

Use Cases

Bei den hier verwendeten Use Cases handelt es sich um Concrete Use Cases. Es werden gezielt und konkret bestimmte Funktion und Eigenschaften adressiert und beschrieben. Im Gegensatz zu den Essential Use Cases sind diese also **nicht** technologiefrei und implementierungsunabhängig. Für die Umsetzung wurde das Template von Alistair Cockburn verwendet.

Registrierung

USE CASE	Registrieren
Scope & Level	Nutzerverwaltung, Primary Task
Preconditions	Account darf nicht schon vorhanden sein
Success End Condition	Account wird erstellt und dauerhaft auf dem Server abgespeichert
Failed End Condition	Account wird nicht erstellt
Actors	Nutzer, Server
Trigger	Nutzer möchte Inserate erstellen, seine Daten dauerhaft speichern
DESCRIPTION	1. Nutzer navigiert zur Registrierung
	2. Wählt Registrierung aus
	3. Nutzer gibt Mail Adresse ein
	4. Nutzer gibt Passwort ein
	5. Nutzer wiederholt Passwort
	6. Nutzer bestätigt Registrierung
	7. Anwendung sendet Daten an den Server
	8. Server überprüft Daten
	9. Server speichert Daten in Datenbank
	10. Server bestätigt Registrierung
	11. Anwendung informiert Nutzer über erfolgreiche Registrierung
EXTENSIONS	3a. Nutzer gibt keine gültiges Mailformat an -> Nutzer wird informiert und muss Mail erneut eingeben
	5a. Passwörter stimmen nicht überein -> Nutzer wird informiert und muss Passwort erneut eingeben
	7a. Server ist nicht erreichbar -> Nutzer wird informiert
	8a. Daten enthalten Fehler -> Server sendet Information an Anwendung, Nutzer wird informiert
	8b. Daten sind schon vorhanden -> Server sendet Information an Anwendung, Nutzer wird informiert
	9a. Datenbank ist nicht erreichbar -> Server sendet Information an Anwendung, Nutzer wird informiert

Profilinformationen aktualisieren

USE CASE	Profil aktualisieren
Scope & Level	Nutzerverwaltung, Accountverwaltung, Primary Task
Preconditions	Account ist im System vorhanden, Nutzer ist authentifiziert
Success End Condition	Nutzer hat erfolgreich sein Profil aktualisiert
Failed End Condition	Nutzer kann sein Profil nicht aktualisieren
Actors	Nutzer
Trigger	Nutzer möchte seine persönlichen Informationen aktualisieren
DESCRIPTION	1. Nutzer navigiert zu seinem Profil
	2. Nutzer wählt "Profil bearbeiten" aus
	3. Nutzer bearbeitet gewünschte Information
	4. Nutzer bestätigt die Eingabe
EXTENSIONS	3a. Eingabe entspricht nicht der geforderten Vorgabe -> Nutzer wird informiert
	4a. Server nicht erreichbar -> Nutzer wird informiert

Inserat erstellen

USE CASE	Inseraterstellung
Scope & Level	Inseratbearbeitung, Primary Task
Preconditions	Nutzer ist authentifiziert
Success End Condition	Inserat wird an den Server übermittelt und in der Datenbank gespeichert
Failed End Condition	Inserat kann nicht erstellt werden
Actors	Nutzer
Trigger	Nutzer möchte Inserat für Artikel erstellen und anderen anbieten
DESCRIPTION	1. Nutzer wählt "Inserat erstellen" aus
	2. Nutzer gibt Information zu seinem Artikel ein
	2a. Nutzer lädt Foto hoch
	2b. Nutzer beschränkt Inserat auf bestimmte Gruppe
	3. Nutzer bestätigt Eingabe
	4. Anwendung sendet Daten an den Server
	5. Server überprüft Daten
	6. Server reichert Anzeige mit internen Werten an
	7. Server speichert Daten in Datenbank
	8. Server bestätigt Erstellung
EXTENSIONS	2a. Nutzer wählt falsches Format aus -> Nutzer wird informiert
	2a. Foto kann nicht hochgeladen werden -> Nutzer wird informiert
	2b. Nutzer ist in keiner Gruppe -> Nutzer bekommt die Option nicht angezeigt
	4. Server nicht erreichbar -> Nutzer wird informiert
	6. Auf Datenbank kann nicht zugegriffen werden -> Server sendet Information an Software, Nutzer wird informiert
	7. Auf Datenbank kann nicht zugegriffen werden -> Server sendet Information an Software, Nutzer wird informiert

Inserat ändern

USE CASE	Inseratänderung
Scope & Level	Inseratbearbeitung, Primary Task
Preconditions	Insert bereits im System vorhanden
Success End Condition	Nutzer ändert Inserat
Failed End Condition	Inserat wird nicht verändert
Actors	Nutzer
Trigger	Nutzer möchte sein Inserat bearbeiten
DESCRIPTION	1. Nutzer navigiert zu "Meine Inserate"
	2. Nutzer wählt gewünschtes Inserat aus
	3. Nutzer betrachtet bestehendes Inserat
	4. Nutzer ändert gewünschte Parameter
	5. Nutzer bestätigt Änderung
	6. Anwendung sendet Daten an Server
	7. Server aktualisiert Daten in der Datenbank
EXTENSIONS	2a. Kein Inserat vorhanden -> Nutzer wird informiert
	6. Server nicht erreichbar -> Nutzer wird informiert
	7. Auf Datenbank kann nicht zugegriffen werden -> Server sendet Information an Software, Nutzer wird informiert

Inserat suchen

USE CASE	Suchen
Scope & Level	Suchen mit Filtern, Primary Task
Preconditions	Inserate müssen in der Datenbank gespeichert sein
Success End Condition	Nutzer sucht erfolgreich nach Inseraten
Failed End Condition	Nutzer kann Suche nicht erfolgreich abschließen
Actors	Nutzer
Trigger	Nutzer möchte Inserate durchsuchen
DESCRIPTION	1. Nutzer navigiert zur "Suche"
	2. Nutzer gibt Suchparameter an
	3. Nutzer bestätigt Suchanfrage
	4. Anwendung sendet Anfrage an Server
	5. Server vergleicht Daten aus der Suche mit Daten auf der Datenbank
	6. Server findet passende Daten
	7. Server sendet passende Daten zur Anwendung
	8. Anwendung stellt Daten dar
EXTENSIONS	4. Server nicht erreichbar -> Nutzer wird informiert
	5. Auf Datenbank kann nicht zugegriffen werden -> Server sendet Information an Software, Nutzer wird informiert
	6. Server findet keine passenden Daten -> Server sendet Information an Anwendung, Nutzer wird informiert

Inserat löschen

USE CASE	Inseratlöschung
Scope & Level	Inseratbearbeitung, Primary Task
Preconditions	Inserat ist in der Datenbank vorhanden
Success End Condition	Inserat wurde gelöscht
Failed End Condition	Inserat kann nicht gelöscht werden
Actors	Nutzer
Trigger	Nutzer möchte Inserat löschen
DESCRIPTION	1. Nutzer navigiert zu "Meine Inserate"
	2. Nutzer wählt gewünschtes Inserat aus
	3. Nutzer bestätigt Löschvorgang
	4. Anwendung sendet Anfrage an Server
	5. Server löscht Daten in der Datenbank
	6. Server bestätigt Löschung
EXTENSIONS	2a. Kein Inserat vorhanden -> Nutzer wird informiert
	4. Server nicht erreichbar -> Nutzer wird informiert
	5. Auf Datenbank kann nicht zugegriffen werden -> Server sendet Information an Software, Nutzer wird informiert

Inserat automatisch löschen

USE CASE#	Automatische Inseratlöschung
Scope & Level	Erinnerung, Primary Task
Preconditions	Inserat ist in der Datenbank vorhanden
Success End Condition	Inserat wurde von der Datenbank gelöscht
Failed End Condition	Inserat wurde nicht gelöscht
Actors	Server
Trigger	Inserat erreicht bestimmtes Alter
DESCRIPTION	1. Server wählt Inserat aus
	2. Server vergleicht Inserat mit Stammdaten
	3. Server löscht Inserat aus der Datenbank
	4. Server sendet Push-Benachrichtigung an den Nutzer der die Anzeige erstellte
EXTENSIONS	1. Kein Inserat vorhanden -> Server beendet Kontrolle
	2. Inseratdaten fehlerhaft -> Server speichert Fehlercode, Admin wird informiert
	2. Stammdaten fehlerhaft -> Server speichert Fehlercode, Admin wird informiert
	3. Auf Datenbank kann nicht zugegriffen werden -> Server sendet Information an Software, Nutzer wird informiert
	4. Push-Dienste nicht verfügbar -> Server versucht erneuten Vorgang nach gewisser Zeit