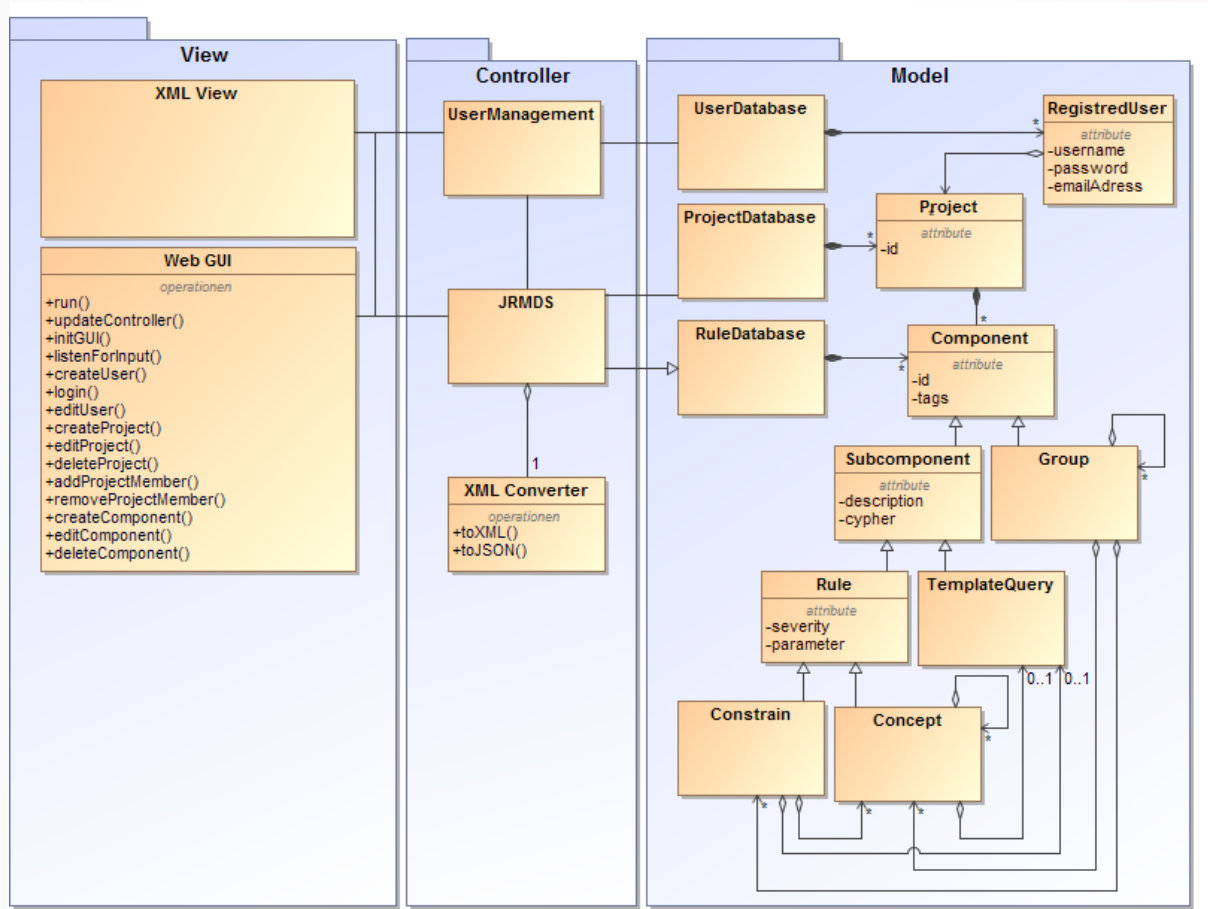
## Nach erfolgreicher Registrierung und Emailvalidierung gespeicherter Benutzer

Benutzer mit voller Berechtigung

## 4.2. Top-Level-Architektur



### 4.3. Analysediagramm

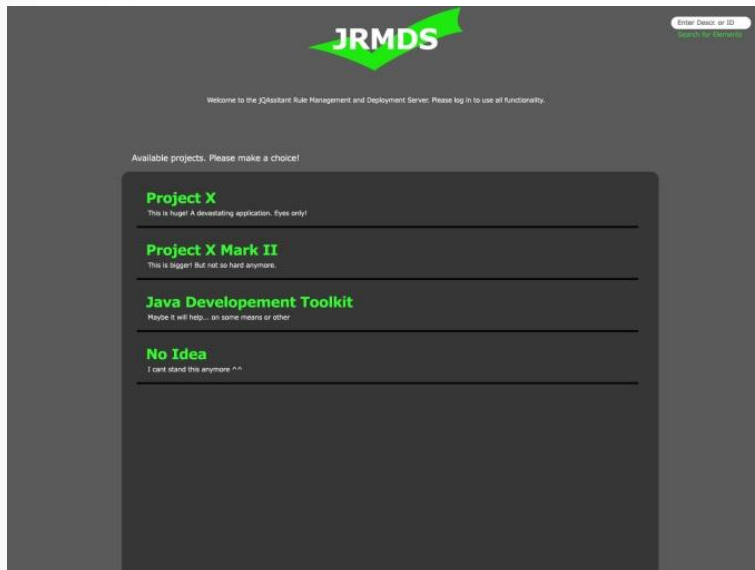


### 4.4. Komponentendiagramm

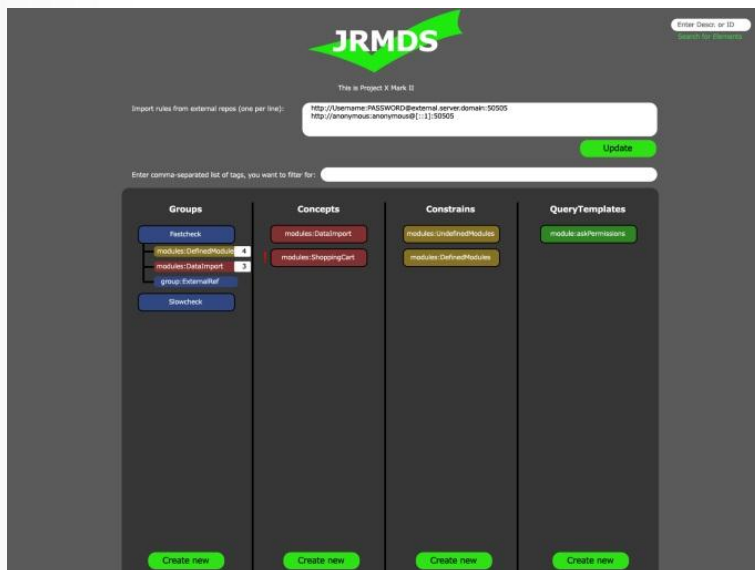
## 5. GUI Prototyp

(wird derzeit überarbeitet)

### Startbildschirm



### Projektverwaltung



## Einstellungen editieren

The screenshot shows the JRMDs settings interface. At the top, there's a header with the JRMDs logo and a user profile icon. Below the header, there's a section for 'This is Project A, Rule B' with a text input for 'Import rules from external source (one per line):' containing two URLs and an 'Update' button. Below this is a search bar with the text 'Enter comma-separated list of tags, you want to filter for:'. The main content area is divided into four tabs: 'Groups', 'Concepts', 'Constraints', and 'QueryTemplates'. The 'Constraints' tab is active, showing a list of constraints. A modal window is open over the 'Constraints' tab, titled 'Edit Constraint'. It contains fields for 'ID:', 'Severity', 'Description', 'Ctype', 'Parameters', 'Name:', 'Value:', and 'Type:'. The 'Type' field has radio buttons for 'Int', 'String', and 'Bool', with 'Bool' selected. A 'Save' button is at the bottom of the modal.

## Suche

The screenshot shows the JRMDs search results page. At the top, there's a header with the JRMDs logo and a user profile icon. Below the header, there's a search bar with the text 'Search results for: foobar'. Below the search bar, there's a message: 'Click on a rule to open the corresponding project. The rule will be highlighted, to perform further actions.' The main content area is divided into four tabs: 'Groups', 'Concepts', 'Constraints', and 'QueryTemplates'. The 'Constraints' tab is active, showing a list of constraints. The 'Groups' tab is also visible, showing a 'Stackcheck' button. The 'Concepts' tab shows 'modules:DataImport' and 'modules:Preparation'. The 'Constraints' tab shows 'modules:UndefinedModules' and 'modules:DefinedModules'. The 'QueryTemplates' tab shows 'module:askPermissions'.

## 6. Entwurfsdatenmodell

## 7. Testfälle

## 8. Glossar

**JRMDS** - diese Anwendung

**Rules** - Concepts oder Constraints

**Group** - Eine Sammlung von mehreren Rules und oder Groups

**Reference** - eine Beziehung zwischen Groups und Rules

**Project** - Vereinigung von Groups und Rules, es kann nur References innerhalb eines Projects geben

**Projectmember** - Ein User, der Zugriff auf ein bestimmtes Project hat

**User** - allgemeiner User ohne bestimmte Zugriffsrechte

**registrierter User** - User nach erfolgreicher Registrierung und Emailvalidierung

**internal rule/group** - innerhalb eines Projects definiert

**external rule/group** - durch URI spezifizierte externe Repositorys

**View** - eine Seite, bzw. Fenster innerhalb der Webanwendung

**Profil** - Seite, in der Benutzername, Emailadresse und Zugehörigkeiten im Project des registrierten Users zu finden sind

**Username** - Der selbstvergebene eindeutige Name, der bei der Registrierung vergeben wurde.