

# Anwendungsfallbeschreibung

# autoPSI

## Inhalt

Inhalt.....	2
Definitionen.....	5
Termin/ Termincontainer.....	5
Objekte suchen.....	5
Objekte herunterladen.....	5
Shared-Objekte.....	5
Aktorenliste.....	6
Aktorenhierarchie.....	6
Übersicht über alle Anwendungsfälle.....	6
Objekte suchen und herunterladen.....	7
Anwendungsfalldiagramm.....	7
Anwendungsfallbeschreibung.....	8
E/A-Diagramm.....	8
Shared Objekte auswählen.....	10
Anwendungsfalldiagramm.....	10
Anwendungsfallbeschreibung.....	10
E/A-Diagramm.....	11
Server starten.....	13
Anwendungsfalldiagramm.....	13
Anwendungsfallbeschreibung.....	13
E/A-Diagramm.....	14
Server stoppen.....	15
Anwendungsfalldiagramm.....	15
.....	15
Anwendungsfallbeschreibung.....	15
E/A-Diagramm.....	16
Termincontainer anlegen.....	17
Anwendungsfalldiagramm.....	17
Anwendungsfallbeschreibung.....	17
E/A-Diagramm.....	18
Termin anlegen.....	19
Anwendungsfalldiagramm.....	19
Anwendungsfallbeschreibung.....	19
E/A-Diagramm.....	20
Kontakt anlegen.....	22
Anwendungsfalldiagramm.....	22
Anwendungsfallbeschreibung.....	22
E/A-Diagramm.....	23
LVA anlegen.....	24
Anwendungsfalldiagramm.....	24
Anwendungsfallbeschreibung.....	24
E/A-Diagramm.....	25
Lehrmittel anlegen.....	26
Anwendungsfalldiagramm.....	26
Anwendungsfallbeschreibung.....	26
E/A-Diagramm.....	27

## autoPSI::Anwendungsfallbeschreibung

Prüfung anlegen.....	29
Anwendungsfalldiagramm.....	29
Anwendungsfallbeschreibung.....	29
E/A-Diagramm.....	30
Termincontainer bearbeiten.....	31
Anwendungsfalldiagramm.....	31
Anwendungsfallbeschreibung.....	31
E/A-Diagramm.....	32
Termin bearbeiten.....	33
Anwendungsfalldiagramm.....	33
Anwendungsfallbeschreibung.....	33
E/A-Diagramm.....	34
LVA bearbeiten.....	35
Anwendungsfalldiagramm.....	35
Anwendungsfallbeschreibung.....	35
E/A-Diagramm.....	36
Kontakt bearbeiten.....	37
Anwendungsfalldiagramm.....	37
Anwendungsfallbeschreibung.....	37
E/A-Diagramm.....	38
Lehrmittel bearbeiten.....	39
Anwendungsfalldiagramm.....	39
Anwendungsfallbeschreibung.....	39
E/A-Diagramm.....	40
Prüfung bearbeiten.....	42
Anwendungsfalldiagramm.....	42
Anwendungsfallbeschreibung.....	42
E/A-Diagramm.....	43
Termin löschen.....	45
Anwendungsfalldiagramm.....	45
Anwendungsfallbeschreibung.....	45
E/A-Diagramm.....	46
Termincontainer löschen.....	47
Anwendungsfalldiagramm.....	47
Anwendungsfallbeschreibung.....	47
E/A-Diagramm.....	48
Kontakt löschen.....	49
Anwendungsfalldiagramm.....	49
Anwendungsfallbeschreibung.....	49
E/A-Diagramm.....	50
LVA löschen.....	51
Anwendungsfalldiagramm.....	51
Anwendungsfallbeschreibung.....	51
E/A-Diagramm.....	52
Lehrmittel löschen.....	53
Anwendungsfalldiagramm.....	53
Anwendungsfallbeschreibung.....	53
E/A-Diagramm.....	54

## autoPSI::Anwendungsfallbeschreibung

Prüfung löschen.....	55
Anwendungsfalldiagramm.....	55
Anwendungsfallbeschreibung.....	55
E/A-Diagramm.....	56
Objekt an Termincontainer anhängen.....	57
Anwendungsfalldiagramm.....	57
Anwendungsfallbeschreibung.....	57
E/A-Diagramm.....	58
Objekt an Termin anhängen.....	59
Anwendungsfalldiagramm.....	59
Anwendungsfallbeschreibung.....	59
E/A-Diagramm.....	60
Server starten.....	61
Anwendungsfalldiagramm.....	61
Anwendungsfallbeschreibung.....	61
E/A-Diagramm.....	62

## Definitionen

### ***Termin/ Termincontainer***

Ein Termincontainer beinhaltet einzelne Termine. Dadurch werden die dem Termincontainer zugeordneten Termine zu einem Gesamt ereignis verbunden.

### ***Objekte suchen***

Wird ein Objekt gesucht, so ist damit, so nicht ausdrücklich anders erwähnt, die Suche im JavaSpace gemeint.

### ***Objekte herunterladen***

Ein Objekt wird aus dem JavaSpace heruntergeladen.

### ***Shared-Objekte***

Objekte, die anderen Studntn über den JavaSpace zum Download zur Verfügung gestellt werden.

## Aktorenliste

<i>Name</i>	<i>Rechte</i>	<i>Anmerkungen</i>
Benutzer	Space-Zugriff, Offline-Aktivitäten	Der Benutzer kann Objekte im Space suchen, diese herunterladen und in die offline-Datenbank integrieren. Weiters kann er die offline-Datenbank mittels dafür geeigneter Benutzeroberfläche verwalten.
Server-Administrator	Server-Betreuung	Starten des Servers und Erstellen des JavaSpaces

## Aktorenhierarchie

Server-Administrator ist eine Spezialisierung des Benutzers.

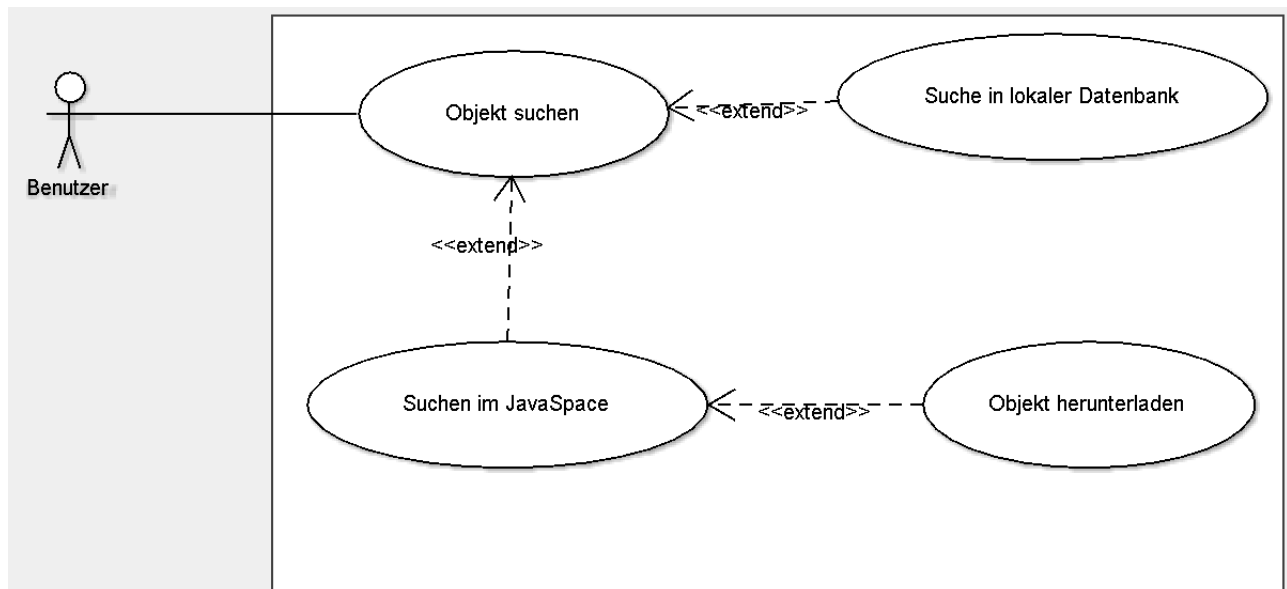
## Übersicht über alle Anwendungsfälle

<i>Paket</i>	<i>Kürzel</i>	<i>Titel</i>	<i>Priorität</i>
JavaSpace		Objekte Suchen und herunterladen	Hoch [fertig]
JavaSpace		Shared-Objekte auswählen	Hoch [fertig]
hsqldb		Termincontainer anlegen	Hoch felly
hsqldb		Termin anlegen	Hoch felly
hsqldb		LVA anlegen	Hoch felly
hsqldb		Kontakt anlegen	Hoch felly
hsqldb		Lehrmittel anlegen	Hoch felly
hsqldb		Prüfung anlegen	Hoch felly
hsqldb		Termincontainer bearbeiten	Mittel stephe
hsqldb		Termin bearbeiten	Mittel stephe
hsqldb		LVA bearbeiten	Mittel stephe
hsqldb		Kontakt bearbeiten	Mittel stephe
hsqldb		Lehrmittel bearbeiten	Mittel stephe
hsqldb		Prüfung bearbeiten	Mittel stephe
hsqldb		Termincontainer	Niedrig rudi

<i>Paket</i>	<i>Kürzel</i>	<i>Titel</i>	<i>Priorität</i>
		löschen	
hsqldb		Termin löschen	Niedrig rudi
hsqldb		LVA löschen	Niedrig rudi
hsqldb		Kontakt löschen	Niedrig rudi
hsqldb		Lehrmittel löschen	Niedrig rudi
hsqldb		Prüfung löschen	Niedrig rudi
hsqldb		Objekt an Termincontainer anhängen	Mittel [fertig]
hsqldb		Objekt an Termin anhängen	Mittel [fertig]
Server		Server starten	Hoch [fertig]
Server		Server stoppen	Mittel [fertig]

## Objekte suchen und herunterladen

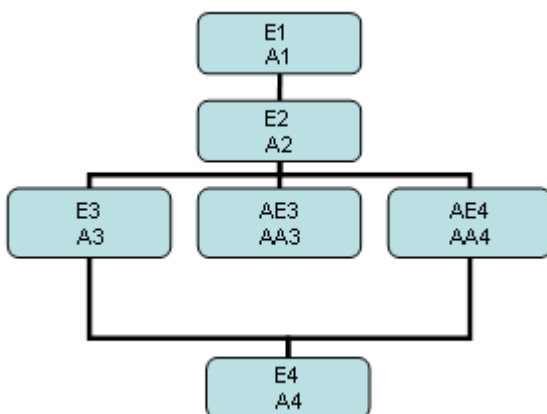
### Anwendungsfalldiagramm



## Anwendungsfallbeschreibung

<b>Identification Summary</b>	<b>Titel</b>	Objekte suchen und herunterladen
	<b>Kurzbeschreibung</b>	Objekte lokal suchen oder im Space suchen und herunterladen
	<b>Beteiligte Akteure</b>	User
	<b>Erstellungsdaten</b>	Alpay Firato, 15.05.2006, V 1.0
<b>Flow of Events</b>	<b>Hauptszenario</b>	Das Suchfenster wird geöffnet, Suchparameter werden eingegeben. Dann wird der Suchraum ausgewählt und die Suche gestartet. Gefundene Objekte aus dem Space können anschließend heruntergeladen werden.
	<b>Alternative Szenarien</b>	Die Suche bringt keine Ergebnisse.
	<b>Fehlersituationen</b>	Verbindung zum Space oder zur Datenbank kann nicht hergestellt werden. Gefundene Objekte können nicht heruntergeladen werden
	<b>Vorbedingung</b>	Verbindung zum Space bzw. zur Datenbank hergestellt.
	<b>Nachbedingung</b>	Objekte wurde gefunden und auf Wunsch in die lokale Datenbank gespeichert.
<b>Non-functional-Constraints</b>	<b>Non-Functional-Constraints</b>	Gefundene Objekte im Space werden nicht automatisch heruntergeladen.

## E/A-Diagramm





## autoPSI::Anwendungsfallbeschreibung

E1) User klickt auf „Suche“ im Hauptmenü

A1) Suchfenster wird geöffnet

E2) User gibt Suchparameter ein, wählt Suchraum und startet Suche

A2) Das System sucht im Suchraum nach entsprechenden Objekte.

E3) Das System findet passende Objekte.

A3) Die gefundenen Objekte werden angezeigt.

AE3) Das System findet keine passenden Objekte.

AA3) Eine Fehlermeldung wird angezeigt.

AE4) User bricht die Suche ab.

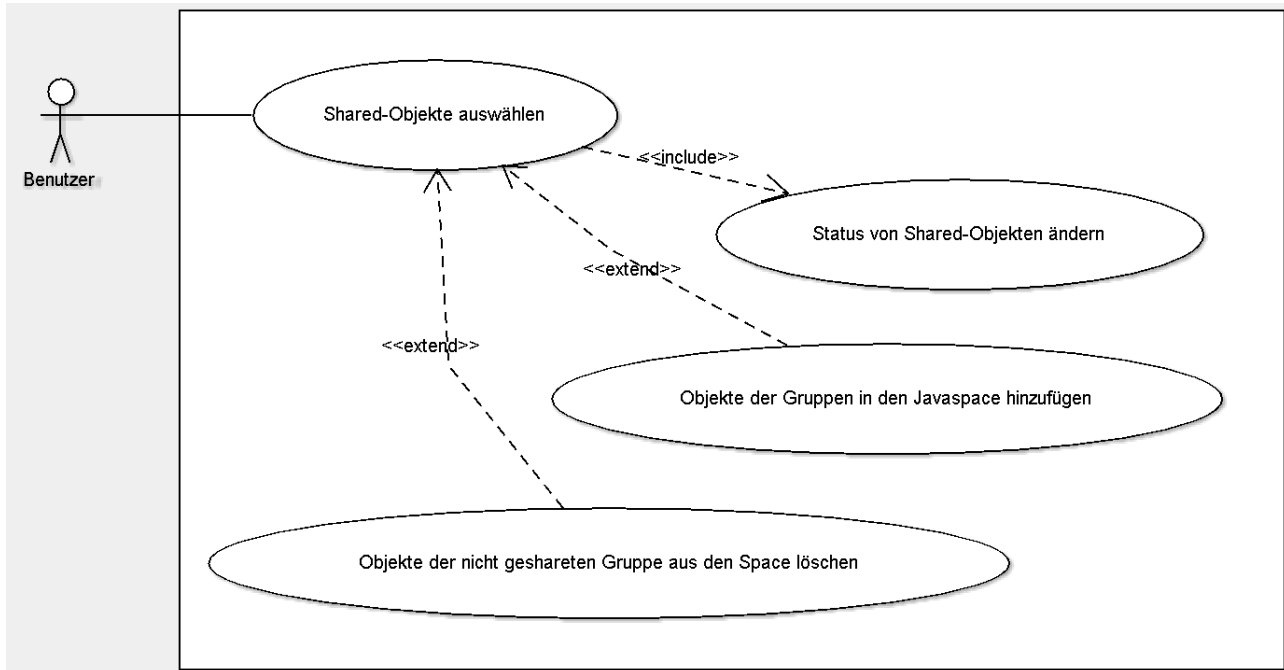
AA4) Die bis zum Abbruch gefundenen Objekte werden angezeigt und die Suche beendet.

E4) User wählt die zu herunterladenden Objekte aus und bestätigt die Auswahl.

A4) Objekte werden heruntergeladen und vom System in die lokale Datenbank gespeichert.

## Shared Objekte auswählen

### Anwendungsfalldiagramm

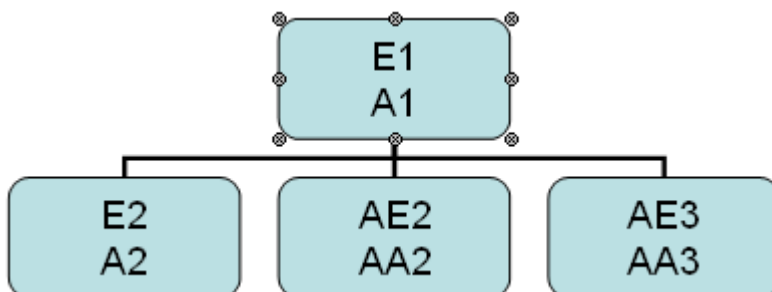


### Anwendungsfallbeschreibung

<b>Identifikation Summary</b>	<b>Titel</b>	Shared Objekte auswählen
	<b>Kurzbeschreibung</b>	Auswählen der Objekte, die in den Space gestellt werden sollen
	<b>Beteiligte Akteure</b>	Benutzer
	<b>Erstellungsdaten</b>	Alpay Firato, 15.05.2006, V 1.0

<b>Flow of Events</b>	<b>Hauptszenario</b>	In einem Fenster kann der Benutzer aus einer Liste auswählen welche Objektgruppen geshared werden sollen. Bei der nächsten Verbindung mit dem Space werden die ausgewählten Objekte in den Space geladen. Abgewählte Objekte werden aus dem Space gelöscht. Der Objektgruppenstatus wird in die Datenbank gespeichert.
	<b>Alternative Szenarien</b>	Der User bricht die Aktion ohne Änderung ab.
	<b>Fehlersituationen</b>	Die Verbindung zur Datenbank besteht nicht.
	<b>Vorbedingung</b>	Objekte sind vorhanden.
	<b>Nachbedingung</b>	Bei der nächsten Verbindung zum Space werden die ausgewählten Objekte in den Space hochgeladen.
<b>Non-functional</b>	<b>Non-Functional-Constraints</b>	.

### E/A-Diagramm



E1) Der Benutzer wählt aus einer Liste die zu sharenden Objektgruppen aus.

A1) Das System versucht die ausgewählten Objektgruppen in den Space zu stellen.

autoPSI::Anwendungsfallbeschreibung

E2) Die Gruppen wurden erfolgreich in den Space gestellt.

A2) Eine Bestätigungsmeldung wird ausgegeben.

AE2) Die Änderungen können nicht in die Datenbank gespeichert werden.

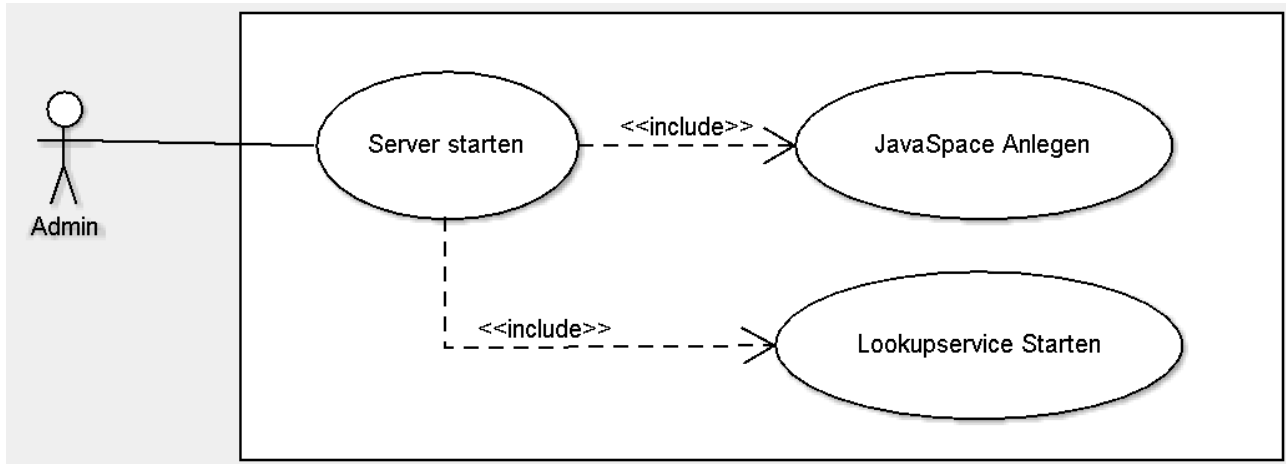
AA2) Die Objekte werden nicht in den Space gestellt, außerdem wird eine Fehlermeldung ausgegeben.

AE3) Die Objektgruppen können nicht in den Space gestellt werden.

AA3) Ein Hinweisse wird ausgegeben. Bei der nächsten Verbindung zum Space wird das Hochladen erneut versucht.

## Server starten

### Anwendungsfalldiagramm

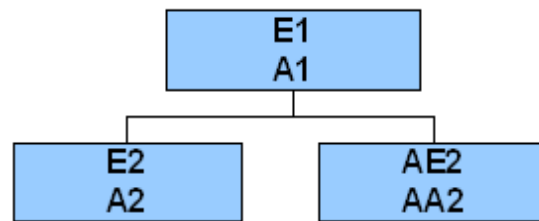


### Anwendungsfallbeschreibung

<b>Identification Summary</b>	<b>Titel</b>	Server starten
	<b>Kurzbeschreibung</b>	Server wird gestartet, JavaSpace wird erstellt und initialisiert, und anschließend beim Lookup-Service registriert.
	<b>Beteiligte Aktoren</b>	Server-Administrator
	<b>Erstellungsdaten</b>	Alpay Firato, 15.05.2006, V 1.0
<b>Flow of Events</b>	<b>Hauptszenario</b>	Der Server-Administrator startet den Serverprozess. Ein JavaSpace wird angelegt und beim Lookup-Service registriert. Es wird eine Erfolgsmeldung ausgegeben.
	<b>Alternative Szenarien</b>	
	<b>Fehlersituationen</b>	Falls der Server nicht gestartet bzw. der JavaSpace nicht erstellt werden kann oder beim Lookup-Service nicht registriert werden kann, wird eine Fehlermeldung ausgegeben.
	<b>Vorbedingung</b>	Das System läuft
	<b>Nachbedingung</b>	Der Server ist gestartet, Clients können über den Lookup-Service auf den Space zugreifen und ihn benutzen.

<b>Non-functional-Constraints</b>	<b>Non-Functional-Constraints</b>	Im Normalfall wird der Server nicht so oft neu gestartet.
-----------------------------------	-----------------------------------	---

### ***E/A-Diagramm***



E1) Der Server Administrator startet den Server.

A1) Das System versucht einen Javaspaces anzulegen und ihn beim Lookup-Service zu registrieren.

E2) Javaspaces wurde angelegt und der Lookup-Service wurde registriert.

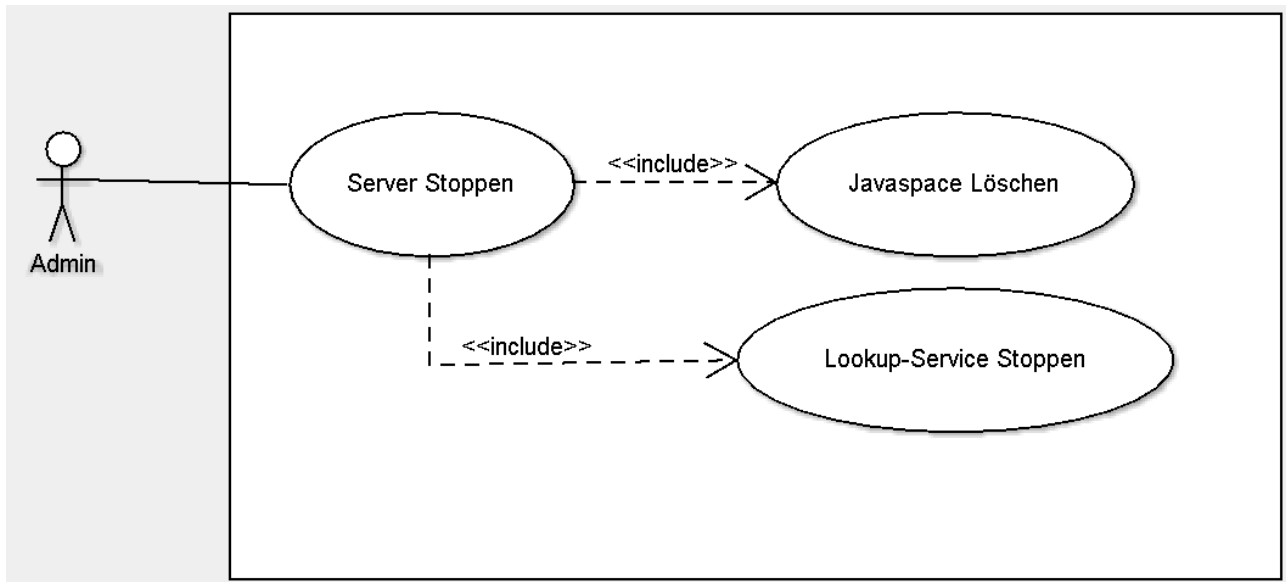
A2) Eine Bestätigungsmeldung wird ausgegeben.

AE2) Der Javaspaces konnte nicht angelegt werden oder beim Registrieren des Spaces beim Lookup-Service war nicht erfolgreich.

AA2) Das System gibt eine Fehlermeldung aus.

## Server stoppen

### Anwendungsfalldiagramm

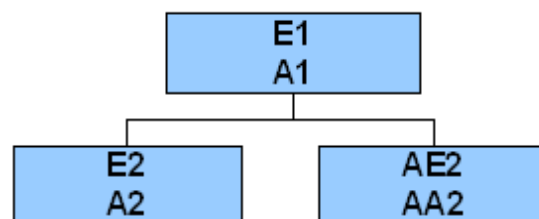


### Anwendungsfallbeschreibung

Identification Summary	<b>Titel</b>	Server stppen
	<b>Kurzbeschreibung</b>	Server wird gestppt, JavaSpace wird gelöscht und der Lookup-Service wird de registriert.
	<b>Beteiligte Aktoren</b>	Server-Administrator
	<b>Erstellungsdaten</b>	Alpay Firato, 15.05.2006, V 1.0

<b>Flow of Events</b>	<b>Hauptszenario</b>	Der Server-Administrator stoppt den Server. Der JavaSpace wird gelöscht und anschließend der Lookup-Service deregistriert. Es wird eine Erfolgsmeldung ausgegeben.
	<b>Alternative Szenarien</b>	
	<b>Fehlersituationen</b>	Falls der JavaSpace nicht gelöscht werden kann, wird eine Fehlermeldung ausgegeben.
	<b>Vorbedingung</b>	Das System läuft
	<b>Nachbedingung</b>	Der Server ist gestoppt..
<b>Non-functional-Constraints</b>	<b>Non-Functional-Constraints</b>	Im Normalfall wird der Server nicht so oft gestoppt.

## E/A-Diagramm



E1) Der Server Administrator stoppt den Server.

A1) Das System versucht den beim Starten erzeugten Javaspaces zu löschen und den Lookup-Service zu deregistrieren.

E2) Javaspaces wurden gelöscht und der Lookup-Service wurde deregistriert.

A2) Eine Bestätigungsmeldung wird ausgegeben.

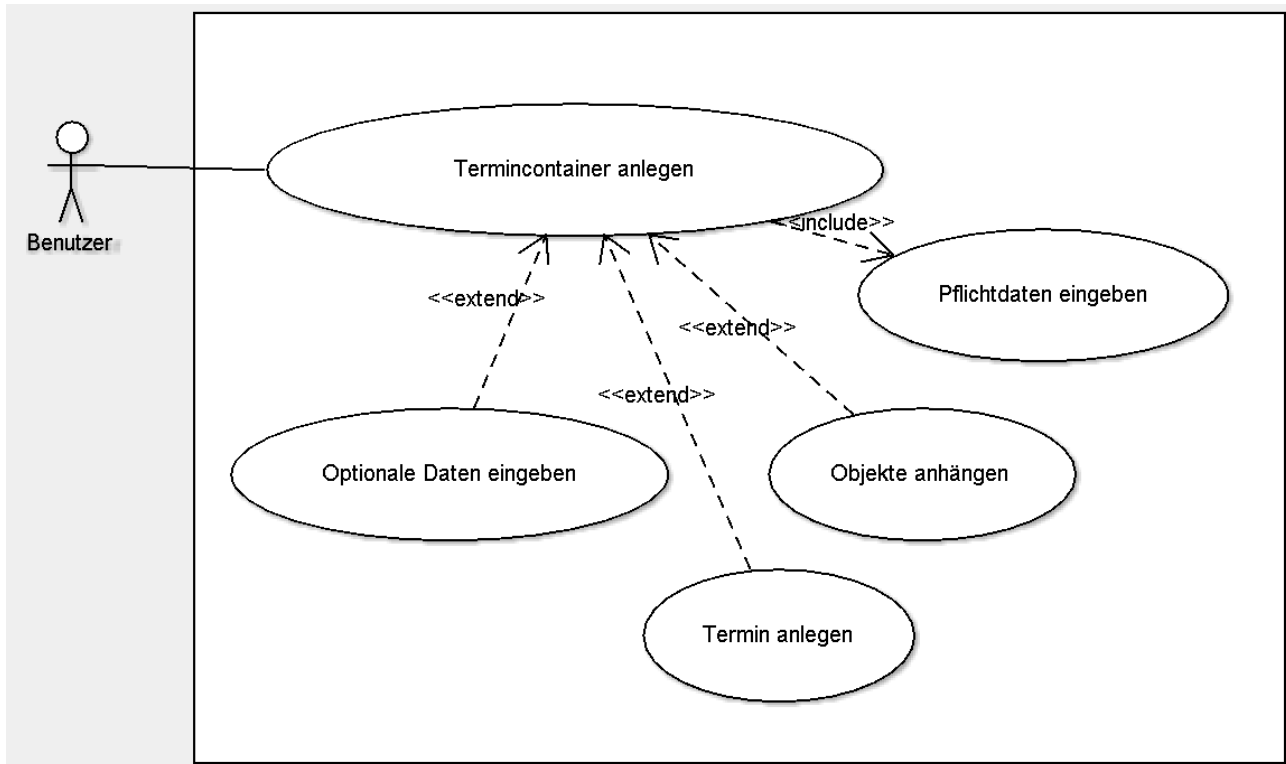
AE2) Der Javaspaces konnte nicht gelöscht oder konnte vom Lookup-Service nicht deregistriert werden.

AA2) Das System gibt eine Fehlermeldung aus.



## Termincontainer anlegen

### Anwendungsfalldiagramm

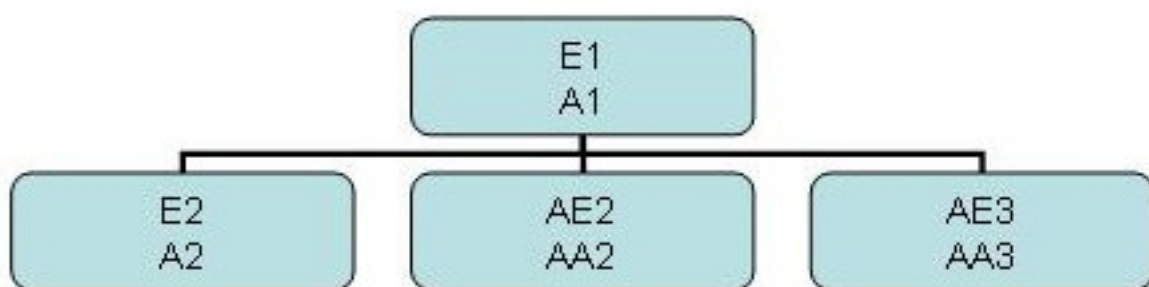


### Anwendungsfallbeschreibung

Summary on Identificati	<b>Titel</b>	Termincontainer anlegen
	<b>Kurzbeschreibung</b>	Ein neuer Termincontainer wird hinzugefügt.
	<b>Beteiligte Akteure</b>	Benutzer
	<b>Erstellungsdaten</b>	Daniel Feledi, 15.05.2006, V 1.0

<b>Flow of Events</b>	<b>Hauptszenario</b>	Ein neuer Termincontainer wird in die lokale Datenbank eingefügt, nachdem die Daten für den neuen Termincontainer eingegeben wurden.
	<b>Alternative Szenarien</b>	Eingabe wurde vom Benutzer abgebrochen.
	<b>Fehlersituationen</b>	Falls der Termincontainer nicht in die Datenbank eingefügt werden kann, wird eine Fehlermeldung ausgegeben. Ungültige Eingabe durch Benutzer; eine Fehlermeldung / Warnhinweis wird angezeigt, das Bearbeiten-Fenster wird nicht geschlossen und die Daten werden (noch) nicht in die Datenbank geschrieben.
	<b>Vorbedingung</b>	Das System läuft, Programm gestartet
	<b>Nachbedingung</b>	Der neue Termincontainer wurde in der Datenbank mit den eingegebenen Daten gespeichert.
<b>Non-functional-Constraints</b>	<b>Non-Functional-Constraints</b>	Dieser Anwendungsfall wird oft aufgerufen werden. Termincontainer können nicht mit anderen Termincontainer verbunden werden.

### E/A-Diagramm



E1) Benutzer öffnet das „Datensatz anlegen“ Fenster im Hauptmenü

A1) Das Fenster wird geöffnet

E2) Benutzer wählt „Termincontainer“, gibt die Daten ein und bestätigt die Eingabe

A2) Die Daten werden in der Datenbank gespeichert

AE2) Benutzer wählt „Termincontainer“, gibt Daten ein, fügt Objekt(e) an Termincontainer und bestätigt die Eingabe

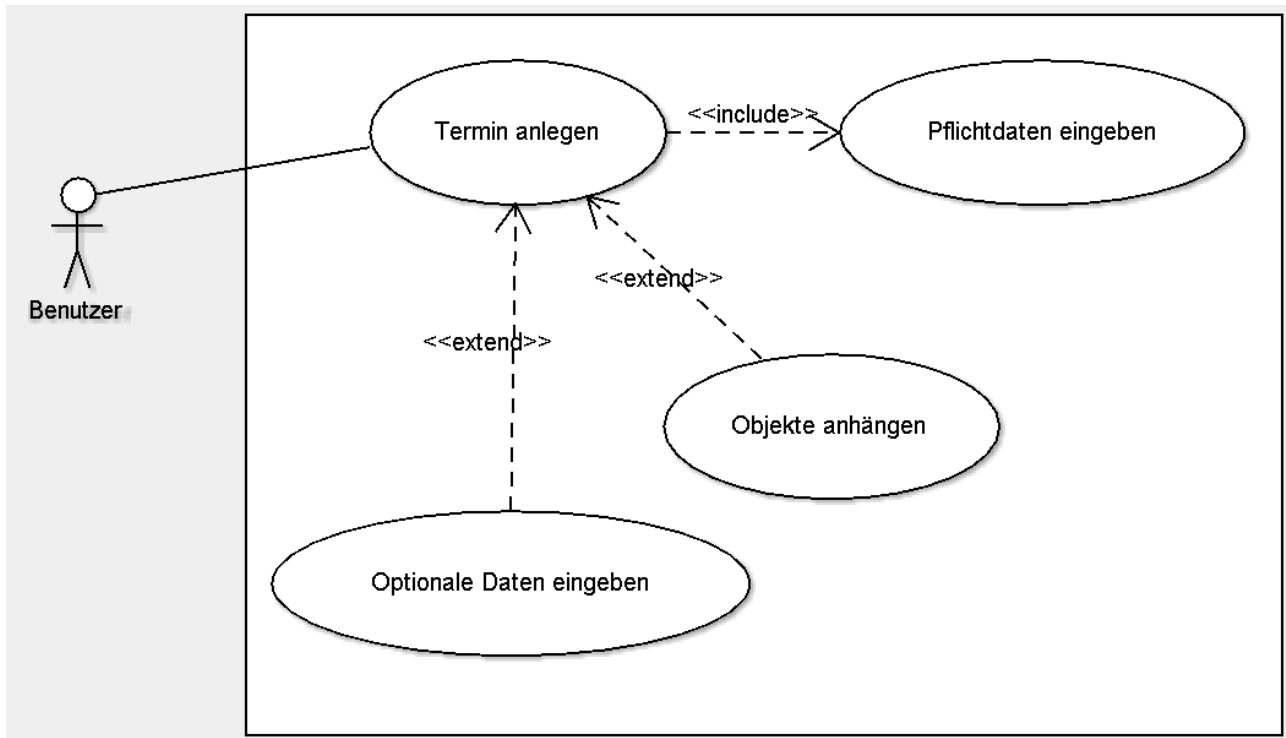
AA2) Daten werden in der Datenbank gespeichert und das Objekt an den Termincontainer gehängt

AE3) Der Benutzer bricht Aktion ab

AA3) Es werden keine Daten in die Datenbank geschrieben

## Termin anlegen

### Anwendungsfalldiagramm

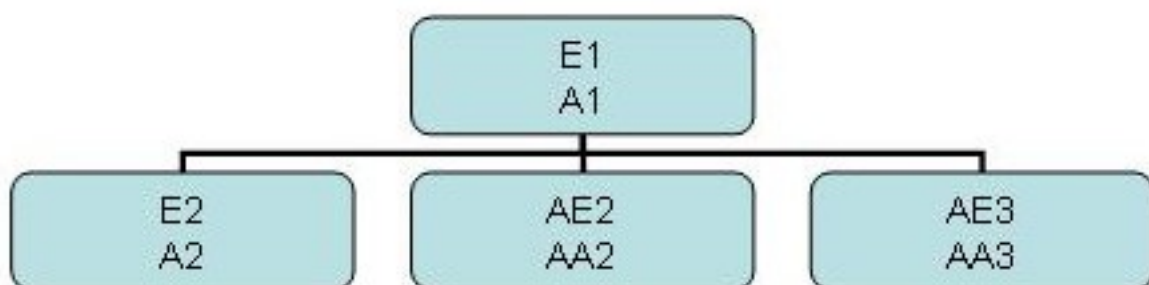


### Anwendungsfallbeschreibung

Summary Identification	<b>Titel</b>	Termin anlegen
	<b>Kurzbeschreibung</b>	Ein neuer Termin wird hinzugefügt.
	<b>Beteiligte Aktoren</b>	Benutzer
	<b>Erstellungsdaten</b>	Daniel Feledi, 15.05.2006, V 1.0

<b>Flow of Events</b>	<b>Hauptszenario</b>	Ein neuer Termin wird in die lokale Datenbank eingefügt, nachdem die Daten für den neuen Termin eingegeben wurden und dieser an einen Termincontainer gebunden wurde.
	<b>Alternative Szenarien</b>	Eingabe wurde vom Benutzer abgebrochen.
	<b>Fehlersituationen</b>	Falls der Termin nicht in die Datenbank eingefügt werden kann, wird eine Fehlermeldung ausgegeben. Ungültige Eingabe durch Benutzer; eine Fehlermeldung / Warnhinweis wird angezeigt, das Bearbeiten-Fenster wird nicht geschlossen und die Daten werden (noch) nicht in die Datenbank geschrieben.
	<b>Vorbedingung</b>	Das System läuft, Programm gestartet
	<b>Nachbedingung</b>	Der neue Termin wurde in der Datenbank mit den eingegebenen Daten gespeichert.
<b>Non-functional-Constraint</b>	<b>Non-Functional-Constraints</b>	Dieser Anwendungsfall wird oft aufgerufen werden. Termine können nicht mit anderen Terminen verbunden werden.

## E/A-Diagramm



E1) Benutzer öffnet das „Datensatz anlegen“ Fenster im Hauptmenü

A1) Das Fenster wird geöffnet

E2) Benutzer wählt „Termin“, gibt die Daten ein, hängt den Termin an einen Termincontainer und bestätigt die Eingabe

A2) Die Daten werden in der Datenbank gespeichert

AE2) Benutzer wählt „Termin“, gibt Daten ein, hängt Termin an Termincontainer fügt Objekt(e) an

autoPSI::Anwendungsfallbeschreibung

Termin und bestätigt die Eingabe

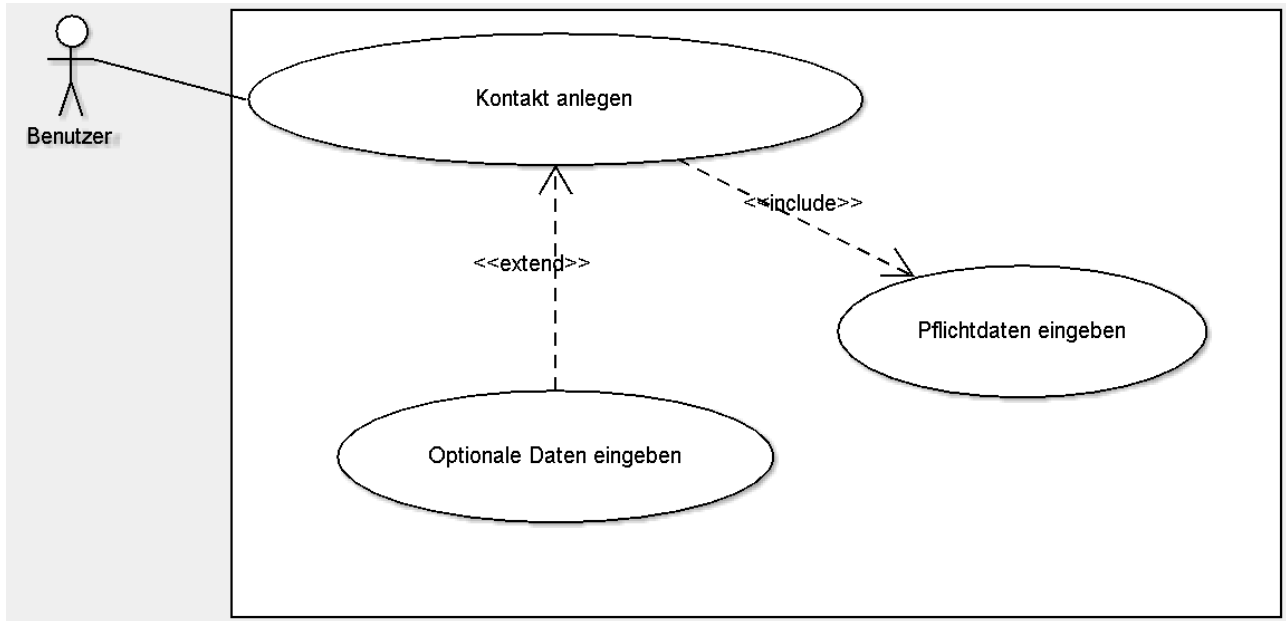
AA2) Daten werden in der Datenbank gespeichert und das Objekt an den Termin gehängt

AE3) Der Benutzer bricht Aktion ab

AA3) Es werden keine Daten in die Datenbank geschrieben

## Kontakt anlegen

### Anwendungsfalldiagramm

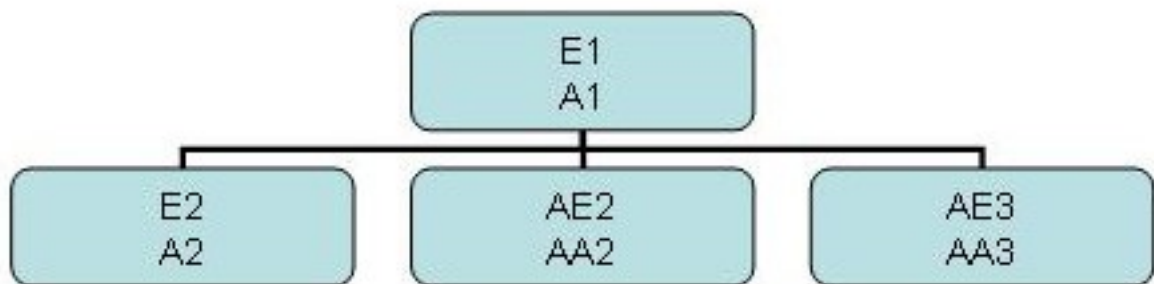


### Anwendungsfallbeschreibung

Summary Identification	<b>Titel</b>	Kontakt anlegen
	<b>Kurzbeschreibung</b>	Ein neuer Kontakt wird hinzugefügt.
	<b>Beteiligte Aktoren</b>	Benutzer
	<b>Erstellungsdaten</b>	Daniel Feledi, 15.05.2006, V 1.0

<b>Flow of Events</b>	<b>Hauptszenario</b>	Ein neuer Kontakt wird in die lokale Datenbank eingefügt, nachdem die Daten für den neuen Kontakt eingegeben wurden.
	<b>Alternative Szenarien</b>	Eingabe wurde vom Benutzer abgebrochen.
	<b>Fehlersituationen</b>	Falls der Kontakt nicht in die Datenbank eingefügt werden kann, wird eine Fehlermeldung ausgegeben. Ungültige Eingabe durch Benutzer; eine Fehlermeldung / Warnhinweis wird angezeigt, das Bearbeiten-Fenster wird nicht geschlossen und die Daten werden (noch) nicht in die Datenbank geschrieben.
	<b>Vorbedingung</b>	Das System läuft, Programm gestartet
	<b>Nachbedingung</b>	Der neue Kontakt wurde in der Datenbank mit den eingegebenen Daten gespeichert.
<b>Non-functional-Constraint</b>	<b>Non-Functional-Constraints</b>	

### E/A-Diagramm



E1) Benutzer öffnet das „Datensatz anlegen“ Fenster im Hauptmenü

A1) Das Fenster wird geöffnet

E2) Benutzer wählt „Kontakt“, gibt die Daten ein und bestätigt die Eingabe

A2) Die Daten werden in der Datenbank gespeichert

AE2) Benutzer wählt „Kontakt“, gibt Daten ein, fügt Objekt(e) an Kontakt und bestätigt die Eingabe

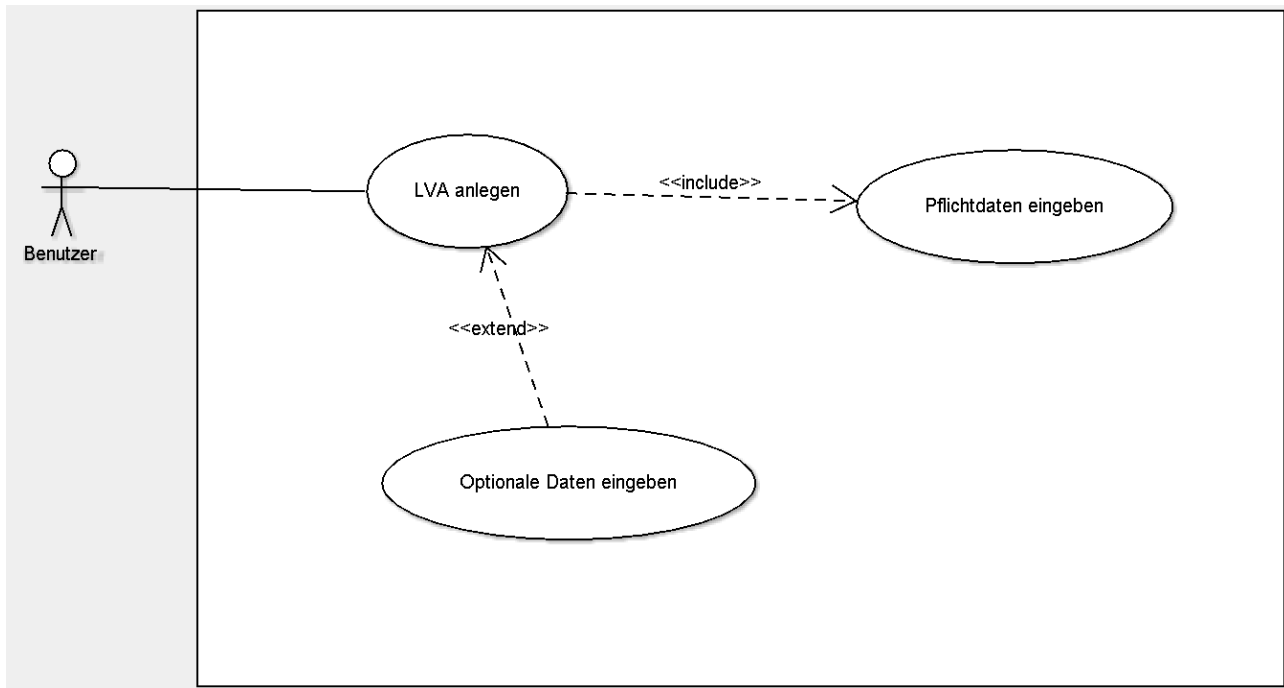
AA2) Daten werden in der Datenbank gespeichert und Objekte an Kontakt gehängt.

AE3) Der Benutzer bricht Aktion ab

AA3) Es werden keine Daten in die Datenbank geschrieben

## LVA anlegen

### Anwendungsfalldiagramm



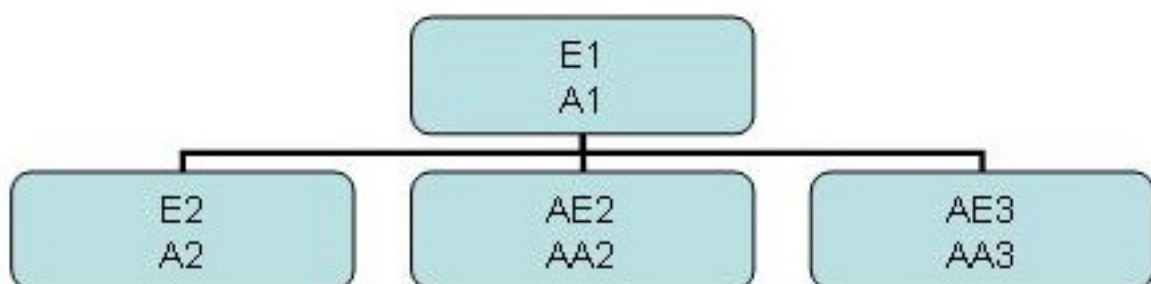
### Anwendungsfallbeschreibung

<b>Summary</b> <b>Identification</b>	<b>Titel</b>	LVA anlegen
	<b>Kurzbeschreibung</b>	Eine neue LVA wird hinzugefügt.
	<b>Beteiligte Aktoren</b>	Benutzer
	<b>Erstellungsdaten</b>	Daniel Feledi, 15.05.2006, V 1.0



<b>Flow of Events</b>	<b>Hauptszenario</b>	Eine neue LVA wird in die lokale Datenbank eingefügt, nachdem die Daten für die neue LVA eingegeben wurden.
	<b>Alternative Szenarien</b>	Eingabe wurde vom Benutzer abgebrochen.
	<b>Fehlersituationen</b>	Falls die LVA nicht in die Datenbank eingefügt werden kann, wird eine Fehlermeldung ausgegeben. Ungültige Eingabe durch Benutzer; eine Fehlermeldung / Warnhinweis wird angezeigt, das Bearbeiten-Fenster wird nicht geschlossen und die Daten werden (noch) nicht in die Datenbank geschrieben.
	<b>Vorbedingung</b>	Das System läuft, Programm gestartet
	<b>Nachbedingung</b>	Die neue LVA wurde in der Datenbank mit den eingegebenen Daten gespeichert.
<b>Non-functional</b>	<b>Non-Functional-Constraints</b>	

## E/A-Diagramm



E1) Benutzer öffnet das „Datensatz anlegen“ Fenster im Hauptmenü

A1) Das Fenster wird geöffnet

E2) Benutzer wählt „LVA“, gibt die Daten ein und bestätigt die Eingabe

A2) Die Daten werden in der Datenbank gespeichert

AE2) Benutzer wählt „LVA“, gibt Daten ein, fügt Objekt(e) an LVA und bestätigt die Eingabe

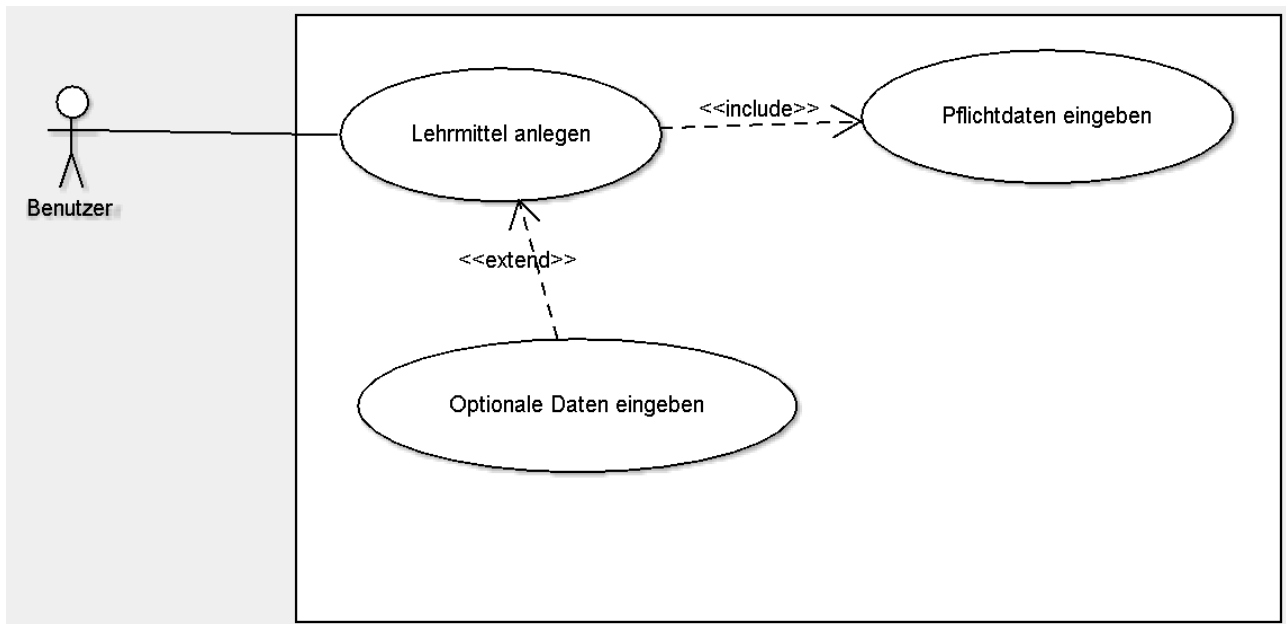
AA2) Daten werden in der Datenbank gespeichert und Objekte an LVA gehängt.

AE3) Der Benutzer bricht Aktion ab

AA3) Es werden keine Daten in die Datenbank geschrieben

## Lehrmittel anlegen

### Anwendungsfalldiagramm

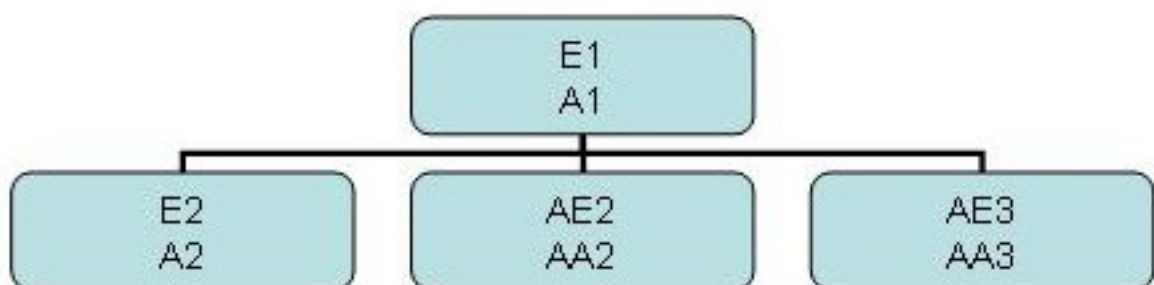


### Anwendungsfallbeschreibung

Summary on Identification	<b>Titel</b>	Lehrmittel anlegen
	<b>Kurzbeschreibung</b>	Ein neues Lehrmittel wird hinzugefügt.
	<b>Beteiligte Aktoren</b>	Benutzer
	<b>Erstellungsdaten</b>	Daniel Feledi, 15.05.2006, V 1.0

<b>Flow of Events</b>	<b>Hauptszenario</b>	Ein neues Lehrmittel wird in die lokale Datenbank eingefügt, nachdem die Daten für das neue Lehrmittel eingegeben wurden.
	<b>Alternative Szenarien</b>	Eingabe wurde vom Benutzer abgebrochen.
	<b>Fehlersituationen</b>	Falls das Lehrmittel nicht in die Datenbank eingefügt werden kann, wird eine Fehlermeldung ausgegeben. Ungültige Eingabe durch Benutzer; eine Fehlermeldung / Warnhinweis wird angezeigt, das Bearbeiten-Fenster wird nicht geschlossen und die Daten werden (noch) nicht in die Datenbank geschrieben.
	<b>Vorbedingung</b>	Das System läuft, Programm gestartet
	<b>Nachbedingung</b>	Das neue Lehrmittel wurde in der Datenbank mit den eingegebenen Daten gespeichert.
<b>Non-functional-Constraint</b>	<b>Non-Functional-Constraints</b>	.

## E/A-Diagramm



E1) Benutzer öffnet das „Datensatz anlegen“ Fenster im Hauptmenü

A1) Das Fenster wird geöffnet

E2) Benutzer wählt „Lehrmittel“, gibt die Daten ein und bestätigt die Eingabe

A2) Die Daten werden in der Datenbank gespeichert

AE2) Benutzer wählt „Lehrmittel“, gibt Daten ein, fügt Objekt(e) an Lehrmittel und bestätigt die Eingabe

AA2) Daten werden in der Datenbank gespeichert und Objekte an Lehrmittel gehängt.

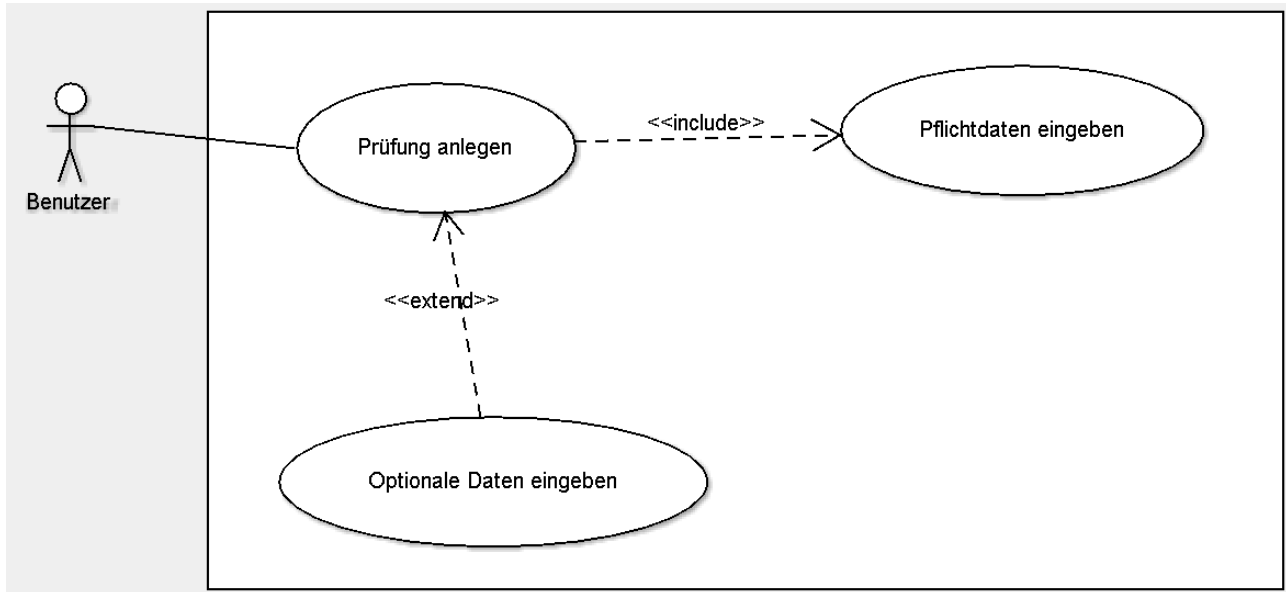
AE3) Der Benutzer bricht Aktion ab

autoPSI::Anwendungsfallbeschreibung

AA3) Es werden keine Daten in die Datenbank geschrieben

## Prüfung anlegen

### Anwendungsfalldiagramm

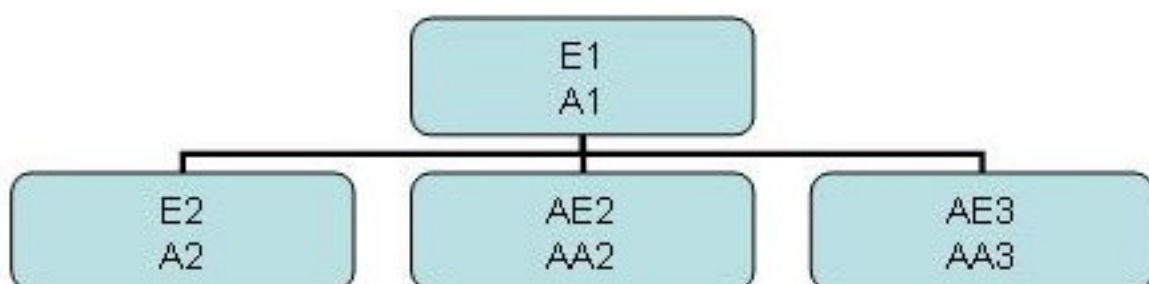


### Anwendungsfallbeschreibung

<b>Summary</b> <b>on</b> <b>Identifi</b>	<b>Titel</b>	Prüfung anlegen
	<b>Kurzbeschreibung</b>	Eine neue Prüfung wird hinzugefügt.
	<b>Beteiligte Aktoren</b>	Benutzer
	<b>Erstellungsdaten</b>	Daniel Feledi, 15.05.2006, V 1.0

<b>Flow of Events</b>	<b>Hauptszenario</b>	Eine neue Prüfung wird in die lokale Datenbank eingefügt, nachdem die Daten für die neue Prüfung eingegeben wurden.
	<b>Alternative Szenarien</b>	Eingabe wurde vom Benutzer abgebrochen.
	<b>Fehlersituationen</b>	Falls die Prüfung nicht in die Datenbank eingefügt werden kann, wird eine Fehlermeldung ausgegeben. Ungültige Eingabe durch Benutzer; eine Fehlermeldung / Warnhinweis wird angezeigt, das Bearbeiten-Fenster wird nicht geschlossen und die Daten werden (noch) nicht in die Datenbank geschrieben.
	<b>Vorbedingung</b>	Das System läuft, Programm gestartet
	<b>Nachbedingung</b>	Die neue Prüfung wurde in der Datenbank mit den eingegebenen Daten gespeichert.
<b>Non-functional-Constraint</b>	<b>Non-Functional-Constraints</b>	

### E/A-Diagramm



E1) Benutzer öffnet das „Datensatz anlegen“ Fenster im Hauptmenü

A1) Das Fenster wird geöffnet

E2) Benutzer wählt „Prüfung“, gibt die Daten ein und bestätigt die Eingabe

A2) Die Daten werden in der Datenbank gespeichert

AE2) Benutzer wählt „Prüfung“, gibt Daten ein, fügt Objekt(e) an Prüfung und bestätigt die Eingabe

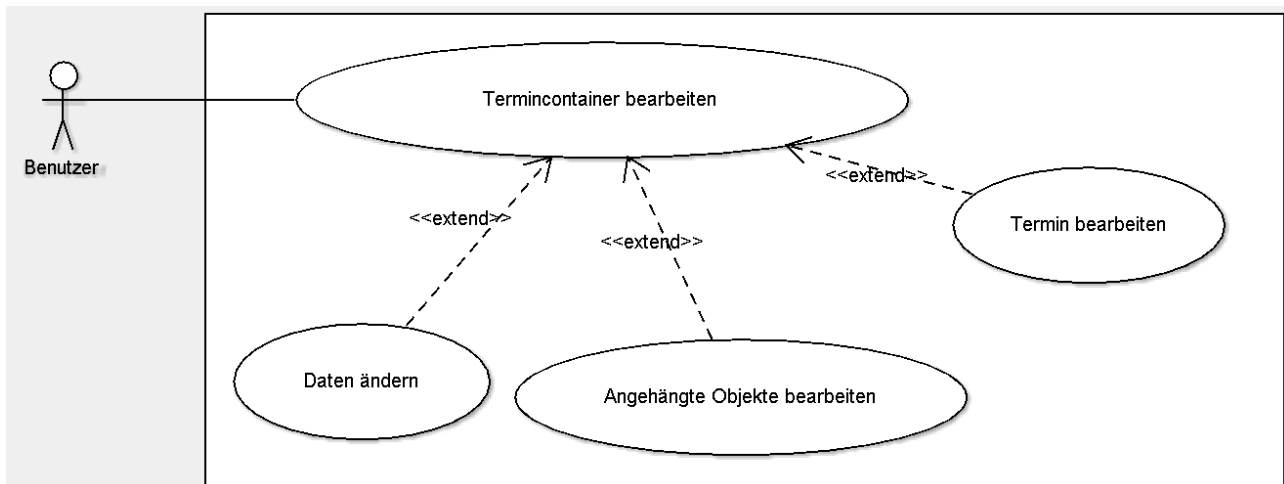
AA2) Daten werden in der Datenbank gespeichert und Objekte an Prüfung gehängt.

AE3) Der Benutzer bricht Aktion ab

AA3) Es werden keine Daten in die Datenbank geschrieben

## Termincontainer bearbeiten

### Anwendungsfalldiagramm

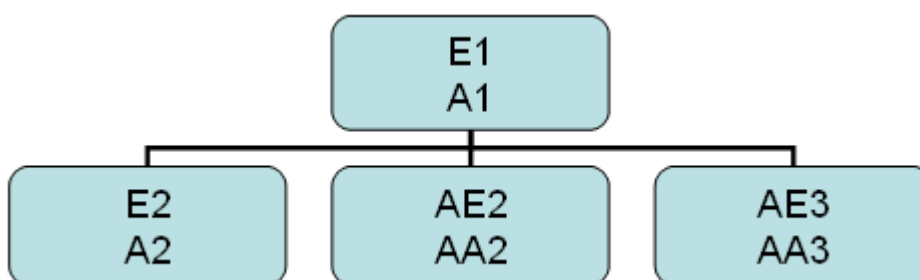


### Anwendungsfallbeschreibung

Summary on Identifi	<b>Titel</b>	Termincontainer bearbeiten
	<b>Kurzbeschreibung</b>	Bereits existierender Termincontainer wird verändert.
	<b>Beteiligte Aktoren</b>	Benutzer
	<b>Erstellungsdaten</b>	Stephan Zapotocky, 15.05.2006, V 1.0

<b>Flow of Events</b>	<b>Hauptszenario</b>	Ein bereits in der DB existierender Termincontainer wird ausgewählt und die zugehörigen Daten geändert / bearbeitet.
	<b>Alternative Szenarien</b>	Abbruch der Bearbeitung durch Benutzer; keine Daten werden in die DB geschrieben.
	<b>Fehlersituationen</b>	Termin bzw. dessen Container kann nicht aus Datenbank gelesen werden; eine Fehlermeldung wird angezeigt. Termincontainer kann nicht in die Datenbank geschrieben werden; eine Fehlermeldung wird angezeigt. Ungültige Eingabe durch Benutzer; eine Fehlermeldung / Warnhinweis wird angezeigt, das Bearbeiten-Fenster wird nicht geschlossen und die Daten werden (noch) nicht in die Datenbank geschrieben.
	<b>Vorbedingung</b>	Das System läuft, Programm gestartet, Termincontainer eingefügt
	<b>Nachbedingung</b>	Die neu eingegeben Daten wurden in die Datenbank geschrieben.
<b>Non-functional-Constraint</b>	<b>Non-Functional-Constraints</b>	

### E/A-Diagramm



E1) Der Benutzer wählt einen Termin im Kalender aus und öffnet über einen Button das Bearbeitungsfenster für den zugehörigen Termincontainer

A1) Das Bearbeitungsfenster für den Termincontainer wird angezeigt.

E2) Der Benutzer bearbeitet die Daten des Termincontainers und bestätigt seine Änderungen per



Button.

A2) Das System speichert die Änderungen in die Datenbank.

AE2) Der Benutzer bricht die Änderung des Termincontainers ab.

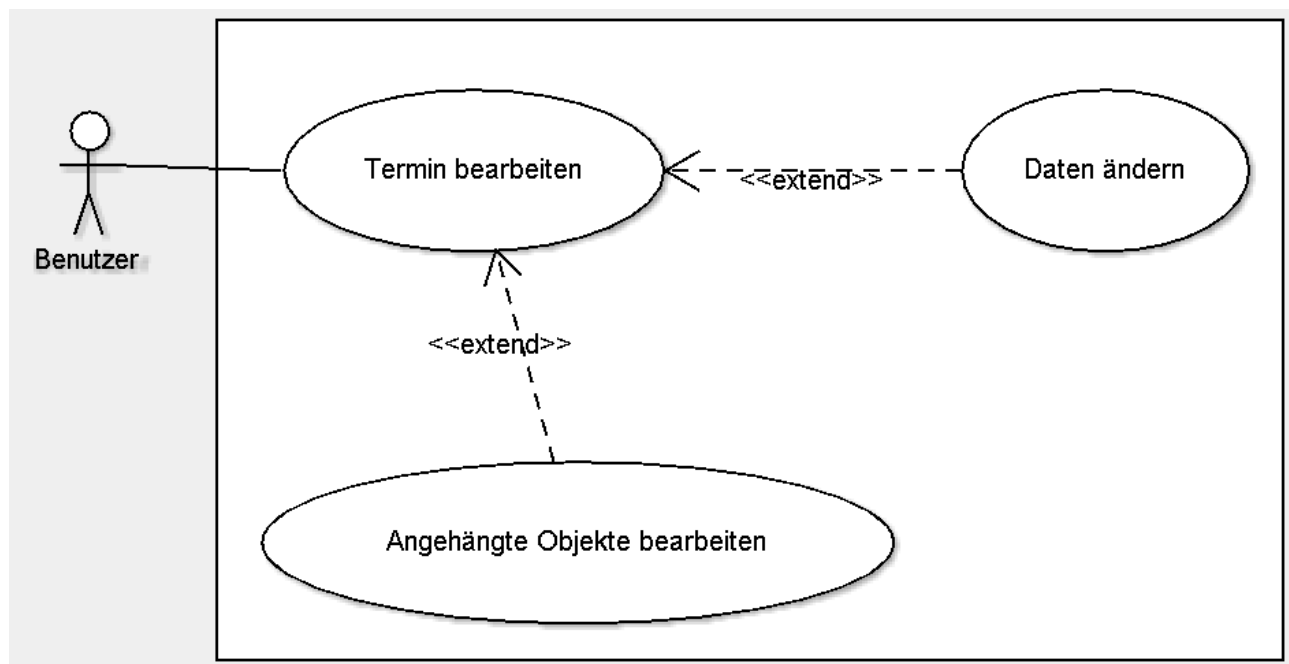
AE2) Das Bearbeitungsfenster schließt sich. Es werden keine Änderungen in der Datenbank vorgenommen.

AE3) Der Benutzer bearbeitet die Daten des Termincontainers und bestätigt seine Änderungen per Button. Allerdings tritt ein Fehler in der DB-Verbindung auf.

AA3) Ein entsprechender Warnhinweis wird angezeigt. Das Bearbeitungsfenster bleibt offen.

## Termin bearbeiten

### Anwendungsfalldiagramm

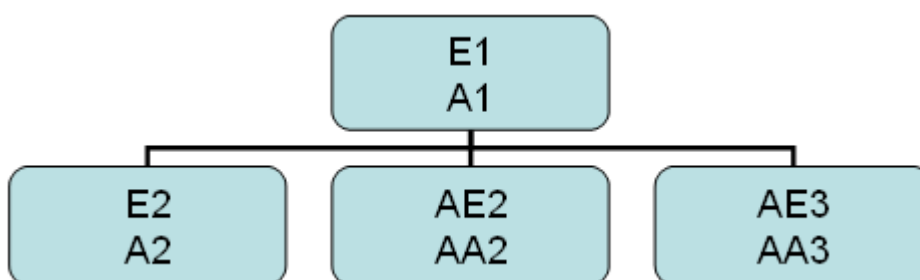


### Anwendungsfallbeschreibung

Summary on Identificati	<b>Titel</b>	Termin bearbeiten
	<b>Kurzbeschreibung</b>	Schon existierender Termin wird verändert.
	<b>Beteiligte Akteure</b>	Benutzer
	<b>Erstellungsdaten</b>	Stephan Zapotocky, 15.05.2006, V 1.0

<b>Flow of Events</b>	<b>Hauptszenario</b>	Ein bereits in der DB existierender Termin wird ausgewählt und die zugehörigen Daten geändert / bearbeitet.
	<b>Alternative Szenarien</b>	Abbruch der Bearbeitung durch Benutzer; keine Daten werden in die DB geschrieben.
	<b>Fehlersituationen</b>	Termin kann nicht aus Datenbank gelesen werden; eine Fehlermeldung wird angezeigt. Termin kann nicht in die Datenbank geschrieben werden; eine Fehlermeldung wird angezeigt. Ungültige Eingabe durch Benutzer; eine Fehlermeldung / Warnhinweis wird angezeigt, das Bearbeiten-Fenster wird nicht geschlossen und die Daten werden (noch) nicht in die Datenbank geschrieben.
	<b>Vorbedingung</b>	Das System läuft, Programm gestartet, Termin eingefügt
	<b>Nachbedingung</b>	Die neu eingegeben Daten wurden in die Datenbank geschrieben.
<b>Non-functional-Constraint</b>	<b>Non-Functional-Constraints</b>	

### E/A-Diagramm



E1) Der Benutzer wählt einen Termin im Kalender aus und öffnet über einen Button dessen Bearbeitungsfenster.

A1) Das Bearbeitungsfenster für den Termin wird angezeigt.

E2) Der Benutzer bearbeitet die Daten des Termins und bestätigt seine Änderungen per Button.

A2) Das System speichert die Änderungen in die Datenbank.

AE2) Der Benutzer bricht die Änderung des Termins ab.

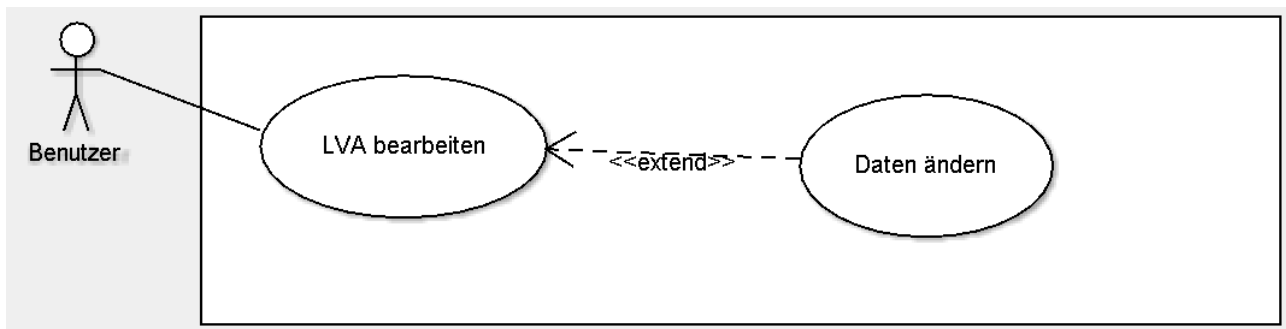
AE2) Das Bearbeitungsfenster schließt sich. Es werden keine Änderungen in der Datenbank vorgenommen.

AE3) Der Benutzer bearbeitet die Daten des Termins und bestätigt seine Änderungen per Button.  
Allerdings tritt ein Fehler in der DB-Verbindung auf.

AA3) Ein entsprechender Warnhinweis wird angezeigt. Das Bearbeitungsfenster bleibt offen.

## LVA bearbeiten

### Anwendungsfalldiagramm

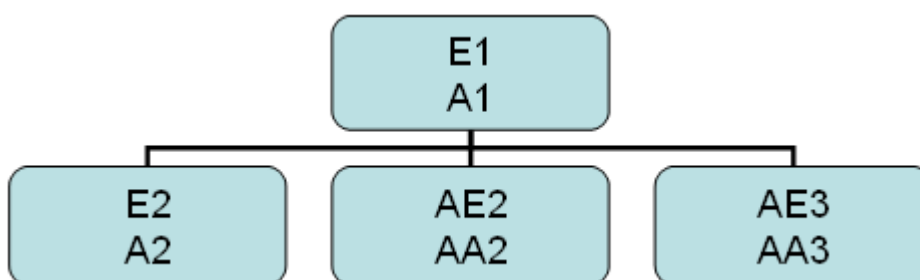


### Anwendungsfallbeschreibung

<b>Summary</b> <b>on</b> <b>Identificati</b>	<b>Titel</b>	LVA bearbeiten
	<b>Kurzbeschreibung</b>	Bereits existierende LVA wird verändert.
	<b>Beteiligte Aktoren</b>	Benutzer
	<b>Erstellungsdaten</b>	Stephan Zapotocky, 15.05.2006, V 1.0

<b>Flow of Events</b>	<b>Hauptszenario</b>	Eine bereits in der DB existierende LVA wird ausgewählt und die zugehörigen Daten geändert / bearbeitet.
	<b>Alternative Szenarien</b>	Abbruch der Bearbeitung durch Benutzer; keine Daten werden in die DB geschrieben.
	<b>Fehlersituationen</b>	LVA kann nicht aus Datenbank gelesen werden; eine Fehlermeldung wird angezeigt. LVA kann nicht in die Datenbank geschrieben werden; eine Fehlermeldung wird angezeigt. Ungültige Eingabe durch Benutzer; eine Fehlermeldung / Warnhinweis wird angezeigt, das Bearbeiten-Fenster wird nicht geschlossen und die Daten werden (noch) nicht in die Datenbank geschrieben.
	<b>Vorbedingung</b>	Das System läuft, Programm gestartet, LVA eingefügt
	<b>Nachbedingung</b>	Die neu eingegeben Daten wurden in die Datenbank geschrieben.
<b>Non-functional-Constraint</b>	<b>Non-Functional-Constraints</b>	

### E/A-Diagramm



E1) Der Benutzer wählt einen Termin im Kalender aus und öffnet über einen Button das Bearbeitungsfenster für die zugehörige LVA.

A1) Das Bearbeitungsfenster für die LVA wird angezeigt.

E2) Der Benutzer bearbeitet die Daten der LVA und bestätigt seine Änderungen per Button.

A2) Das System speichert die Änderungen in die Datenbank.

AE2) Der Benutzer bricht die Änderung der LVA ab.

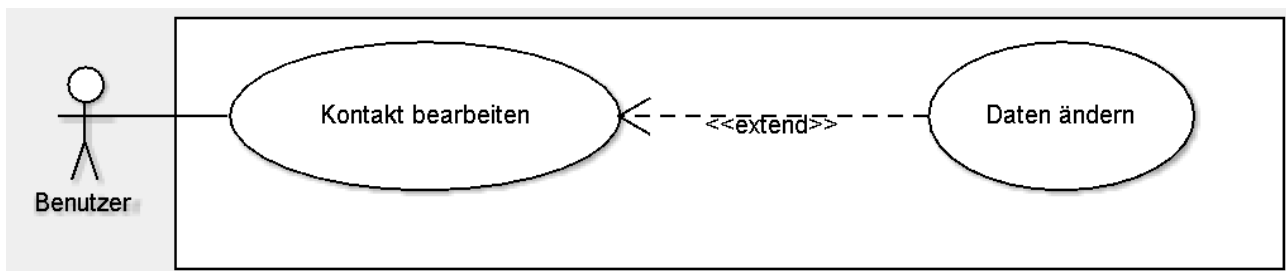
AE2) Das Bearbeitungsfenster schließt sich. Es werden keine Änderungen in der Datenbank vorgenommen.

AE3) Der Benutzer bearbeitet die Daten der LVA und bestätigt seine Änderungen per Button.  
Allerdings tritt ein Fehler in der DB-Verbindung auf.

AA3) Ein entsprechender Warnhinweis wird angezeigt. Das Bearbeitungsfenster bleibt offen.

## Kontakt bearbeiten

### Anwendungsfalldiagramm

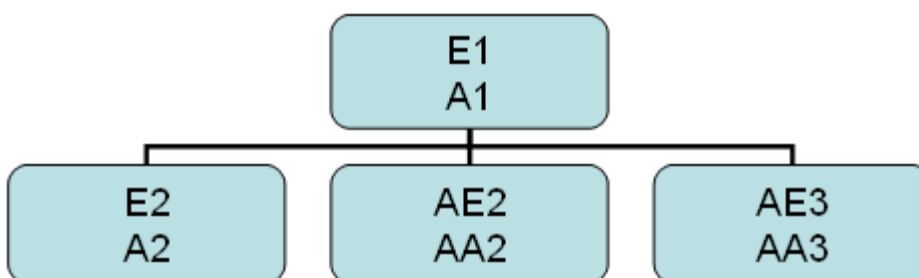


### Anwendungsfallbeschreibung

<b>Summary</b> on <b>Identificati</b>	<b>Titel</b>	Kontakt bearbeiten
	<b>Kurzbeschreibung</b>	Bereits existierender Kontakt wird verändert.
	<b>Beteiligte Aktoren</b>	Benutzer
	<b>Erstellungsdaten</b>	Stephan Zapotocky, 15.05.2006, V 1.0

<b>Flow of Events</b>	<b>Hauptszenario</b>	Ein bereits in der DB existierender Kontakt wird bearbeitet. Über eine durch Suchkriterien einschränkbare Liste wird ein Kontakt ausgewählt, ein Fenster geöffnet und die Kontaktdaten darin können dann bearbeitet werden. Die Änderungen werden in die DB gespeichert.
	<b>Alternative Szenarien</b>	Abbruch der Bearbeitung durch Benutzer; keine Daten werden in die DB geschrieben bzw. keine Änderungen vorgenommen.
	<b>Fehlersituationen</b>	Kontakt kann nicht aus Datenbank gelesen werden; eine Fehlermeldung wird angezeigt. Kontakt kann nicht in die Datenbank geschrieben werden; eine Fehlermeldung wird angezeigt. Ungültige Eingabe durch Benutzer; eine Fehlermeldung / Warnhinweis wird angezeigt, das Bearbeiten-Fenster wird nicht geschlossen und die Daten werden (noch) nicht in die Datenbank geschrieben.
	<b>Vorbedingung</b>	Das System läuft, Programm gestartet, Kontakt eingefügt
	<b>Nachbedingung</b>	Die neu eingegeben Daten wurden in die Datenbank geschrieben.
<b>Non-functional-Constraint</b>	<b>Non-Functional-Constraints</b>	

### E/A-Diagramm



E1) Der Benutzer öffnet die Auflistung der Kontakte und wählt daraus einen aus. Alternativ kann man auch über die Infobar eines angeklickten Termins auf Kontakte zugreifen.

A1) Das Bearbeitungsfenster für den Kontakt wird angezeigt.

E2) Der Benutzer bearbeitet die Daten des Kontakts und bestätigt seine Änderungen per Button.

A2) Das System speichert die Änderungen in die Datenbank.

AE2) Der Benutzer bricht die Änderung des Kontakts ab.

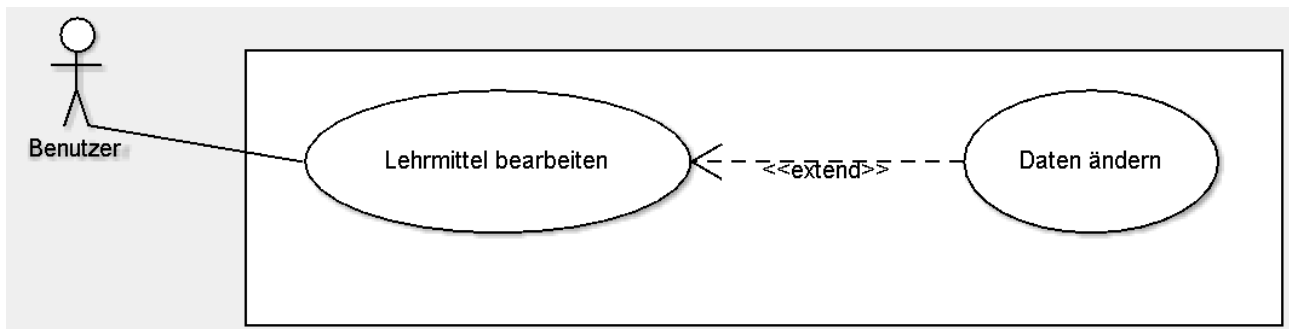
AE2) Das Bearbeitungsfenster schließt sich. Es werden keine Änderungen in der Datenbank vorgenommen.

AE3) Der Benutzer bearbeitet die Daten des Kontakts und bestätigt seine Änderungen per Button. Allerdings tritt ein Fehler in der DB-Verbindung auf.

AA3) Ein entsprechender Warnhinweis wird angezeigt. Das Bearbeitungsfenster bleibt offen.

## Lehrmittel bearbeiten

### Anwendungsfalldiagramm

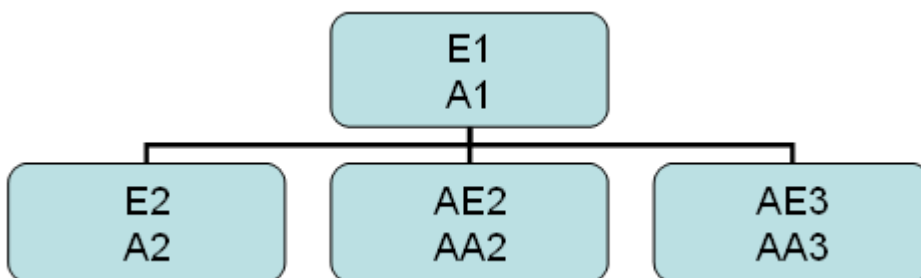


### Anwendungsfallbeschreibung

<b>Summary</b> <b>on</b> <b>Identificati</b>	<b>Titel</b>	Lehrmittel bearbeiten
	<b>Kurzbeschreibung</b>	Bereits existierendes Lehrmittel wird verändert.
	<b>Beteiligte Aktoren</b>	Benutzer
	<b>Erstellungsdaten</b>	Stephan Zapotocky, 15.05.2006, V 1.0

<b>Flow of Events</b>	<b>Hauptszenario</b>	Ein bereits in der DB existierendes Lehrmittel wird bearbeitet. Über das Bearbeitungsfenster eines Termins oder eines Termincontainers kann die Liste der angehängten Lehrmittel bearbeitet werden.
	<b>Alternative Szenarien</b>	Abbruch der Bearbeitung durch Benutzer; keine Daten werden in die DB geschrieben bzw. keine Änderungen vorgenommen.
	<b>Fehlersituationen</b>	Lehrmittel kann nicht aus Datenbank gelesen werden; eine Fehlermeldung wird angezeigt. Lehrmittel kann nicht in die Datenbank geschrieben werden; eine Fehlermeldung wird angezeigt. Ungültige Eingabe durch Benutzer; eine Fehlermeldung / Warnhinweis wird angezeigt, das Bearbeiten-Fenster wird nicht geschlossen und die Daten werden (noch) nicht in die Datenbank geschrieben.
	<b>Vorbedingung</b>	Das System läuft, Programm gestartet, Lehrmittel eingefügt
	<b>Nachbedingung</b>	Die neu eingegeben Daten wurden in die Datenbank geschrieben.
<b>Non-functional-Constraint</b>	<b>Non-Functional-Constraints</b>	

### E/A-Diagramm



E1) Der Benutzer öffnet das Bearbeitungsfenster eines Termins oder eines Termincontainers.

A1) Das Bearbeitungsfenster für den Termin/Termincontainer wird mit einer Liste der angehängten Lehrmittel angezeigt.

E2) Der Benutzer bearbeitet die Lehrmittelliste und bestätigt seine Änderungen per Button.



## autoPSI::Anwendungsfallbeschreibung

A2) Das System speichert die Änderungen in die Datenbank.

AE2) Der Benutzer bricht die Änderung der Lehrmittelliste ab.

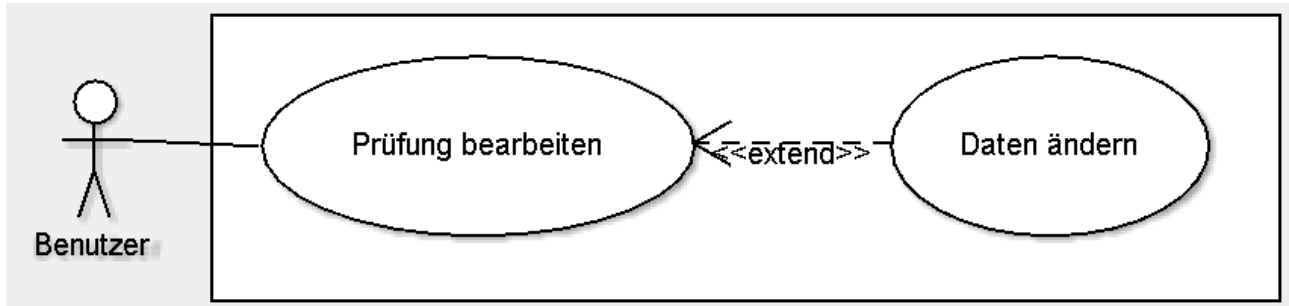
AE2) Das Bearbeitungsfenster schließt sich. Es werden keine Änderungen in der Datenbank vorgenommen.

AE3) Der Benutzer bearbeitet die Lehrmittelliste und bestätigt seine Änderungen per Button.  
Allerdings tritt ein Fehler in der DB-Verbindung auf.

AA3) Ein entsprechender Warnhinweis wird angezeigt. Das Bearbeitungsfenster bleibt offen.

## Prüfung bearbeiten

### Anwendungsfalldiagramm

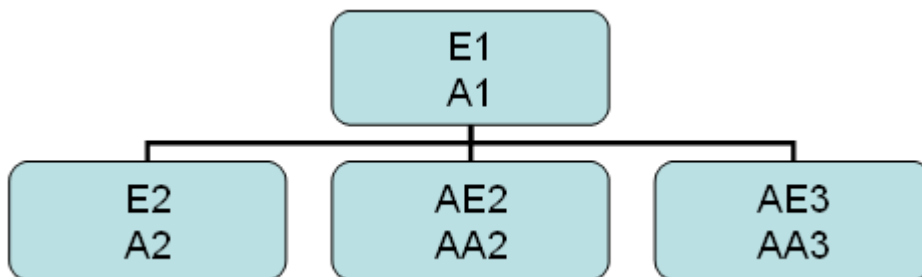


### Anwendungsfallbeschreibung

<b>Summary on Identification</b>	<b>Titel</b>	Prüfung bearbeiten
	<b>Kurzbeschreibung</b>	Daten einer bereits existierenden Prüfung werden geändert.
	<b>Beteiligte Aktoren</b>	Benutzer
	<b>Erstellungsdaten</b>	Stephan Zapotocky, 15.05.2006, V 1.0

<b>Flow of Events</b>	<b>Hauptszenario</b>	Ein bereits in der DB existierender Prüfungstermin wird bearbeitet. Entweder man wählt diesen aus einer Liste im LVA-Bearbeitungsfenster aus oder klickt direkt auf den Prüfungstermin im Kalender. Die Daten können bearbeitet und anschließend in die DB gespeichert werden.
	<b>Alternative Szenarien</b>	Abbruch der Bearbeitung durch Benutzer; keine Daten werden in die DB geschrieben bzw. keine Änderungen vorgenommen.
	<b>Fehlersituationen</b>	Prüfung kann nicht aus Datenbank gelesen werden; eine Fehlermeldung wird angezeigt. Prüfung kann nicht in die Datenbank geschrieben werden; eine Fehlermeldung wird angezeigt. Ungültige Eingabe durch Benutzer; eine Fehlermeldung / Warnhinweis wird angezeigt, das Bearbeiten-Fenster wird nicht geschlossen und die Daten werden (noch) nicht in die Datenbank geschrieben.
	<b>Vorbedingung</b>	Das System läuft, Programm gestartet, Prüfung eingefügt
	<b>Nachbedingung</b>	Die neu eingegeben Daten wurden in die Datenbank geschrieben.
<b>Non-functional-Constraint</b>	<b>Non-Functional-Constraints</b>	

### E/A-Diagramm



E1) Der Benutzer öffnet das Bearbeitungsfenster der Prüfung.

A1) Das Bearbeitungsfenster wird angezeigt.

E2) Der Benutzer bearbeitet den Eintrag der Prüfung und bestätigt seine Änderungen per Button.

## autoPSI::Anwendungsfallbeschreibung

A2) Das System speichert die Änderungen in die Datenbank.

AE2) Der Benutzer bricht die Änderung des Prüfungseintrags ab.

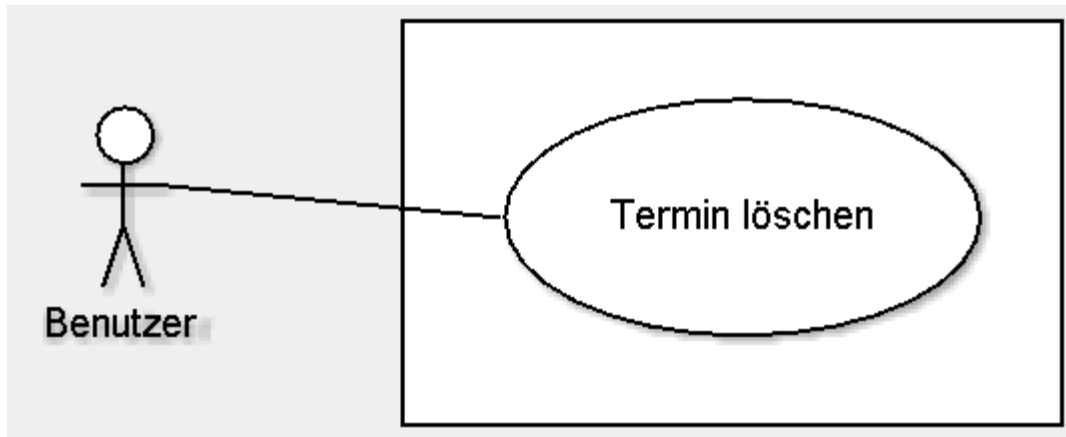
AE2) Das Bearbeitungsfenster schließt sich. Es werden keine Änderungen in der Datenbank vorgenommen.

AE3) Der Benutzer bearbeitet den Prüfungseintrag und bestätigt seine Änderungen per Button.  
Allerdings tritt ein Fehler in der DB-Verbindung auf.

AA3) Ein entsprechender Warnhinweis wird angezeigt. Das Bearbeitungsfenster bleibt offen.

## Termin löschen

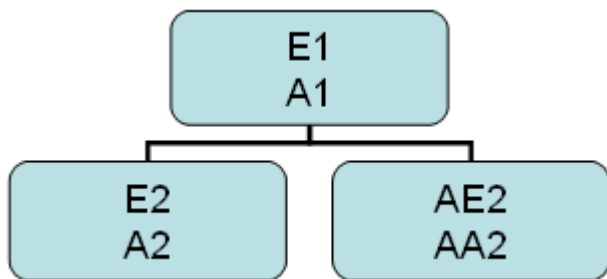
### Anwendungsfalldiagramm



### Anwendungsfallbeschreibung

Summary	Identification	Titel	Termin löschen
		Kurzbeschreibung	Termin wird gelöscht.
		Beteiligte Aktoren	Benutzer
		Erstellungsdaten	Rudolf Mildner, 15.05.2006, V 1.0
	Flow of Events	Hauptszenario	Benutzer wählt einen Termin; Eine Sicherheitsabfrage wird angezeigt. Der Termin wird gelöscht.
		Alternative Szenarien	Der Benutzer beantwortet die Sicherheitsanfrage negativ; der Termin wird nicht gelöscht
		Fehlersituationen	Daten können aus der Datenbank nicht gelöscht werden; Es wird eine Fehlermeldung angezeigt.
		Vorbedingung	Das System läuft, Programm gestartet, Termin eingefügt
		Nachbedingung	Der Termin wurde aus der Datenbank gelöscht.
Non-functional-Constraints	Non-functional-Constraints	Dieser Anwendungsfall wird oft aufgerufen werden.	

### ***E/A-Diagramm***



E1) Der Benutzer wählt einen Termin im Kalender zum Löschen aus.

A1) Eine Sicherheitsabfrage wird angezeigt.

E2) Der Benutzer bestätigt den Löschvorgang durch Klicken des OK-Buttons

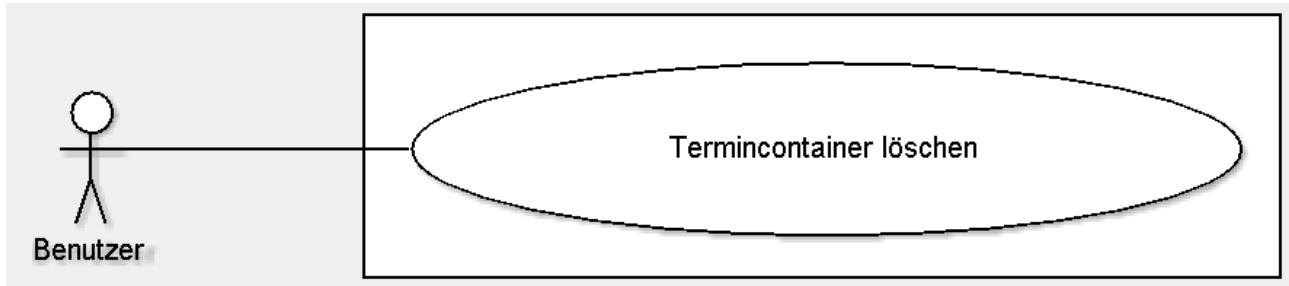
A2) Das System löscht den Termin aus dem Kalender, der Datenbank und eventuell aus dem Space.  
Die Verknüpfung zu eventuell angehängten Objekten wird getrennt, die Objekte werden aber nicht gelöscht.

AE2) Der Benutzer bricht den Löschvorgang ab.

AE2) Der Dialog schließt sich, es werden keine Änderungen vorgenommen.

## Termincontainer löschen

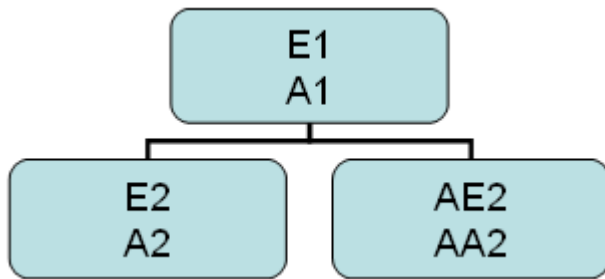
### Anwendungsfalldiagramm



### Anwendungsfallbeschreibung

<b>Summary</b>	<b>Identifikation</b>	<b>Titel</b>	Termincontainer löschen
		<b>Kurzbeschreibung</b>	Termincontainer wird gelöscht.
		<b>Beteiligte Akteure</b>	Benutzer
		<b>Erstellungsdaten</b>	Rudolf Mildner, 15.05.2006, V 1.0
<b>Flow of Events</b>		<b>Hauptscenario</b>	Benutzer wählt einen Termincontainer; Eine Sicherheitsabfrage wird angezeigt. Der Termincontainer wird gelöscht.
		<b>Alternative Szenarien</b>	Der Benutzer beantwortet die Sicherheitsanfrage negativ; der Termincontainer wird nicht gelöscht
		<b>Fehlersituationen</b>	Daten können aus der Datenbank nicht gelöscht werden; Es wird eine Fehlermeldung angezeigt.
		<b>Vorbedingung</b>	Das System läuft, Programm gestartet, Zeitpunkt eingefügt
		<b>Nachbedingung</b>	Der Termincontainer wurde aus der Datenbank gelöscht.
<b>Non-functional-Constraint</b>		<b>Non-Functional-Constraints</b>	Dieser Anwendungsfall wird oft aufgerufen werden.

### ***E/A-Diagramm***



E1) Der Benutzer wählt einen Termincontainer im Kalender über die Infobar eines Termins zum Löschen aus.

A1) Eine Sicherheitsabfrage wird angezeigt.

E2) Der Benutzer bestätigt den Löschvorgang durch Klicken des OK-Buttons

A2) Das System löscht den Termincontainer aus dem Kalender, der Datenbank und eventuell aus dem Space. Die Verknüpfung zu eventuell angehängten Objekten wird getrennt, die Objekte werden aber nicht gelöscht.

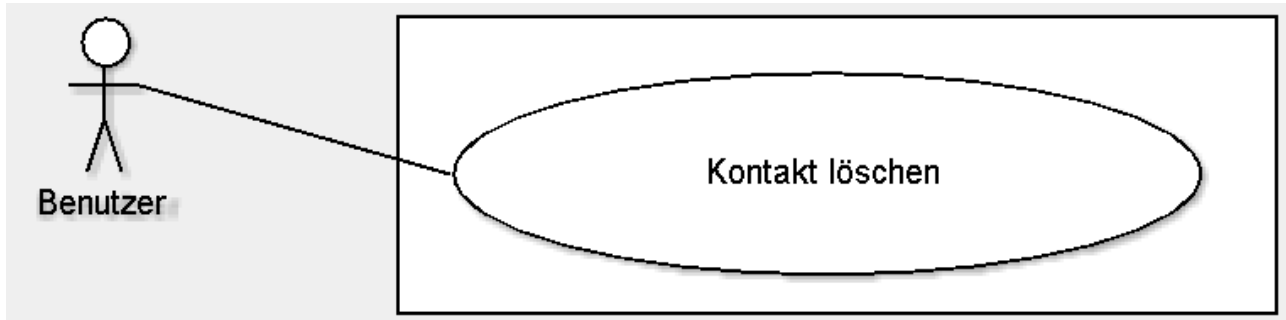
AE2) Der Benutzer bricht den Löschvorgang ab.

AE2) Der Dialog schließt sich, es werden keine Änderungen vorgenommen.



## Kontakt löschen

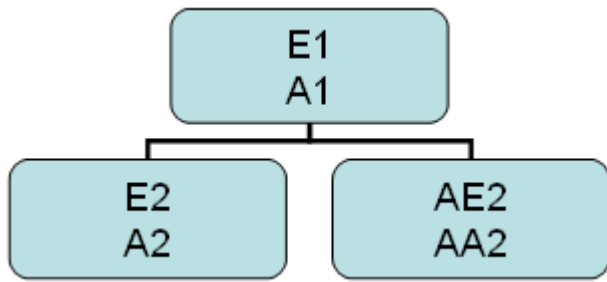
### Anwendungsfalldiagramm



### Anwendungsfallbeschreibung

<b>Summary</b>	<b>Identifikation</b>	<b>Titel</b>	Kontakt löschen
		<b>Kurzbeschreibung</b>	Kontakt wird gelöscht.
		<b>Beteiligte Aktoren</b>	Benutzer
		<b>Erstellungsdaten</b>	Rudolf Mildner, 15.05.2006, V 1.0
<b>Flow of Events</b>		<b>Hauptscenario</b>	Benutzer wählt einen Kontakt; Eine Sicherheitsabfrage wird angezeigt. Der Kontakt wird gelöscht.
		<b>Alternative Szenarien</b>	Der Benutzer beantwortet die Sicherheitsanfrage negativ; der Kontakt wird nicht gelöscht
		<b>Fehlersituationen</b>	Daten können aus der Datenbank nicht gelöscht werden; Es wird eine Fehlermeldung angezeigt.
		<b>Vorbedingung</b>	Das System läuft, Programm gestartet, Kontakt eingefügt
		<b>Nachbedingung</b>	Der Kontakt wurde aus der Datenbank gelöscht.
<b>Non-functional</b>		<b>Non-Functional-Constraints</b>	Dieser Anwendungsfall wird oft aufgerufen werden.

### ***E/A-Diagramm***



E1) Der Benutzer wählt einen Kontakt zum Löschen aus.

A1) Eine Sicherheitsabfrage wird angezeigt.

E2) Der Benutzer bestätigt den Löschvorgang durch Klicken des OK-Buttons

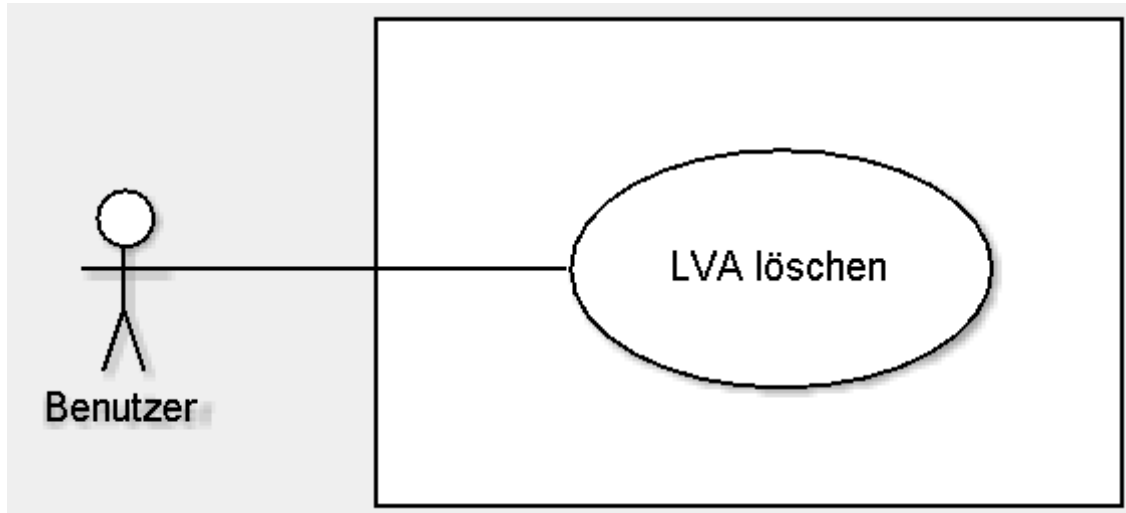
A2) Das System löscht den Kontakt aus der Kontaktliste, der Datenbank und eventuell aus dem Space.

AE2) Der Benutzer bricht den Löschvorgang ab.

AA2) Der Dialog schließt sich, es werden keine Änderungen vorgenommen.

## LVA löschen

### Anwendungsfalldiagramm

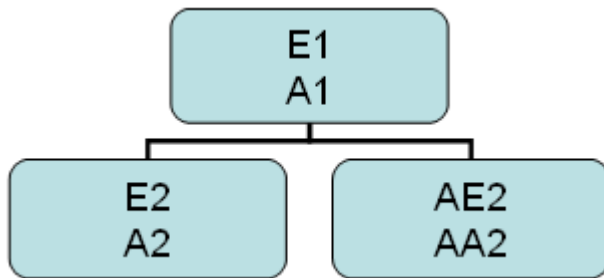


### Anwendungsfallbeschreibung

<b>Summary</b>	<b>Identifikation</b>	<b>Titel</b>	LVA löschen
		<b>Kurzbeschreibung</b>	LVA wird gelöscht.
		<b>Beteiligte Aktoren</b>	Benutzer
		<b>Erstellungsdaten</b>	Rudolf Mildner, 15.05.2006, V 1.0
<b>Flow of Events</b>		<b>Hauptszenario</b>	Benutzer wählt eine LVA; Eine Sicherheitsabfrage wird angezeigt. Die LVA wird gelöscht.
		<b>Alternative Szenarien</b>	Der Benutzer beantwortet die Sicherheitsanfrage negativ; die LVA wird nicht gelöscht
		<b>Fehlersituationen</b>	Daten können aus der Datenbank nicht gelöscht werden; Es wird eine Fehlermeldung angezeigt.
		<b>Vorbedingung</b>	Das System läuft, Programm gestartet, LVA eingefügt
		<b>Nachbedingung</b>	Die LVA wurde aus der Datenbank gelöscht.

<b>Non-functional-Constraint</b>	<b>Non-Functional-Constraints</b>	Dieser Anwendungsfall wird oft aufgerufen werden.
----------------------------------	-----------------------------------	---

### ***E/A-Diagramm***



E1) Der Benutzer wählt eine LVA zum Löschen aus. Dies geschieht entweder über das Bearbeiten-Fenster des Termins/Termincontainers oder über eine aufrufbare LVA-Liste.

A1) Eine Sicherheitsabfrage wird angezeigt.

E2) Der Benutzer bestätigt den Löschvorgang durch Klicken des OK-Buttons

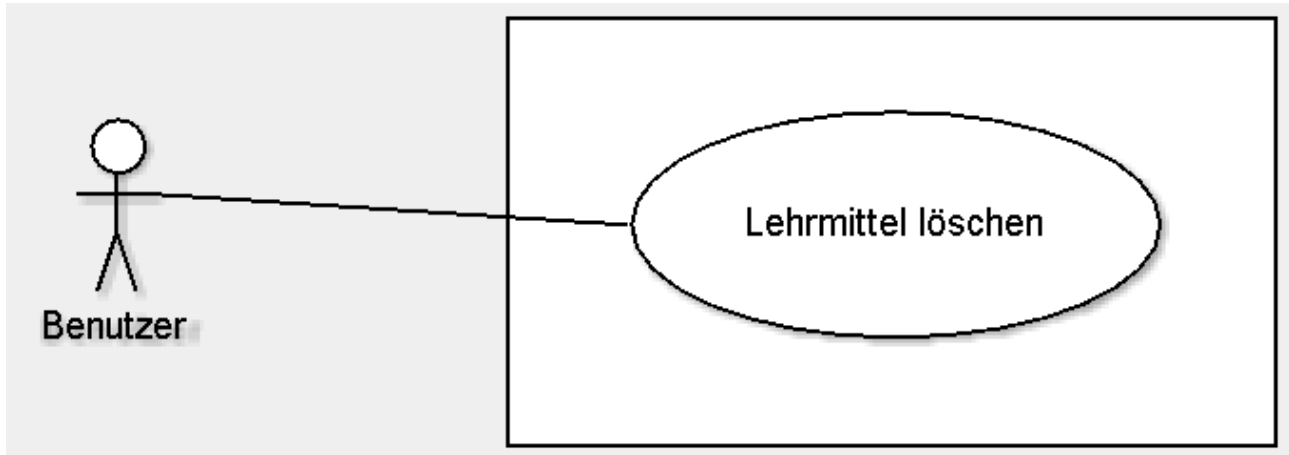
A2) Das System löscht die LVA aus der LVA-Liste, der Datenbank und eventuell aus dem Space. Anknüpfungen der LVA an Termine und Termincontainer werden gelöscht.

AE2) Der Benutzer bricht den Löschvorgang ab.

AE2) Der Dialog schließt sich, es werden keine Änderungen vorgenommen.

## Lehrmittel löschen

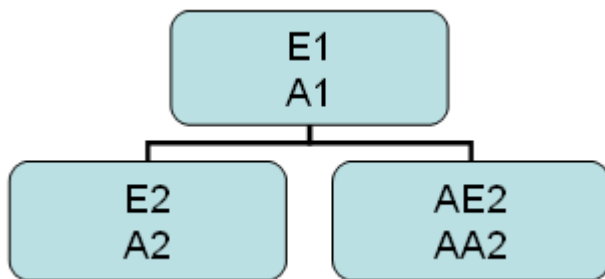
### Anwendungsfalldiagramm



### Anwendungsfallbeschreibung

Summary	Identificati on	<b>Titel</b>	Lehrmittel löschen
		<b>Kurzbeschreibung</b>	Lehrmittel wird gelöscht.
		<b>Beteiligte Aktoren</b>	Benutzer
		<b>Erstellungsdaten</b>	Rudolf Mildner, 15.05.2006, V 1.0
Flow of Events		<b>Hauptszenario</b>	Benutzer wählt ein Lehrmittel; Eine Sicherheitsabfrage wird angezeigt. Das Lehrmittel wird gelöscht.
		<b>Alternative Szenarien</b>	Der Benutzer beantwortet die Sicherheitsanfrage negativ; das Lehrmittel wird nicht gelöscht
		<b>Fehlersituationen</b>	Daten können aus der Datenbank nicht gelöscht werden; Es wird eine Fehlermeldung angezeigt.
		<b>Vorbedingung</b>	Das System läuft, Programm gestartet, Lehrmittel eingefügt
		<b>Nachbedingung</b>	Das Lehrmittel wurde aus der Datenbank gelöscht.
functional- Constraint	Non-	<b>Non-Functional-Constraints</b>	Dieser Anwendungsfall wird oft aufgerufen werden.

### ***E/A-Diagramm***



E1) Der Benutzer wählt ein Lehrmittel zum Löschen aus. Dies geschieht über die Lehrmittelliste in der Infobar des gewählten Termin/Termincontainers.

A1) Eine Sicherheitsabfrage wird angezeigt.

E2) Der Benutzer bestätigt den Löschvorgang durch Klicken des OK-Buttons

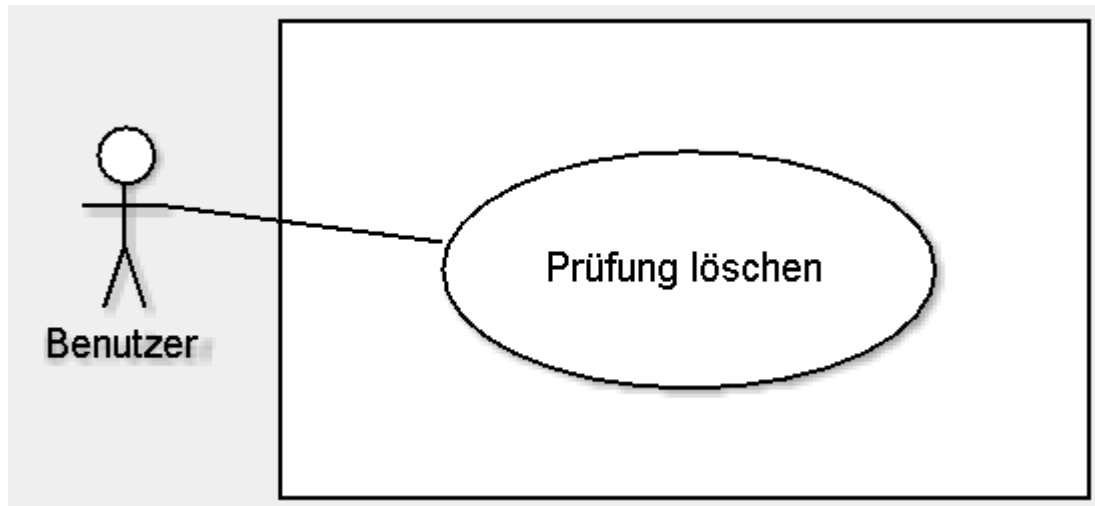
A2) Das System löscht das Lehrmittel aus der Lehrmittelliste, der Datenbank und eventuell aus dem Space. Anknüpfungen des Lehrmittels an Termine und Termincontainer werden gelöscht.

AE2) Der Benutzer bricht den Löschvorgang ab.

AE2) Der Dialog schließt sich, es werden keine Änderungen vorgenommen.

## Prüfung löschen

### Anwendungsfalldiagramm

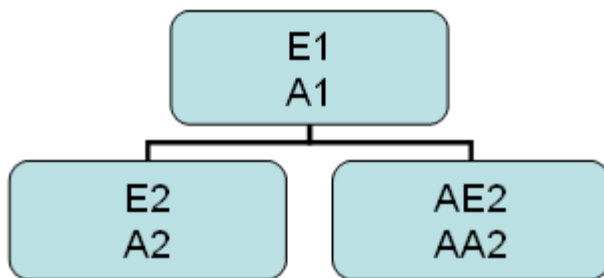


### Anwendungsfallbeschreibung

<b>Summary</b>	<b>Identifikation</b>	<b>Titel</b>	Prüfung löschen
		<b>Kurzbeschreibung</b>	Prüfung wird gelöscht.
		<b>Beteiligte Aktoren</b>	Benutzer
		<b>Erstellungsdaten</b>	Rudolf Mildner, 15.05.2006, V 1.0
<b>Flow of Events</b>		<b>Hauptscenario</b>	Benutzer wählt eine Prüfung; Eine Sicherheitsabfrage wird angezeigt. Die Prüfung wird gelöscht.
		<b>Alternative Szenarien</b>	Der Benutzer beantwortet die Sicherheitsanfrage negativ; die Prüfung wird nicht gelöscht
		<b>Fehlersituationen</b>	Daten können aus der Datenbank nicht gelöscht werden; Es wird eine Fehlermeldung angezeigt.
		<b>Vorbedingung</b>	Das System läuft, Programm gestartet, Prüfung eingefügt
		<b>Nachbedingung</b>	Die Prüfung wurde aus der Datenbank gelöscht.

<b>Non-Functional-Constraint</b>	<b>Non-Functional-Constraints</b>	Dieser Anwendungsfall wird oft aufgerufen werden.
----------------------------------	-----------------------------------	---

### ***E/A-Diagramm***



E1) Der Benutzer wählt eine Prüfung zum Löschen aus. Dies geschieht über den Prüfungseintrag in der Infobar des gewählten Termin/Termincontainers oder über eine Liste aller Prüfungen

A1) Eine Sicherheitsabfrage wird angezeigt.

E2) Der Benutzer bestätigt den Löschvorgang durch Klicken des OK-Buttons

A2) Das System löscht die Prüfung aus der Prüfungsliste, den Eintrag in der Infobar, den Eintrag in der Datenbank und eventuell das Objekt aus dem Space. Anknüpfungen der Prüfung an Termine und Termincontainer werden gelöscht.

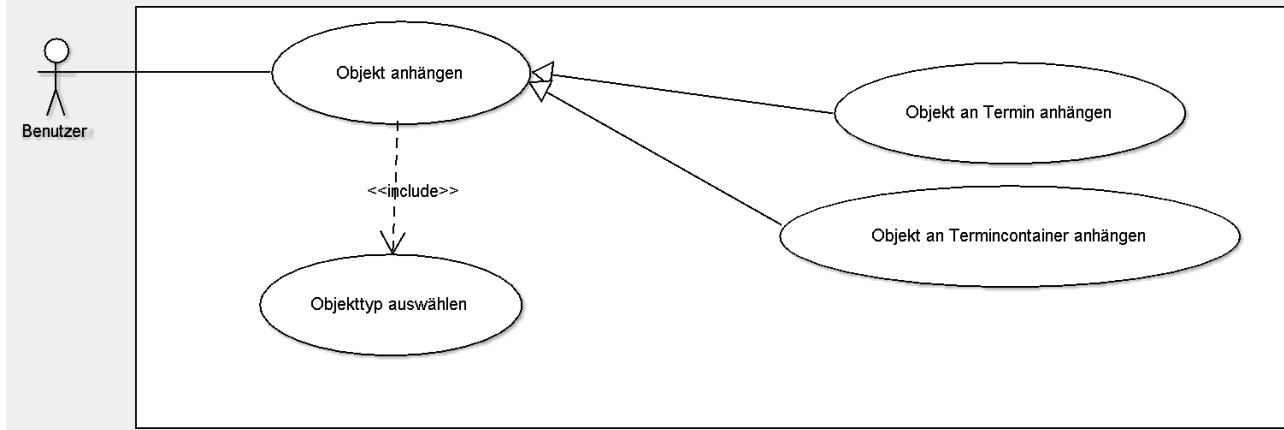
AE2) Der Benutzer bricht den Löschvorgang ab.

AE2) Der Dialog schließt sich, es werden keine Änderungen vorgenommen.



## Objekt an Termincontainer anhängen

### Anwendungsfalldiagramm

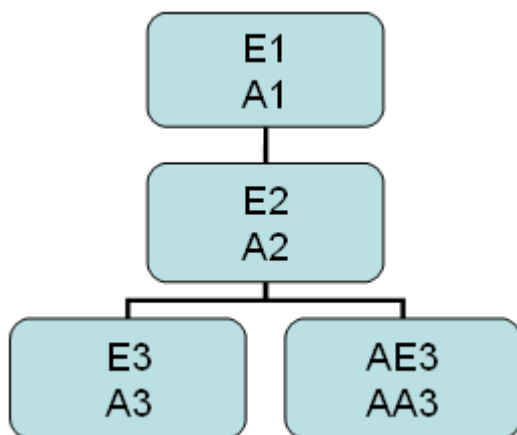


### Anwendungsfallbeschreibung

<b>Identification Summary</b>	<b>Titel</b>	Objekt an Termincontainer anhängen
	<b>Kurzbeschreibung</b>	Ein Objekt aus der Datenbank wird an einen gewählten Termincontainer angehängt
	<b>Beteiligte Aktoren</b>	Benutzer
	<b>Erstellungsdaten</b>	Gruppe 06, 15.05.2006, V 1.0
<b>Flow of Events</b>	<b>Hauptszenario</b>	Der Benutzer wählt einen Termin im Kalender aus. Dieser wird zusammen mit dem Termincontainer in der Infobar angezeigt. Dort kann der Benutzer das Bearbeitungsfenster für den Termincontainer öffnen und Objekte anhängen.
	<b>Alternative Szenarien</b>	Existiert das gewünschte Objekt nicht kann es im Bearbeitungsfenster über einen Button erstellt werden und anschließend angehängt werden.
	<b>Fehlersituationen</b>	Der im Kalender gewählte Termin ist keinem Termincontainer zugeordnet.
	<b>Vorbedingung</b>	Das System läuft, Programm gestartet, Termincontainer existent
	<b>Nachbedingung</b>	Objekt wurde an Termincontainer angehängt.

<b>Non-functional-Constraints</b>	<b>Non-Functional-Constraints</b>	An einen Termincontainer kann kein Termincontainer angehängt werden.
-----------------------------------	-----------------------------------	--

### ***E/A-Diagramm***



E1) Der Benutzer wählt einen Termin im Kalender aus und öffnet über einen Button das Bearbeitungsfenster für den zugehörigen Termincontainer  
A1) Das Bearbeitungsfenster für den Termincontainer wird angezeigt.

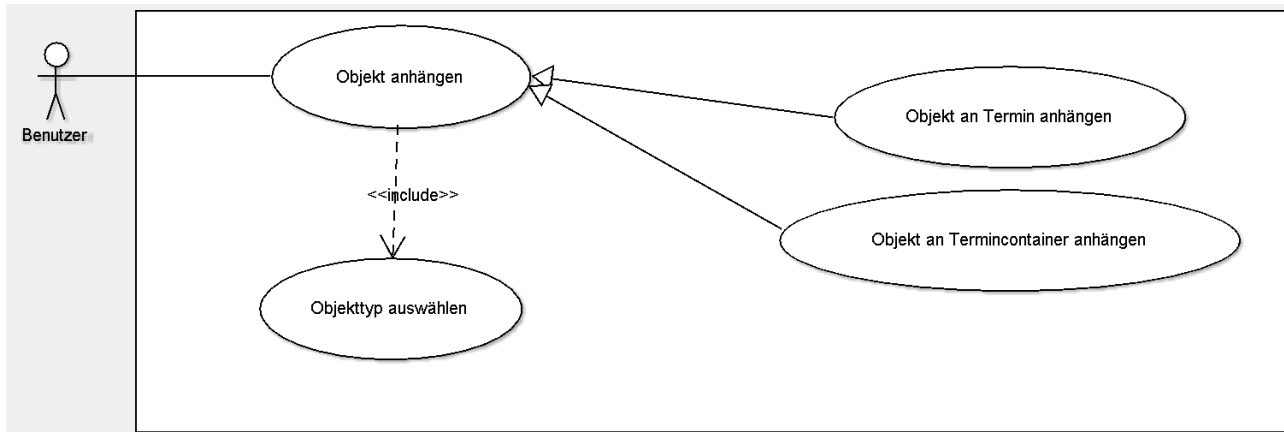
E2) Der Benutzer wählt die anzuhängende Objektart über eine Liste aus.  
A2) Das System sucht nach Objekten dieser Art.

E3) Es existieren Objekte dieser Art. Diese werden in einer Liste angezeigt. Der Benutzer wählt daraus ein Objekt aus und klickt auf den Button „Anhängen“  
A3) Das System hängt das Objekt an den gewählten Termincontainer an.

AE3) Von der gewählten Objektart gibt es keine Objekte  
AA3) Ein entsprechender Hinweis wird angezeigt.

## Objekt an Termin anhängen

### Anwendungsfalldiagramm

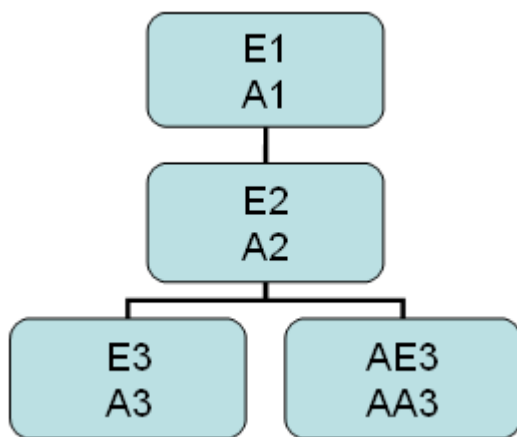


### Anwendungsfallbeschreibung

<b>Identification Summary</b>	<b>Titel</b>	Objekt an Termin anhängen
	<b>Kurzbeschreibung</b>	Ein Objekt aus der Datenbank wird an einen gewählten Termin angehängt
	<b>Beteiligte Aktoren</b>	Benutzer
	<b>Erstellungsdaten</b>	Gruppe 06, 15.05.2006, V 1.0
<b>Flow of Events</b>	<b>Hauptscenario</b>	Der Benutzer wählt einen Termin im Kalender aus. Dieser wird zusammen mit dem Termincontainer in der Infobar angezeigt. Dort kann der Benutzer das Bearbeitungsfenster für den Termin öffnen und Objekte anhängen.
	<b>Alternative Szenarien</b>	Existiert das gewünschte Objekt nicht kann es im Bearbeitungsfenster über einen Button erstellt werden und anschließend angehängt werden.
	<b>Fehlersituationen</b>	Es besteht keine Verbindung zur Datenbank.
	<b>Vorbedingung</b>	Das System läuft, Programm gestartet, Termin existent
	<b>Nachbedingung</b>	Objekt wurde an Termin angehängt.

Non-functional-Constraints	<b>Non-Functional-Constraints</b>	An einen Termin kann weder ein Termin noch ein Termincontainer angehängt werden.
----------------------------	-----------------------------------	--

### **E/A-Diagramm**



E1) Der Benutzer wählt einen Termin im Kalender aus und öffnet über einen Button das zugehörige Bearbeitungsfenster  
 A1) Das Bearbeitungsfenster für den Termin wird angezeigt.

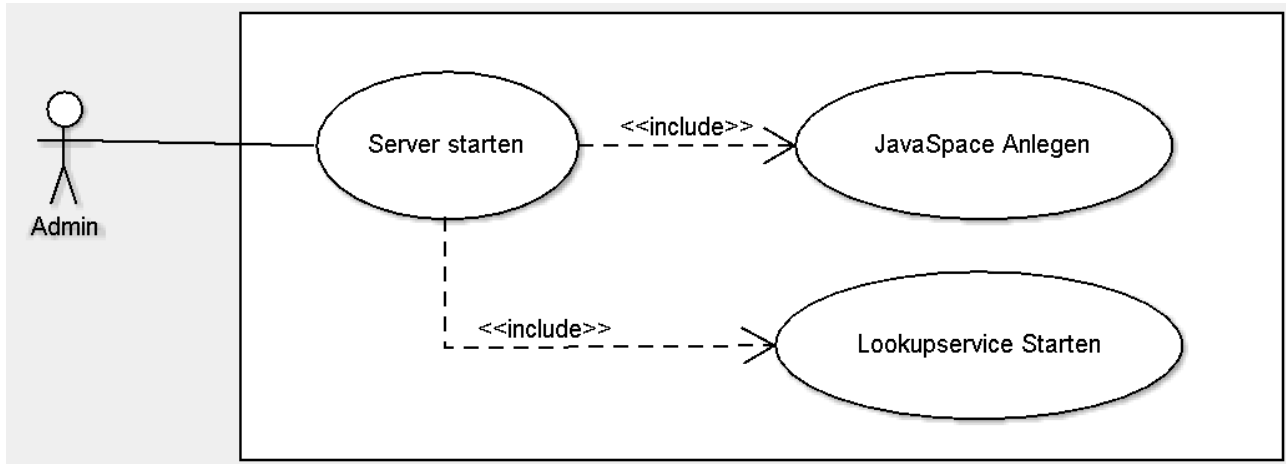
E2) Der Benutzer wählt die anzuhängende Objektart über eine Liste aus.  
 A2) Das System sucht nach Objekten dieser Art.

E3) Es existieren Objekte dieser Art. Diese werden in einer Liste angezeigt. Der Benutzer wählt daraus ein Objekt aus und klickt auf den Button „Anhängen“  
 A3) Das System hängt das Objekt an den gewählten Termin an.

AE3) Von der gewählten Objektart gibt es keine Objekte  
 AA3) Ein entsprechender Hinweis wird angezeigt.

## Server starten

### Anwendungsfalldiagramm

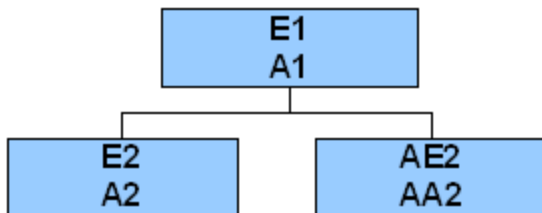


### Anwendungsfallbeschreibung

<b>Summary</b>	<b>Identificati on</b>	<b>Titel</b>	Server starten
		<b>Kurzbeschreibung</b>	Server wird gestartet, JavaSpace wird erstellt und initialisiert
		<b>Beteiligte Aktoren</b>	Server-Administrator
		<b>Erstellungsdaten</b>	Alpay Firato, 15.05.2006, V 1.0
<b>Flow of Events</b>		<b>Hauptszenario</b>	Der Server-Administrator startet den Serverprozess. Ein JavaSpace wird angelegt und beim Lookup-Service registriert. Es wird eine Erfolgsmeldung ausgegeben.
		<b>Alternative Szenarien</b>	
		<b>Fehlersituationen</b>	Falls der Server nicht gestartet bzw. der JavaSpace nicht erstellt werden kann oder beim Lookup-Service nicht registriert werden kann, wird eine Fehlermeldung ausgegeben.
		<b>Vorbedingung</b>	Das System läuft
		<b>Nachbedingung</b>	Der Server ist gestartet, Clients können über den Lookup-Service auf den Space zugreifen und ihn benutzen.

<b>Non-functional-Constraint</b>	<b>Non-Functional-Constraints</b>	Im Normalfall wird der Server nicht so oft neu gestartet.
----------------------------------	-----------------------------------	---

### ***E/A-Diagramm***



E1) Der Server-Administrator startet den Server per Command-Shell

A1) Der Server wird gestartet

E2) Der Server konnte gestartet werden

A2) Eine Erfolgsmeldung wird ausgegeben

AE2) Der Server konnte nicht erfolgreich gestartet werden

AA2) Eine Fehlermeldung wird angezeigt