

User Goal Use Cases

Template

Actors	<i>primary actors</i>	
	<i>alternative primary actors</i>	
	<i>secondary actors</i>	
Goal	<i>actor's goal</i>	
	<i>business goals</i>	
	<i>quality measurement</i>	
Business Rules		
Trigger		
Input	<i>precondition</i>	
	<i>data</i>	
Result	<i>post condition</i>	
	<i>Data</i>	
Use Case szenarios	<i>main scenario</i>	
#	<i>primary actor</i>	<i>expected system reaction</i>
	END OF SCENRAIO	
Alternative Scenarios:		

Registrierung auf der Plattform

Actors	<i>primary actors</i>	Studeininteressierter, Alumni, Professor, Mitarbeiter, unter
---------------	-----------------------	--

		Umständen auch Studenten, als jeder Stakeholder der Anwendung.
	<i>alternative primary actors</i>	keine
	<i>secondary actors</i>	keine
Goal	<i>actor's goal</i>	Der potenzielle Benutzer ist an der Anwendung interessiert und möchte sich bei dieser Anmelden.
	<i>business goals</i>	Die Anmeldung muss für den Benutzer möglichst kurz und simpel ablaufen, damit dieser seine Motivation, die Anwendung zu nutzen nicht verliert.
	<i>quality measurement</i>	Die Anzahl der Registrierungen bleibt konstant oder steigt an.
Business Rules	Registrierte User müssen bekannt sein. Datensicherheit muss gegeben sein.	Die Zugangsdaten der User werden persistent gespeichert, bei der Datenübertragung und -speicherung ist dabei auf Sicherheit zu achten.
Trigger	User hat Interesse am Benutzen der Anwendung	
Input	<i>precondition</i>	keine
	<i>data</i>	keine
Result	<i>post condition</i>	Der User ist auf der Plattform Registriert.
	<i>Data</i>	<ul style="list-style-type: none"> • Username • eMail • Passwort
Use Case szenarios	<i>main scenario</i>	
#	<i>primary actor</i>	<i>expected system reaction</i>
1	Der Benutzer öffnet die Anwendung zum ersten Mal	Das System präsentiert dem Benutzer die Möglichkeit sich anzumelden, wenn

		dieser Bereits über die nötigen Daten verfügt oder sich zu Registrieren.
2	Der Benutzer entscheidet sich, sich zu Registriren	<p>Das System präsentiert die Eingabemasken für die Benötigten Daten (Username, eMail, Passwort) und bietet eine Möglichkeit, die eingegebenen Daten abzusenden. Das System prüft ob alle Felder semantisch korrekt ausgefüllt wurden.</p> <p>IF es liegt ein Fehler vor: entsprechendes Feld markieren, auf neueingabe warten.</p>
3	Das System überprüft, ob Nutzernamen und eMail bereits im System vorkommen	<p>IF Nutzernamen und eMail liegen bereits vor: entsprechende Felder markieren, auf andere Eingabe des Nutzers warten.</p> <p>IF Nutzernamen und eMail einmalig wird der Benutzer mit seinen Daten eingeloggt.</p>
	END OF SCENRAIO	
Alternative Scenarios:	Der User bricht die Registrierung ab	Das System präsentiert die Startansicht aus Schritt #1

Anmelden mit vorhandenen Daten

Actors	<i>primary actors</i>	Studeininteressierter, Alumni, Professor,
---------------	-----------------------	---

		Mitarbeiter, unter Umständen auch Studenten, als jeder Stakeholder der Anwendung.
	<i>alternative primary actors</i>	keine
	<i>secondary actors</i>	keine
Goal	<i>actor's goal</i>	Der Registrierte Benutzer möchte sich mit seinen Daten auf der Plattform einloggen.
	<i>business goals</i>	Der Login soll dem Benutzer die Wahl lassen, ob er Benutzernamen oder eMail einträgt, um so die Wahrscheinlichkeit einer falschen Eingabe zu verringern.
	<i>quality measurement</i>	Der größte Teil der Anmeldeversuche ist nach dem ersten Versuch erfolgreich.
Business Rules	Daten müssen überprüft werden. Datensicherheit muss gegeben sein.	Die eingegebenen Daten müssen auf Validität überprüft werden. Während des gesamten Prozesses muss auf die Datensicherheit geachtet werden.
Trigger	keine Eindeutigen	
Input	<i>precondition</i>	Der Nutzer ist bereits registriert.
	<i>data</i>	<ul style="list-style-type: none"> • Username • eMail • Passwort
Result	<i>post condition</i>	Der User ist eingeloggt.
	<i>Data</i>	keine
Use Case szenarios	<i>main scenario</i>	
#	<i>primary actor</i>	<i>expected system reaction</i>

1	Der Benutzer öffnet die Anwendung und ist nicht eingeloggt.	Das System präsentiert dem Benutzer die Möglichkeit sich anzumelden, wenn dieser Bereits über die nötigen Daten verfügt oder sich zu Registrieren.
2	Der Benutzer entscheidet sich, sich anzumelden	<p>Das System präsentiert die Eingabemasken für die Benötigten Daten (Username oder eMail und Passwort), bietet eine Möglichkeit ein vergessene s Passwort zu zurück zu setzen und bietet eine Möglichkeit, die eingegebenen Daten abzusenden. Das System prüft ob alle Felder semantisch korrekt ausgefüllt wurden.</p> <p>IF es liegt ein Fehler vor: entsprechendes Feld markieren, auf Neueingabe warten.</p>
3	Das System überprüft, ob Nutzernamen/eMail bereits im System vorkommen und ob das eingegebene Passwort mit Nutzernamen/eMail übereinstimmt	<p>IF Nutzernamen/eMail liegen im System nicht vor: Entsprechendes Feld markieren und den Benutzer über fehlerhafte Daten benachrichtigen</p> <p>IF Kombination Nutzernamen/eMail und Passwort korrekt: User einloggen</p>
	END OF SCENRAIO	
Alternative Scenarios:	Der User hat sein Passwort vergessen	
1	Der User nutzt die in Schritt #2 des main scenario beschriebene Möglichkeit, sein Passwort zurück zu setzen	Das System präsentiert eine Eingabemaske für Nutzernamen/eMail des Benutzers und eine Möglichkeit, die Daten abzuschicken.

2	Das System startet eine asynchrone Verifikation des Benutzers	Sobald die Verifikation gelungen ist präsentiert das System eine Eingabemaske für das neue Passwort und eine Möglichkeit das neue Passwort abzusenden.
3	Der Benutzer ändert sein Passwort und schickt es ab	Das System setzt das neue Passwort für den entsprechenden Benutzer.

Ansehen der eigenen Benachrichtigungen

Actors	<i>primary actors</i>	Studeininteressierter, Alumni, Professor, Mitarbeiter, Studenten, also jeder Stakeholder der Anwendung.
	<i>alternative primary actors</i>	keine
	<i>secondary actors</i>	Benutzer, der für das Senden der Benachrichtigung gesorgt hat.
Goal	<i>actor's goal</i>	Der Benutzer möchte eine Übersicht haben, welche neuen Benachrichtigungen er erhalten hat und ggf. mit den für ihn Interessanten interagieren.
	<i>business goals</i>	Der Benutzer sollte nur alle für ihn nötigen Benachrichtigungen erhalten, darüber hinaus jedoch keine Zusätzlichen, um ihn nicht zu nerven und die Benachrichtigungen auf der Wichtigste zu reduzieren.
	<i>quality measurement</i>	Der Nutzer verbietet der Anwendung auf Systemebene nicht das Senden von

		Benachrichtigungen.
Business Rules	keine	keine
Trigger	Der Benutzer ist auf den Indikator aufmerksam geworden, der das Vorhandensein von ungelesenen Benachrichtigungen kommuniziert.	
Input	<i>precondition</i>	Der Nutzer ist eingeloggt
	<i>data</i>	Daten der Benachrichtigung(en)
Result	<i>post condition</i>	Der User hat mit einer Benachrichtigung interagiert
	<i>oder</i>	Alle Benachrichtigungen als "gesehen" markiert.
	<i>Data</i>	keine
Use Case szenarios	<i>main scenario</i>	
#	primary actor	expected system reaction
1	Der Benutzer wählt vom Homescreen die Ansicht für Benachrichtigungen aus	Das System präsentiert die Benachrichtigungen für den Nutzer. Das System kennzeichnet bereits gesehene, besuchte und neue Benachrichtigungen. Das System gibt eine kurze Zusammenfassung der wichtigsten Daten der Benachrichtigungen aus.
2	Der Benutzer entscheidet sich, in die Detailansicht einer Benachrichtigung zu wechseln	Das System zeigt die jeweilige Detailansicht der Benachrichtigung an. Bei tiefer Verschachtelung werden unwichtige Kinder oder Elternelemente nicht angezeigt. IF Benutzer geht einen Schritt zurück: Schritt 1, Benachrichtigung als besucht markieren.

3	Der Benutzer verlässt die Benachrichtigungsansicht	Das System zeigt den Homescreen an, alle Benachrichtigungen werden als gelesen markiert.
	END OF SCENRAIO	
Alternative Scenarios:	keine	

Ändern der Profildaten

Actors	<i>primary actors</i>	Studeininteressierter, Alumni, Professor, Mitarbeiter, Studenten, also jeder Stakeholder der Anwendung.
	<i>alternative primary actors</i>	keine
	<i>secondary actors</i>	keine
Goal	<i>actor's goal</i>	Der Benutzer möchte bestimmte Daten auf seinem Profil ändern.
	<i>business goals</i>	Die Daten müssen möglichst der Wahrheit entsprechen, bei Studenten ist explizit eine Verifizierung nötig. Beim ändern des Passwortes ist eine neue Identifizierung des Benutzer nötig.
	<i>quality measurement</i>	nicht möglich
Business Rules	keine	keine
Trigger	keine fest definierten	
Input	<i>precondition</i>	Der Nutzer ist eingeloggt
	<i>data</i>	Nutzerdaten (ID in der Datenbank)
Result	<i>post condition</i>	Der User hat seine Daten erfolgreich geändert.

	<i>Data</i>	<ul style="list-style-type: none"> • Username • eMail • Passwort • Bild • Profiltyp • (opt.) verifizierter Student • (opt.) aktueller Beruf
Use Case szenarios	<i>main scenario</i>	
#	primary actor	expected system reaction
1	Der Benutzer wählt vom Homescreen die Ansicht des eigenen Profils aus	<p>Das System präsentiert eine Ansicht mit den bisherigen Profildaten. Diese umfassen:</p> <ul style="list-style-type: none"> • Username • eMail • Passwort • Bild • Profiltyp • (opt.) verifizierter Student • (opt.) aktueller Beruf <p>Das System bietet außerdem eine Möglichkeit an, um diese Daten zu ändern.</p> <p>Weiterhin bietet das System eine Möglichkeit, sich als Student zu verifizieren.</p>
2	Der Benutzer entscheidet sich, die Daten zu ändern.	Das System präsentiert Eingabemasken für die einzelnen Informationen.
3	Der Benutzer ändert eine Information in der Eingabemaske	<p>Das System überprüft die Daten auf validität.</p> <p>IF geänderter Wert ist Passwort: Erneut aktuelles Passwort abfragen, um Benutzer eindeutig zu identifizieren.</p>
4	Der Benutzer ist mit den eingegebenen Daten zufrieden und beendet seine Eingabe.	Das System speichert die eingegebenen Daten und kehrt zu Schritt #1 zurück.

	END OF SCENRAIO	
Alternative Scenarios:	<i>User bricht Eingabe ab</i>	
1	Der User entschließt sich dazu, dass er doch keine Daten ändern möchte.	Das System kehrt zu Schritt #1 des Huptszenarios zurück , ohne die Daten zu speichern.
	END OF SCENARIO	
	<i>User will sich als Student verifizieren</i>	
1	User startet die Aktion zum verifizieren als Student.	Das System präsentiert eine Eingabmaske für die Eingabe einer Hochschul-Mailadresse
2	Der Student gibt seine Mailadresse ein	<p>Das System überprüft, ob die angegebene Adresse zu einem Hochschulnetzwerk gehört.</p> <p>IF Adresse gehört zu Hochschulnetzwerk: System validiert User, kehrt zur Hauptansicht zurück und gibt visuellen Hinweis auf Validierung geben.</p> <p>ELSE: System präsentiert Fehlermeldung</p>
	END OF SCENARIO	

Nach einer Hochshule oder einem Studiengang suchen

Actors	<i>primary actors</i>	Studeininteressierter, Alumni, Professor, Mitarbeiter, Studenten, also jeder Stakeholder der Anwendung.
	<i>alternative primary actors</i>	keine

	<i>secondary actors</i>	keine
Goal	<i>actor's goal</i>	Der Benutzer suchte nach einer Hochschule oder einem Studiengang und möchte mit entsprechenden Ergebnissen interagieren.
	<i>business goals</i>	Der User muss das gewünschte Ergebnis so schnell und einfach wie möglich angezeigt bekommen.
	<i>quality measurement</i>	kein
Business Rules	keine	keine
Trigger	keine fest definierten	
Input	<i>precondition</i>	Der Nutzer ist eingeloggt
	<i>data</i>	Nutzerdaten, Daten über vorhandene Hochschulen und Studiengänge
Result	<i>post condition</i>	Der User wählt ein passendes Ergebnis aus.
	<i>Data</i>	<ul style="list-style-type: none"> • Studiengang • Hochschule
Use Case szenarios	<i>main scenario</i>	
#	<i>primary actor</i>	<i>expected system reaction</i>
1	Benutzer öffnet die Suchfunktion	Das System präsentiert eine Eingabemaske für die Suchparameter
2	Der Benutzer gibt den Namen eines Studienganges und/oder einer Hochschule an.	Das System präsentiert eine Liste mit treffen. Ein Suchergebnis kann ein Studiengang sein (dazu wird immer auch angegeben, an welcher Hochschule dieser angeboten wird). Dem Benutzer wird außerdem eine Möglichkeit geboten, die Sortierung der Studiengänge zu ändern.
3	Der Benutzer möchte die	Das System passt die

	Sortierung ändern.	Sortierung entsprechend an.
4	Der Benutzer entscheidet sich, in die Detailansicht eines Suchergebnisses zu wechseln	Das System stellt die Detailansicht des Studienganges dar.
	END OF SCENRAIO	
Alternative Scenarios:	<i>keine</i>	

Einen neuen Thread schreiben

Actors	<i>primary actors</i>	Studeininteressierter, Alumni, Professor, Mitarbeiter, Studenten, also jeder Stakeholder der Anwendung.
	<i>alternative primary actors</i>	keine
	<i>secondary actors</i>	keine
Goal	<i>actor's goal</i>	Der Benutzer möchte einen neuen Thread in einer Bestimmten Kategorie zu einem Studiengang schreiben.
	<i>business goals</i>	Das schreiben des Threads muss für den User im Vordergrund stehen, er sollte so wenig Meta-Eingaben wie möglich tätigen müssen.
	<i>quality measurement</i>	kein
Business Rules	Der User muss berechtigt sein, den Thread in der entsprechenden Kategorie schreiben zu dürfen	Das System muss überprüfen, ob der Typ des Profils des Benutzers mit der jeweiligen Kategorie kompatibel ist.
Trigger	keine fest definierten	
Input	<i>precondition</i>	Der Nutzer ist eingeloggt. Der Benutzer befindet sich

		in der Detailansicht eines Studienganges.
	<i>data</i>	Nutzerdaten
Result	<i>post condition</i>	Der User hat einen neuen Thread verfasst
	<i>Data</i>	Inhalte des Threads
Use Case szenarios	<i>main scenario</i>	
#	<i>primary actor</i>	<i>expected system reaction</i>
1	Der Benutzer wählt die Funktion zum schreiben eines neuen Threads aus	Das System präsentiert auf Basis des Profiltyps des Nutzers die möglichen Kategorien, in denen der Benutzer einen Thread schreiben darf.
2	Der Benutzer wählt einen Threadtypen aus	Das System präsentiert eine Eingabemaske zum schreiben des Threads. IF Thread ist Erfahrungsbericht: System präsentiert eine Eingabemaske zur Bewertung (positiv oder negativ)
3	Der Benutzer kennzeichnet den Threads als "fertig"	Das System speichert den Thread, er wird veröffentlicht.
	END OF SCENRAIO	
Alternative Scenarios:	<i>keine</i>	

Eine Antwort auf einen Thread schreiben

Actors	<i>primary actors</i>	Studeininteressierter, Alumni, Professor, Mitarbeiter, Studenten, also jeder Stakeholder der Anwendung.
---------------	-----------------------	---

	<i>alternative primary actors</i>	keine
	<i>secondary actors</i>	Verfasser des Elternthreads
Goal	<i>actor's goal</i>	Der Benutzer möchte auf einen vorhandenen Thread in einer beliebigen Kategorie antworten.
	<i>business goals</i>	Das schreiben des Threads muss für den User im Vordergrund stehen. Der User sollte immer die Möglichkeit haben, das Elternelement (auf das er antwortet) ohne abbrechen des schreiben anzusehen.
	<i>quality measurement</i>	kein
Business Rules	keine	keine
Trigger	keine fest definierten	
Input	<i>precondition</i>	Der Nutzer ist eingeloggt und befindet sich in der Einzelansicht eines Threads.
	<i>data</i>	Nutzerdaten, Threaddaten
Result	<i>post condition</i>	Der User hat eine Antwort auf einen Thread verfasst.
	<i>Data</i>	Inhalte des Threads, Inhalte der Antwort
Use Case szenarios	<i>main scenario</i>	
#	<i>primary actor</i>	<i>expected system reaction</i>
1	Der Benutzer wählt an einem Thread (dieser kann auch selbst ein Kindelement eines Threads sein) die Funktion zum Antworten aus.	Das System Präsentiert eine Eingabemaske zum schreiben eines Threads.
2	Der Benutzer kennzeichnet die Antwort als "fertig"	Das System speichert die Antwort, er wird veröffentlicht.
	END OF SCENRAIO	

Alternative Scenarios:	<i>keine</i>	
-------------------------------	--------------	--

Eine Antwort oder einen Thread editieren

Actors	<i>primary actors</i>	Studeininteressierter, Alumni, Professor, Mitarbeiter, Studenten, also jeder Stakeholder der Anwendung.
	<i>alternative primary actors</i>	keine
	<i>secondary actors</i>	potentielle Verfasser von antworten auf den zu editierenden Thread
Goal	<i>actor's goal</i>	Der Benutzer möchte einen bereits von ihm geschriebenen Thread editieren (Thread umfasst auch Antworten auf Threads)
	<i>business goals</i>	Das Editieren muss Problemlos möglich sein, andere User müssen erkennen können, dass der Thread editiert wurde.
	<i>quality measurement</i>	kein
Business Rules	User dürfen nur selbst verfasste Threads ändern dürfen.	Das System muss überprüfen, ob der User mit dem verfassenden User übereinstimmt.
Trigger	Der User entdeckt einen Fehler in einem bereits geschriebenen Thread	
Input	<i>precondition</i>	Der Nutzer ist eingeloggt und befindet sich in einer Ansicht, in der er den geschriebenen Thread lesen kann.
	<i>data</i>	Nutzerdaten, Threaddaten

Result	<i>post condition</i>	Der User die Inhalte des Threads editiert
	<i>Data</i>	Geänderte Inhalte des Threads
Use Case szenarios	<i>main scenario</i>	
#	<i>primary actor</i>	<i>expected system reaction</i>
1	Der Benutzer wählt die Funktion zum editieren des Threads aus.	Das System präsentiert eine Eingabemaske mit den aktuellen Inhalten des Threads.
2	Der Benutzer kennzeichnet die Bearbeitung als "fertig"	Das System speichert den bearbeiteten Inhalt, er wird veröffentlicht. Der entsprechende Thread wird als editiert gekennzeichnet.
	END OF SCENARIO	
Alternative Scenarios:	<i>keine</i>	

Eine Antwort oder einen Thread löschen

Actors	<i>primary actors</i>	Studeninteressierter, Alumni, Professor, Mitarbeiter, Studenten, also jeder Stakeholder der Anwendung.
	<i>alternative primary actors</i>	keine
	<i>secondary actors</i>	potentielle Verfasser von Antworten auf den zu editierenden Thread
Goal	<i>actor's goal</i>	Der Benutzer möchte einen bereits von ihm geschriebenen Thread löschen (Thread umfasst auch Antworten auf Threads)
	<i>business goals</i>	Der Inhalt des Threads muss gelöscht werden, der

		Knoten muss jedoch vorhanden bleiben, damit nicht auch potentielle Kinder des Threads gelöscht werden.
	<i>quality measurement</i>	keine
Business Rules	User dürfen nur selbst verfasste Threads löschen dürfen.	Das System muss überprüfen, ob der User mit dem verfassenden User übereinstimmt.
Trigger	kein Eindeutiger	
Input	<i>precondition</i>	Der Nutzer ist eingeloggt und befindet sich in einer Ansicht, in der er den geschriebenen Thread lesen kann.
	<i>data</i>	Nutzerdaten, Threaddaten
Result	<i>post condition</i>	Der User hat den Thread gelöscht
	<i>Data</i>	keine
Use Case szenarios	<i>main scenario</i>	
#	<i>primary actor</i>	<i>expected system reaction</i>
1	Der Benutzer wählt die Funktion zum löschen des Threads aus.	Das System fragt nach einer expliziten Bestätigung zur Löschung des Threads.
2	Der Benutzer bestätigt das Löschen	Das System löscht den Thread. Der entsprechende Thread wird als gelöscht gekennzeichnet, der Knoten und alle seine Kinder bleiben jedoch vorhanden.
	END OF SCENRAIO	
Alternative Scenarios:	<i>keine</i>	

Eine Bewertung zu einem Thread abgeben

Actors	<i>primary actors</i>	Studeininteressierter, Alumni, Professor, Mitarbeiter, Studenten, also jeder Stakeholder der Anwendung.
	<i>alternative primary actors</i>	keine
	<i>secondary actors</i>	Verfasser des Threads
Goal	<i>actor's goal</i>	Der Benutzer möchte eine Bewertung zu einem Thread abgeben (positiv oder negativ)
	<i>business goals</i>	Die abgegebene Bewertung muss für den Benutzer sofort erkennbar sein.
	<i>quality measurement</i>	keine
Business Rules	keine	keine
Trigger	kein Eindeutiger	
Input	<i>precondition</i>	Der Nutzer ist eingeloggt und befindet sich in der Ansicht des Studienganges oder der Detailansicht eines Threads
	<i>data</i>	Nutzerdaten, Threaddaten
Result	<i>post condition</i>	Der Nutzer hat die Gesamtbewertung eines Threads geändert
	<i>Data</i>	Threadwertung
Use Case szenarios	<i>main scenario</i>	
#	<i>primary actor</i>	<i>expected system reaction</i>
1	Der Benutzer wählt die Funktion zur positiven oder Negativen Bewertung eines Threads aus.	Das System visualisiert die eingegangene Bewertung
	END OF SCENRAIO	
Alternative Scenarios:	<i>keine</i>	

