

# SEP Projektmappe

*PROJEKTMAPPE DES PROJEKTES*

## SemestervErwaltungsPlan

**Dokumentation des Projektes**

*Gruppe E:*

Kaan Erdogan

Andreas Häusler

Dave Makila

Johannes Morzeck

Tobias Sieber

Merlin von Rössing

## **Inhaltsverzeichnis**

Projektbeschreibung	3
Iteration I	5
User-Stories:	5
MSCs	12
UML Klassendiagramm	17
Funktionalitätsplanung	18
Iteration II	20
User-Stories	20
MSCs	26
UML Klassendiagramm	41
Funktionalitätsplanung	42
Unittests	43
Systemtests	44
Iteration III	47
User-Stories	47
MSCs	53
Funktionalitätsplanung	66
Unittests	67
Systemtests	68
Nutzerhandbuch	71
Technische Anforderungen	71
Installationsanleitung	72
Bedienungsanleitung	73

# Projektbeschreibung

## Einleitung:

Ziel ist die Implementierung eines Programms zur Verwaltung sämtlicher organisatorischer Aufgaben, die im Laufe eines Semesters zur Verwaltung des SEP anfallen. Dafür soll ein Programm entwickelt werden, in welchem die einzelnen Gruppen, Studierenden, Betreuer und sonstige organisatorische Pläne eingebettet sind, um eine reibungslose Organisation der Lehrveranstaltung zu gewährleisten.

## Iteration I:

In der ersten Iteration werden die grundlegenden Strukturen implementiert, auf deren Grundlage das Programm basiert. Wenn das Programm gestartet wird, soll der Betreuer die Möglichkeit haben, ein entsprechendes Semester auswählen zu können, zu welchem er eine Reihe an Details erhalten möchte, dabei können diese jederzeit editierbar sein. Außerdem lassen sich neue Semester anlegen und auch wieder löschen. Zu jedem Semester soll eine entsprechende Struktur von Klassen, den zu den Klassen gehörigen Gruppen, und den zu den Gruppen gehörigen Studierenden existieren. Des Weiteren soll das Programm nach der Schaffung der grundlegenden Struktur seine Daten atomar, konsistent, isoliert und dauerhaft speichern (ACID-Kriterien) und die Anwendung muss auf mehreren Clients gleichzeitig ausgeführt werden können, ohne die genannten Kriterien dabei zu verletzen.

## Iteration II:

In der zweiten Iteration soll es zu jedem Semester eine organisatorische Übersicht geben, welche den Wochenplan und den Semesterplan beinhaltet, diese beiden Pläne sollen vom Benutzer angelegt und editiert werden können, um zusätzliche Informationen anzureichern. Basierend auf dem Semesterplan soll für jede Klasse ein Anwesenheitsplan erstellt werden, in dem der Benutzer die Fehlzeiten aller Studenten minutengenau eintragen und übersichtlich einsehen kann. Zu jeder der oben genannten Entitäten (s. Iteration 1) soll der Betreuer die Möglichkeit haben, Notizen hinzufügen zu können. Notizen lassen sich in gut, mittel, schlecht und ohne Zuordnung unterscheiden, dabei unterscheiden sich diese visuell. Darüber hinaus soll dem Benutzer die Möglichkeit geboten werden, ein Benutzerkonto anlegen zu können, auf welches er mit einem zuvor selbst festgelegten Passwort zugreifen kann. Zudem sollen die Benutzerkonten und sämtliche Daten des Programms persistent gespeichert werden können.

## Iteration III:

In der letzten Iteration wird das Programm um Templates und frei wählbare Zusatzfeatures erweitert. Dem Benutzer soll es möglich sein, einen E-Maildienst mit seinem Benutzerkonto verbinden zu können, sodass er in dieser Iteration automatisiert E-Mails verschicken kann. Der Benutzer soll Templates für das Versenden von E-Mails anlegen können. Außerdem soll der Benutzer die Möglichkeit haben, direkt über das Programm einzelnen Studierenden oder ganzen Gruppen auf Grundlage der Templates E-Mails schreiben zu können. Zuletzt sollen die Gruppenmitglieder drei weitere Zusatzfeatures implementieren, welche zum Thema der Aufgabenstellung passen. Diese Zusatzfeatures sollen zuvor mit dem Betreuer abgesprochen werden.

**Mögliche Zusatzfeatures:**

- Nachrichten versenden (Chat) Registrierte Nutzer können sich gegenseitig Nachrichten senden, diese werden persistent gespeichert. Außerdem kriegt der Empfänger eine visuelle Rückmeldung, falls er eine neue Nachricht erhalten hat.
- E-Mails können mit Anhang versendet werden Beim Versenden von E-Mails können Dateien angehängt werden.
- Verknüpfung mit GitLab, um Metriken auf Gruppen-, Klassen- und Personen-Ebene zu erhalten Metriken sollen sowohl grafisch, als auch textuell angezeigt werden können.
- Favorisieren von Gruppen und Personen Es können ausgewählte Gruppen und/oder Personen favorisiert werden, welche beim Starten der Anwendung, als erstes angezeigt werden sollen.
- Datenbestände können als PDF exportieren werden Alle Datenbestände eines Semesters lassen sich als PDF-Datei exportieren.
- Zurücksetzen der Daten Dieses Zusatzfeature erlaubt es alle Datenbestände, die in der Anwendung gespeichert sind zurückzusetzen.

# Iteration I

## User-Stories:

<b>User Story-ID</b>	001
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich mich im System anmelden müssen, um Fremdzugriff auf das System zu vermeiden.
<b>Geschätzter Realisierungsaufwand</b>	4 Std.
<b>Priorität</b>	Niedrig
<b>Autor(en)</b>	Merlin von Rössing, Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	
<b>Zugehörige Szenarien</b>	bMSC_Login

<b>User Story-ID</b>	002
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich eine Benachrichtigung bekommen, wenn die von mir eingegebenen Login-Daten nicht korrekt sind, damit ich weiß, dass der Login-Vorgang fehlgeschlagen ist.
<b>Geschätzter Realisierungsaufwand</b>	1 Std.
<b>Priorität</b>	Niedrig
<b>Autor(en)</b>	Merlin von Rössing
<b>Abhängigkeiten zu anderen User Stories</b>	001
<b>Zugehörige Szenarien</b>	bMSC_Login_Feedback

<b>User Story-ID</b>	003
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich ein zentrales Fenster haben, in welchen ich alle nötigen Details von Semestern sehen kann, um eine möglichst hohe Übersichtlichkeit vorzufinden
<b>Geschätzter Realisierungsaufwand</b>	12 Std.
<b>Priorität</b>	Hoch
<b>Autor(en)</b>	Johannes Morzeck, Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	
<b>Zugehörige Szenarien</b>	bMSC_HomeView

<b>User Story-ID</b>	004
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich existierende Semester sehen können, um einen Überblick über die Semesterplanung zu bekommen.
<b>Geschätzter Realisierungsaufwand</b>	4 Std.
<b>Priorität</b>	Hoch
<b>Autor(en)</b>	Johannes Morzeck, Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	003
<b>Zugehörige Szenarien</b>	bMSC_ShowSemester

<b>User Story-ID</b>	005
<b>User Story-Beschreibung</b>	Als Betreuer möchte ein Semester auswählen können, um dieses Semester bearbeiten zu können.
<b>Geschätzter Realisierungsaufwand</b>	4 Std.
<b>Priorität</b>	Hoch
<b>Autor(en)</b>	Johannes Morzeck, Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	004
<b>Zugehörige Szenarien</b>	bMSC_EditSemester

<b>User Story-ID</b>	006
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich ein neues Semester erstellen können, um noch nichtexistierende Semester anlegen zu können.
<b>Geschätzter Realisierungsaufwand</b>	8 Std.
<b>Priorität</b>	Hoch
<b>Autor(en)</b>	Johannes Morzeck, Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	004,005
<b>Zugehörige Szenarien</b>	bMSC_CreateSemester

<b>User Story-ID</b>	007
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich Semester löschen können, um Duplikate oder falsch erstellte Semester aus dem System entfernen zu können
<b>Geschätzter Realisierungsaufwand</b>	1 Std.
<b>Priorität</b>	mittel
<b>Autor(en)</b>	Johannes Morzeck, Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	004
<b>Zugehörige Szenarien</b>	bMSC_DeleteSemester

<b>User Story-ID</b>	008
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich eine Bestätigung geben müssen, um ein Semester zu löschen, um ein versehentliches löschen zu verhindern.
<b>Geschätzter Realisierungsaufwand</b>	2 Std.
<b>Priorität</b>	Niedrig
<b>Autor(en)</b>	Johannes Morzeck, Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	007
<b>Zugehörige Szenarien</b>	bMSC_DeleteSemesterFeedback

<b>User Story-ID</b>	009
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich existierende Klassen sehen können, um einen Überblick über die Klassenplanung zu bekommen.
<b>Geschätzter Realisierungsaufwand</b>	4 Std.
<b>Priorität</b>	Hoch
<b>Autor(en)</b>	Johannes Morzeck, Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	003
<b>Zugehörige Szenarien</b>	bMSC_ShowGroupage

<b>User Story-ID</b>	010
<b>User Story-Beschreibung</b>	Als Betreuer möchte eine Klasse auswählen können, um diese Klassen weiter zu bearbeiten.
<b>Geschätzter Realisierungsaufwand</b>	2 Std.
<b>Priorität</b>	Hoch
<b>Autor(en)</b>	Johannes Morzeck, Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	009
<b>Zugehörige Szenarien</b>	bMSC_EditGroupage

<b>User Story-ID</b>	011
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich eine neue Klasse erstellen können, um noch nichtexistierende Klassen anlegen zu können.
<b>Geschätzter Realisierungsaufwand</b>	4 Std.
<b>Priorität</b>	Hoch
<b>Autor(en)</b>	Johannes Morzeck, Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	009,010
<b>Zugehörige Szenarien</b>	bMSC_CreateGroupage

<b>User Story-ID</b>	012
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich Klassen löschen können, um Duplikate oder falsch erstellte Klassen aus dem System entfernen zu können
<b>Geschätzter Realisierungsaufwand</b>	1 Std.
<b>Priorität</b>	Hoch
<b>Autor(en)</b>	Johannes Morzeck, Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	009
<b>Zugehörige Szenarien</b>	bMSC_DeleteGroupage

<b>User Story-ID</b>	013
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich eine Bestätigung geben müssen, um eine Klasse zu löschen, um ein versehentliches löschen zu verhindern.
<b>Geschätzter Realisierungsaufwand</b>	2 Std.
<b>Priorität</b>	Niedrig
<b>Autor(en)</b>	Johannes Morzeck, Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	012
<b>Zugehörige Szenarien</b>	bMSC_DeleteGroupageFeedback

<b>User Story-ID</b>	014
<b>User Story-Beschreibung</b>	Als Betreuer möchte ein Gruppen auswählen können, um diese Gruppen weiter zu bearbeiten.
<b>Geschätzter Realisierungsaufwand</b>	4 Std.
<b>Priorität</b>	Hoch
<b>Autor(en)</b>	Johannes Morzeck, Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	003
<b>Zugehörige Szenarien</b>	bMSC_EditGroup

<b>User Story-ID</b>	015
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich eine neue Gruppe erstellen können, um noch nichtexistierende Gruppe anlegen zu können.
<b>Geschätzter Realisierungsaufwand</b>	4 Std.
<b>Priorität</b>	Hoch
<b>Autor(en)</b>	Johannes Morzeck, Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	014
<b>Zugehörige Szenarien</b>	bMSC_CreateGroup



<b>User Story-ID</b>	016
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich Gruppe löschen können, um Duplikate oder falsch erstellte Gruppen aus dem System entfernen zu können
<b>Geschätzter Realisierungsaufwand</b>	1 Std.
<b>Priorität</b>	mittel
<b>Autor(en)</b>	Johannes Morzeck, Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	014
<b>Zugehörige Szenarien</b>	bMSC_DeleteGroup

<b>User Story-ID</b>	017
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich eine Bestätigung geben müssen, um eine Gruppe zu löschen, um ein versehentliches löschen zu verhindern.
<b>Geschätzter Realisierungsaufwand</b>	2 Std.
<b>Priorität</b>	Niedrig
<b>Autor(en)</b>	Johannes Morzeck, Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	016
<b>Zugehörige Szenarien</b>	bMSC_DeleteGroupFeedback

<b>User Story-ID</b>	018
<b>User Story-Beschreibung</b>	Als Betreuer möchte einen Studenten auswählen können, um diese Studenten weiter zu bearbeiten.
<b>Geschätzter Realisierungsaufwand</b>	2 Std.
<b>Priorität</b>	Hoch
<b>Autor(en)</b>	Johannes Morzeck, Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	003
<b>Zugehörige Szenarien</b>	bMSC_EditStudent

<b>User Story-ID</b>	019
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich einen neuen Studenten erstellen können, um noch nichtexistierende Studenten anlegen zu können.
<b>Geschätzter Realisierungsaufwand</b>	4 Std.
<b>Priorität</b>	Hoch
<b>Autor(en)</b>	Johannes Morzeck, Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	018
<b>Zugehörige Szenarien</b>	bMSC_CreateStudent

<b>User Story-ID</b>	020
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich Studenten löschen können, um Duplikate oder falsch erstellte Studenten aus dem System entfernen zu können
<b>Geschätzter Realisierungsaufwand</b>	1 Std.
<b>Priorität</b>	mittel
<b>Autor(en)</b>	Johannes Morzeck, Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	018
<b>Zugehörige Szenarien</b>	bMSC_DeleteStudent

<b>User Story-ID</b>	021
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich eine Bestätigung geben müssen, um einen Studenten zu löschen, um ein versehentliches löschen zu verhindern.
<b>Geschätzter Realisierungsaufwand</b>	2 Std.
<b>Priorität</b>	Niedrig
<b>Autor(en)</b>	Johannes Morzeck, Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	020
<b>Zugehörige Szenarien</b>	bMSC_DeleteStudentFeedback

<b>User Story-ID</b>	022
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich Studenten zu bestimmten Gruppen hinzufügen können, damit der Student der zugehörigen Gruppe zugeordnet werden kann.
<b>Geschätzter Realisierungsaufwand</b>	4 Std.
<b>Priorität</b>	Hoch
<b>Autor(en)</b>	Johannes Morzeck, Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	
<b>Zugehörige Szenarien</b>	bMSC_StudentInGroup

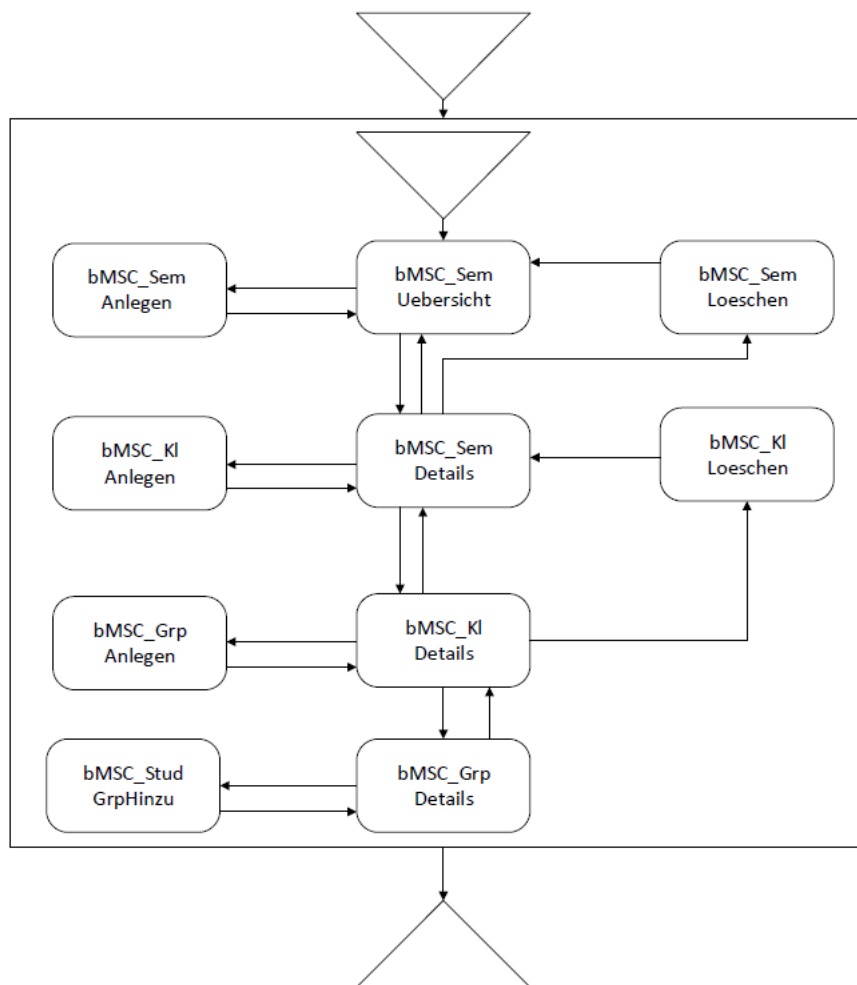
<b>User Story-ID</b>	023
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich Gruppen zu bestimmten Klassen hinzufügen können, damit eine Gruppe der zugehörigen Klasse zugeordnet werden kann.
<b>Geschätzter Realisierungsaufwand</b>	4 Std.
<b>Priorität</b>	mittel
<b>Autor(en)</b>	Johannes Morzeck, Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	
<b>Zugehörige Szenarien</b>	bMSC_GroupeInCGroupage

<b>User Story-ID</b>	024
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich Klassen zu bestimmten Semestern hinzufügen können, damit eine Klasse dem zugehörigen Semester zugeordnet werden kann.
<b>Geschätzter Realisierungsaufwand</b>	4 Std.
<b>Priorität</b>	mittel
<b>Autor(en)</b>	Johannes Morzeck, Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	
<b>Zugehörige Szenarien</b>	bMSC_GroupageInSemester

<b>User Story-ID</b>	025
<b>User Story-Beschreibung</b>	Als Benutzer möchte ich, dass die Anwendung auf mehreren Clients ausgeführt werden kann, ohne dass dabei Dateninkonsistenzen auftreten, damit bei Parallelbetrieb keine Dateninkonsistenzen auftreten.
<b>Geschätzter Realisierungsaufwand</b>	1 Std.
<b>Priorität</b>	Hoch
<b>Autor(en)</b>	Johannes Morzeck, Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	012,002
<b>Zugehörige Szenarien</b>	bMSC_Concurrency

## MSCs

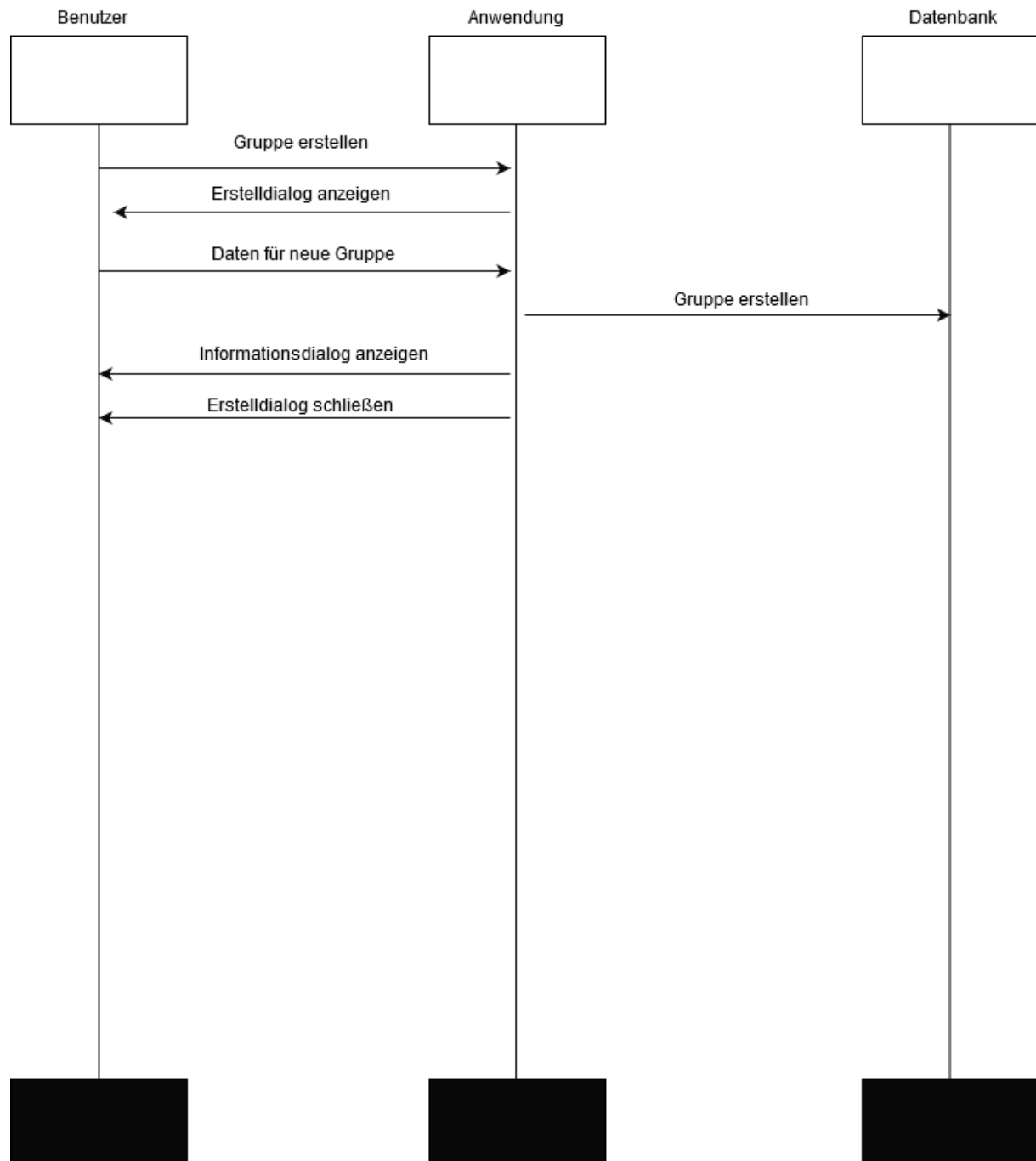
hMSC:



bMSC's:

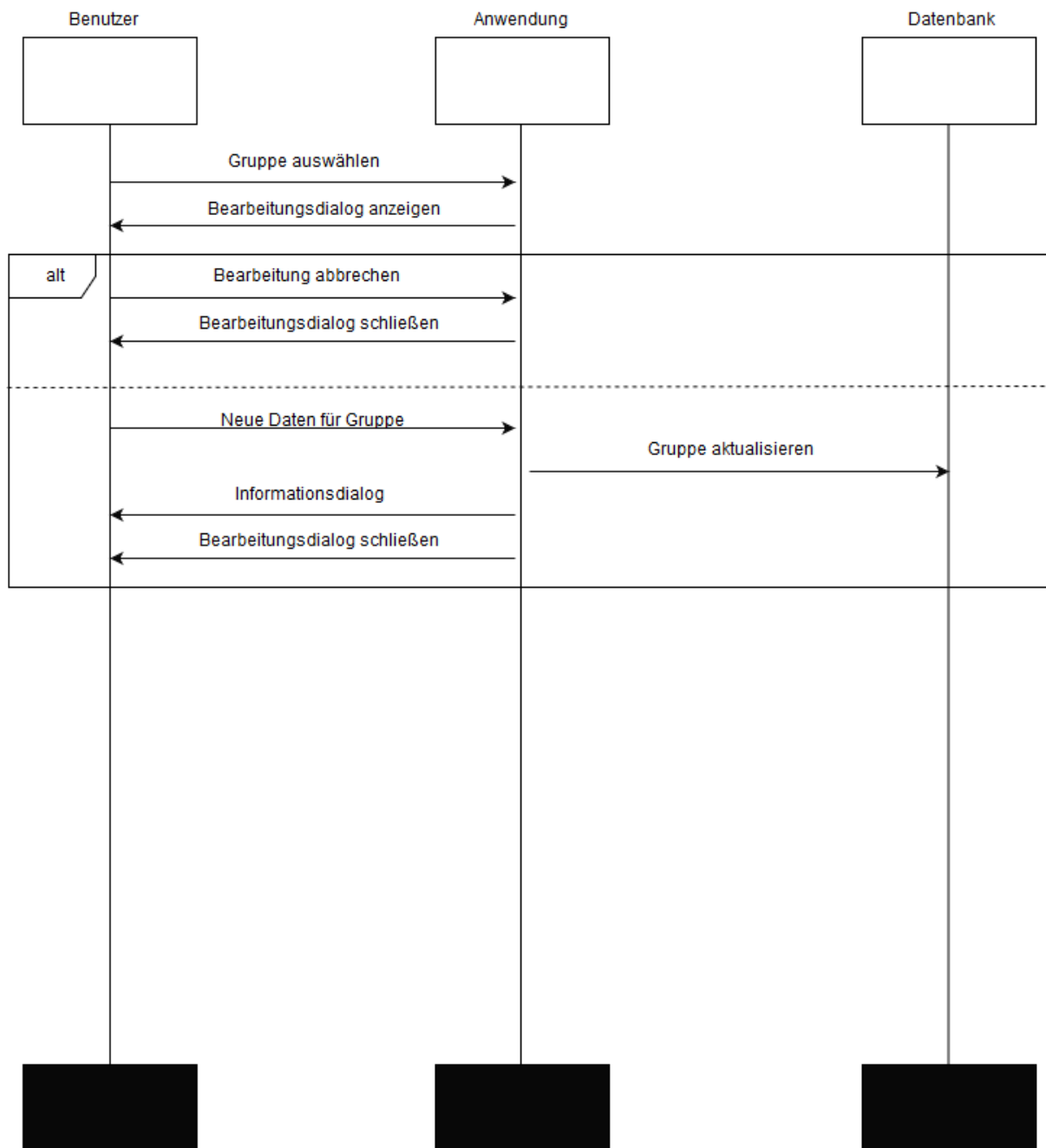
### Erstellen einer Gruppe

Dies ist repräsentativ für das Erstellen eines Studenten, einer Klasse und eines Semesters, nur die jeweiligen Attribute ändern sich etwas.



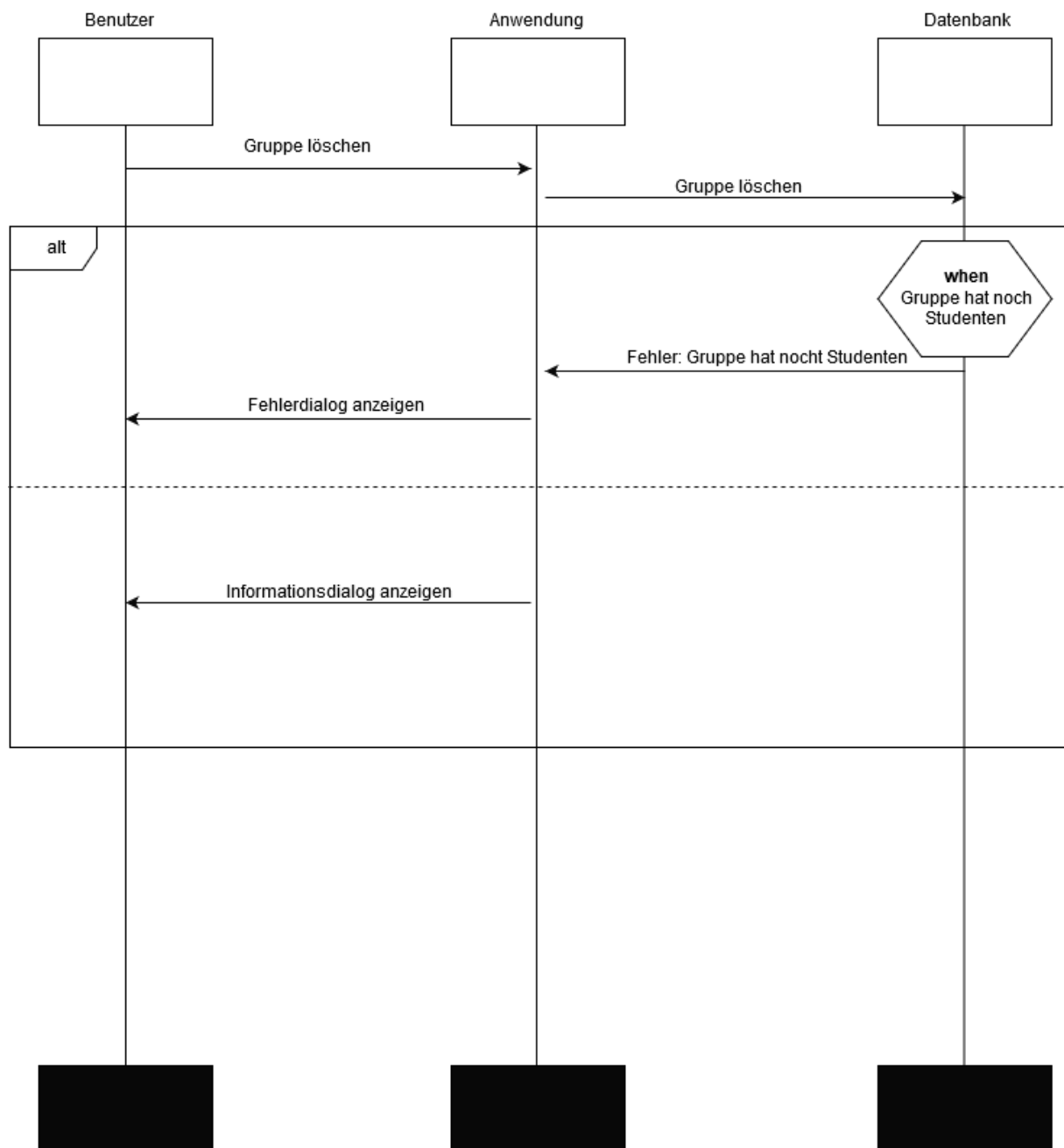
## Editieren einer Gruppe

Dies ist repräsentativ für das Editieren eines Studenten, einer Klasse und eines Semesters, nur die jeweiligen Attribute ändern sich etwas.

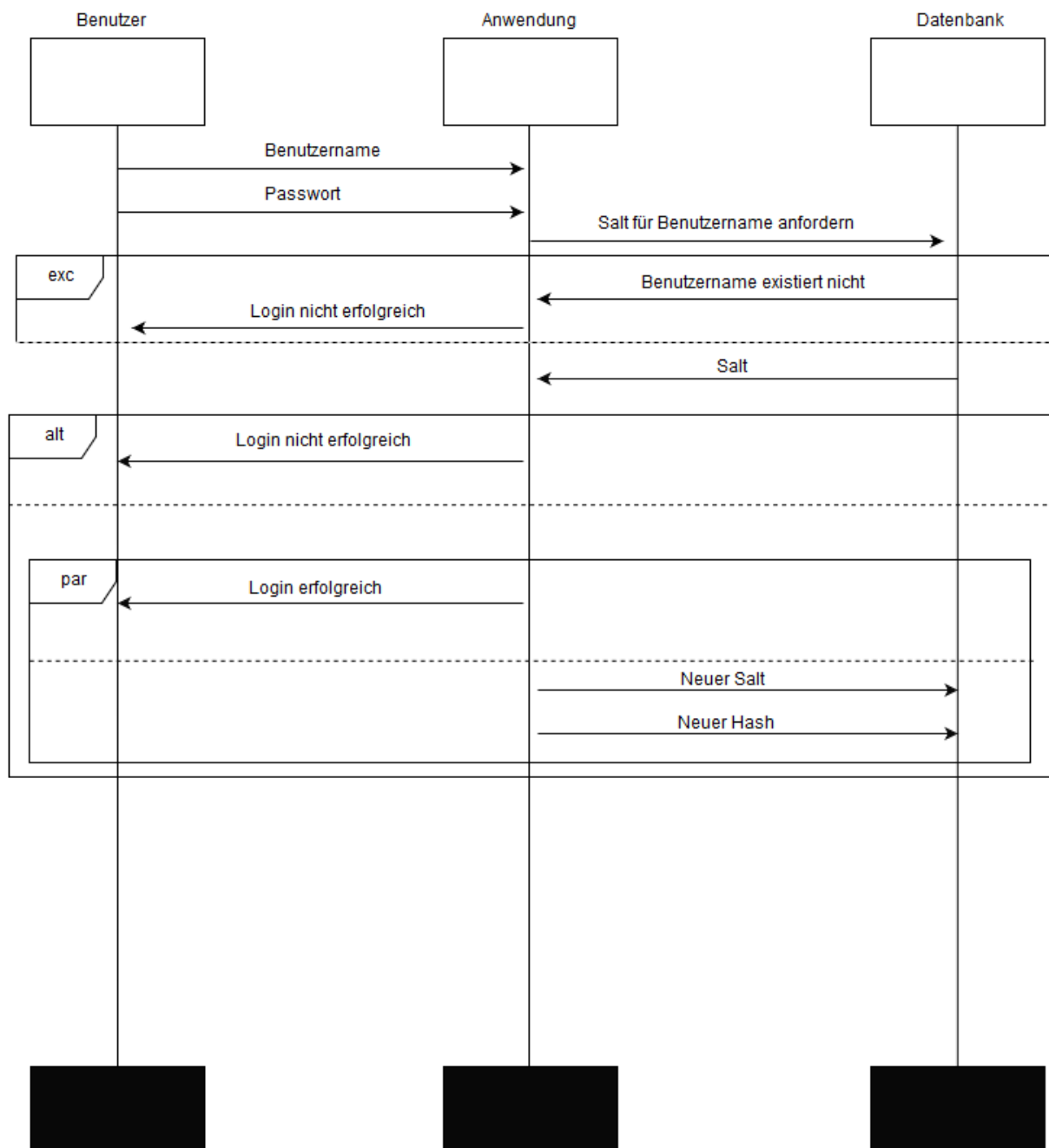


## Löschen einer Gruppe

Dies ist repräsentativ für das Löschen eines Studenten, einer Klasse und eines Semesters.

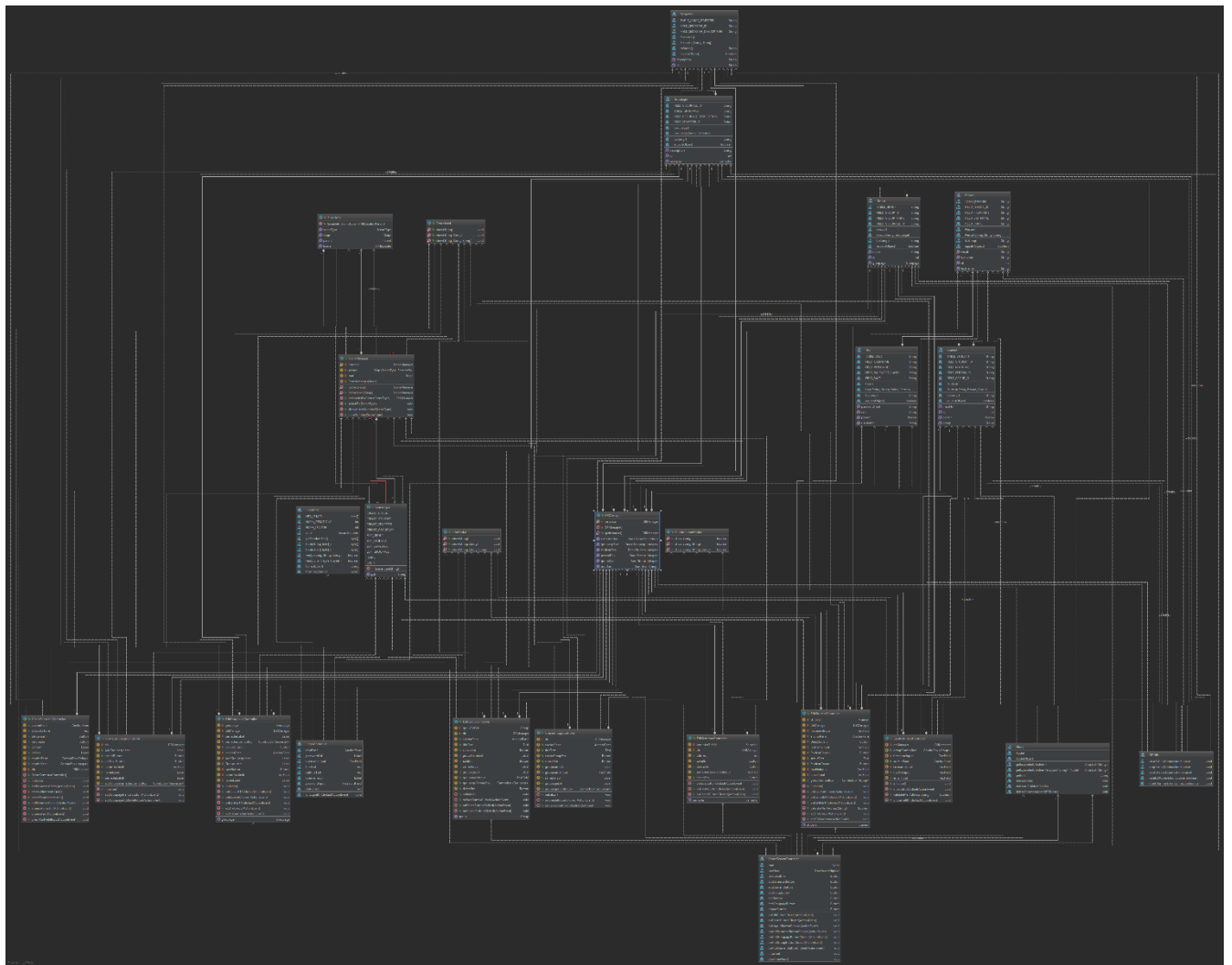


Login repräsentiert das einloggen in das Programm.





## UML Klassendiagramm



## Funktionalitätsplanung

Id	Funktionalität	Geschätzter Aufwand	Verantwortlicher	Abhängige Funktionalitäten	Quellcode-referenz	Status
<b>1.</b>	<b>Dokumentation</b>					
1.1	Erstellen der User Stories	3h	Johannes, Tobias	bMSCs, hMSC	N.A.	fertig
1.2	Erstellen des hMSCs	4h	Andreas	bMSCs	N.A.	fertig
1.3	Erstellen der bMSCs	5h	Kaan, Alisan, Merlin, Tobias, Dave	Klassendiagramm	N.A.	fertig
1.4	Erstellen des Klassendiagramms	2h	Merlin, Tobias	N.A.	N.A.	fertig
1.5	Funktionalitätsplanung	1h	Merlin, Tobias	N.A.	N.A.	fertig
1.6	Erstellen der Präsentation	2h	Johannes, Tobias	N.A.	N.A.	
<b>2.</b>	<b>Datenbank</b>					
2.1	Zugriff auf die Datenbank	3h	Merlin, Alisan	Modelle	DBManager	fertig
2.2	Erstellen einer Datenbank	1h	Merlin	DBManager, Modelle	DBUtils	fertig
<b>3.</b>	<b>Controller</b>					
3.1	Erstellen eines Semesters	5h	Dave, Merlin, Johannes, Tobias	SemesterModel	CreateSemesterController	fertig
3.2	Editieren eines Semesters	1h	Johannes	SemesterModel	EditSemesterController	fertig
3.3	Erstellen einer Klasse	5h	Kaan, Merlin, Johannes, Tobias	GroupageModel	CreateGroupageController	fertig
3.4	Editieren einer Klasse	1h	Tobias	GroupageModel	EditGroupageController	fertig
3.5	Erstellen einer Gruppe	4h	Johannes	GroupModel	CreateGroupController	fertig
3.6	Editieren einer Gruppe	4h	Johannes	GroupModel	EditGroupController	fertig
3.7	Erstellen eines Studenten	5h	Tobias	StudentModel	CreateStudentController	fertig
3.8	Editieren eines Studenten	4h	Tobias	StudentModel	EditStudentController	fertig
3.9	Einloggen in das Programm	2h	Merlin	HashUtils, UserModel	LoginController	fertig
3.10	Übersicht und dient zur Öffnung der anderen Controller	30h	Andreas	Alle Models	HomeViewController	fertig

<b>4.</b>	<b>Models</b>					
4.1	Schnittstelle Java/Datenbank	1,5h	Merlin	N.A.	SemesterModel	fertig
4.2	Repräsentation der Klasse aus der Datenbank	1,5h	Merlin	N.A.	GroupageModel	fertig
4.3	Schnittstelle Java/Datenbank	1,5h	Merlin	N.A.	GroupModel	fertig
4.4	Schnittstelle Java/Datenbank	1,5h	Merlin	N.A.	StudentModel	fertig
4.5	Schnittstelle Java/Datenbank	1,5h	Merlin	N.A.	PersonModel	fertig
4.6	Schnittstelle Java/Datenbank	1,5h	Merlin	N.A.	UserModel	fertig
<b>5.</b>	<b>Utils</b>					
5.1	Zugriff auf die Datenbank	0,5h	Merlin	DBManager	DBUtils	fertig
5.2	Verschlüsselung der Passwörter	2,5h	Merlin	DBManager	HashUtils	fertig
5.3	Fensterverwaltung	1,5h	Merlin	N.A.	SceneManager	fertig
5.4	Start des Programms	0,1h	Merlin	SEP	Launcher	fertig
5.5	Main Klasse	0,1h	Merlin	N.A.	SEP	fertig

# Iteration II

## User-Stories

<b>User Story-ID</b>	026
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich mich im System registrieren müssen, um die Funktionalitäten der Software verwenden zu können.
<b>Geschätzter Realisierungsaufwand</b>	10 Std.
<b>Priorität</b>	Hoch
<b>Autor(en)</b>	Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	
<b>Zugehörige Szenarien</b>	bMSC_Register

<b>User Story-ID</b>	027
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich eine Übersicht vorfinden, welche Betreuer im System angemeldet sind, um Betreuer besser verwalten zu können.
<b>Geschätzter Realisierungsaufwand</b>	8 Std.
<b>Priorität</b>	Hoch
<b>Autor(en)</b>	Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	
<b>Zugehörige Szenarien</b>	bMSC_User_Admin

<b>User Story-ID</b>	028
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich existierende Betreuer löschen können, um sicherzustellen, dass sich nur gewünschte Personen anmelden können.
<b>Geschätzter Realisierungsaufwand</b>	10 Std.
<b>Priorität</b>	Hoch
<b>Autor(en)</b>	Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	027
<b>Zugehörige Szenarien</b>	bMSC_UserDelete

<b>User Story-ID</b>	029
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich, dass sich Betreuer bei der Registrierung mit einem Einwahlschlüssel registrieren müssen, um zu verhindern, dass sich Unbefugte nicht anmelden können.
<b>Geschätzter Realisierungsaufwand</b>	5 Std.
<b>Priorität</b>	Hoch
<b>Autor(en)</b>	Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	026, 030
<b>Zugehörige Szenarien</b>	bMSC_InviteCodeRegister

<b>User Story-ID</b>	030
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich einen einzigartigen Einwahlschlüssel erzeugen können, damit sich Betreuer damit registrieren können.
<b>Geschätzter Realisierungsaufwand</b>	7 Std.
<b>Priorität</b>	Hoch
<b>Autor(en)</b>	Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	026, 029
<b>Zugehörige Szenarien</b>	bMSC_CreateInviteCode

<b>User Story-ID</b>	031
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich die Daten von Betreuern ändern können, um diese möglichst aktuell zu halten.
<b>Geschätzter Realisierungsaufwand</b>	4 Std.
<b>Priorität</b>	Hoch
<b>Autor(en)</b>	Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	027
<b>Zugehörige Szenarien</b>	bMSC_EditUser

<b>User Story-ID</b>	032
<b>User Story-Beschreibung</b>	Als Benutzer möchte ich meine letzte Sitzung wiederherstellen, können um ohne Verzögerung weiterarbeiten zu können.
<b>Geschätzter Realisierungsaufwand</b>	6 Std.
<b>Priorität</b>	Gering
<b>Autor(en)</b>	Andreas Häusler
<b>Abhängigkeiten zu anderen User Stories</b>	keine
<b>Zugehörige Szenarien</b>	bMSC_RestoreSession

<b>User Story-ID</b>	033
<b>User Story-Beschreibung</b>	Als Benutzer möchte ich zu jedem Semester einen Wochenplan anlegen können, um zeitliche Abläufe innerhalb einer Woche zu koordinieren.
<b>Geschätzter Realisierungsaufwand</b>	6 Std.
<b>Priorität</b>	Hoch
<b>Autor(en)</b>	Andreas Häusler
<b>Abhängigkeiten zu anderen User Stories</b>	keine
<b>Zugehörige Szenarien</b>	bMSC_CreateWeekPlan

<b>User Story-ID</b>	034
<b>User Story-Beschreibung</b>	Als Benutzer möchte ich vorhandene Wochenpläne bearbeiten können, um Informationen aktuell zu halten.
<b>Geschätzter Realisierungsaufwand</b>	6 Std.
<b>Priorität</b>	Hoch
<b>Autor(en)</b>	Andreas Häusler
<b>Abhängigkeiten zu anderen User Stories</b>	033
<b>Zugehörige Szenarien</b>	bMSC_EditWeekPlan

<b>User Story-ID</b>	035
<b>User Story-Beschreibung</b>	Als Benutzer möchte ich zu jedem Semester einen Semesterplan anlegen können, um zeitliche Abläufe innerhalb eines Semesters zu koordinieren.
<b>Geschätzter Realisierungsaufwand</b>	6 Std.
<b>Priorität</b>	Hoch
<b>Autor(en)</b>	Andreas Häusler
<b>Abhängigkeiten zu anderen User Stories</b>	keine
<b>Zugehörige Szenarien</b>	bMSC_CreateSemesterPlan

<b>User Story-ID</b>	036
<b>User Story-Beschreibung</b>	Als Benutzer möchte ich vorhandene Semesterpläne bearbeiten können, um Informationen aktuell zu halten.
<b>Geschätzter Realisierungsaufwand</b>	6 Std.
<b>Priorität</b>	Hoch
<b>Autor(en)</b>	Andreas Häusler
<b>Abhängigkeiten zu anderen User Stories</b>	035
<b>Zugehörige Szenarien</b>	bMSC_EditSemesterPlan

<b>User Story-ID</b>	037
<b>User Story-Beschreibung</b>	Als Benutzer möchte ich ausgehend vom Semesterplan für jede Klasse einen Anwesenheitsplan erstellen können, um die Fehlzeiten aller Studenten minutengenau festzuhalten.
<b>Geschätzter Realisierungsaufwand</b>	6 Std.
<b>Priorität</b>	Hoch
<b>Autor(en)</b>	Andreas Häusler
<b>Abhängigkeiten zu anderen User Stories</b>	keine
<b>Zugehörige Szenarien</b>	bMSC_CreateTardyPlan

<b>User Story-ID</b>	039
----------------------	-----

<b>User Story-Beschreibung</b>	Als Benutzer möchte ich für jedes Semester, jede Klasse, jede Gruppe und jeden Studenten Notizen hinzufügen können, um flexibel Informationen zu ergänzen.
<b>Geschätzter Realisierungsaufwand</b>	6 Std.
<b>Priorität</b>	Hoch
<b>Autor(en)</b>	Andreas Häusler
<b>Abhängigkeiten zu anderen User Stories</b>	keine
<b>Zugehörige Szenarien</b>	bMSC_NotesList

<b>User Story-ID</b>	040
<b>User Story-Beschreibung</b>	Als Benutzer möchte ich Notizen nach gut, mittel, schlecht und nicht zugeordnet klassifizieren können, um sie ihrer Bedeutung nach gewichten zu können.
<b>Geschätzter Realisierungsaufwand</b>	6 Std.
<b>Priorität</b>	Hoch
<b>Autor(en)</b>	Andreas Häusler
<b>Abhängigkeiten zu anderen User Stories</b>	039
<b>Zugehörige Szenarien</b>	bMSC_NotesList

<b>User Story-ID</b>	038
<b>User Story-Beschreibung</b>	Als Benutzer möchte ich vorhandene Anwesenheitspläne bearbeiten können, um Informationen aktuell zu halten.
<b>Geschätzter Realisierungsaufwand</b>	6 Std.
<b>Priorität</b>	Hoch
<b>Autor(en)</b>	Andreas Häusler
<b>Abhängigkeiten zu anderen User Stories</b>	037
<b>Zugehörige Szenarien</b>	bMSC_EddiTardyPlan

<b>User Story-ID</b>	041
<b>User Story-Beschreibung</b>	Als Benutzer möchte ich die Klassifikation der Notizen visuell unterscheiden können, um mich schneller zu orientieren.
<b>Geschätzter Realisierungsaufwand</b>	6 Std.
<b>Priorität</b>	Hoch
<b>Autor(en)</b>	Andreas Häusler
<b>Abhängigkeiten zu anderen User Stories</b>	039, 040
<b>Zugehörige Szenarien</b>	bMSC_NotesList

<b>User Story-ID</b>	042
<b>User Story-Beschreibung</b>	Als Benutzer möchte ich vorhandene Notizen bearbeiten können, um Informationen aktuell zu halten.
<b>Geschätzter Realisierungsaufwand</b>	6 Std.
<b>Priorität</b>	Hoch
<b>Autor(en)</b>	Andreas Häusler
<b>Abhängigkeiten zu anderen User Stories</b>	039
<b>Zugehörige Szenarien</b>	bMSC_NotesList

<b>User Story-ID</b>	043
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich Chat-Nachrichten an andere Benutzer senden können.
<b>Geschätzter Realisierungsaufwand</b>	8 Std.
<b>Priorität</b>	Mittel
<b>Autor(en)</b>	Johannes Morzeck
<b>Abhängigkeiten zu anderen User Stories</b>	keine
<b>Zugehörige Szenarien</b>	bMSC_Nachricht_senden

<b>User Story-ID</b>	044
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich Chat-Nachrichten von anderen Benutzern empfangen können
<b>Geschätzter Realisierungsaufwand</b>	8 Std.
<b>Priorität</b>	Mittel
<b>Autor(en)</b>	Johannes Morzeck
<b>Abhängigkeiten zu anderen User Stories</b>	043
<b>Zugehörige Szenarien</b>	bMSC_Nachricht_empfangen

<b>User Story-ID</b>	045
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich eine visuelle Benachrichtigung bekommen, wenn ich eine Nachricht über den Chat-Dienst erhalten habe.
<b>Geschätzter Realisierungsaufwand</b>	5 Std.
<b>Priorität</b>	Mittel
<b>Autor(en)</b>	Johannes Morzeck
<b>Abhängigkeiten zu anderen User Stories</b>	044
<b>Zugehörige Szenarien</b>	bMSC_Nachricht_empfangen

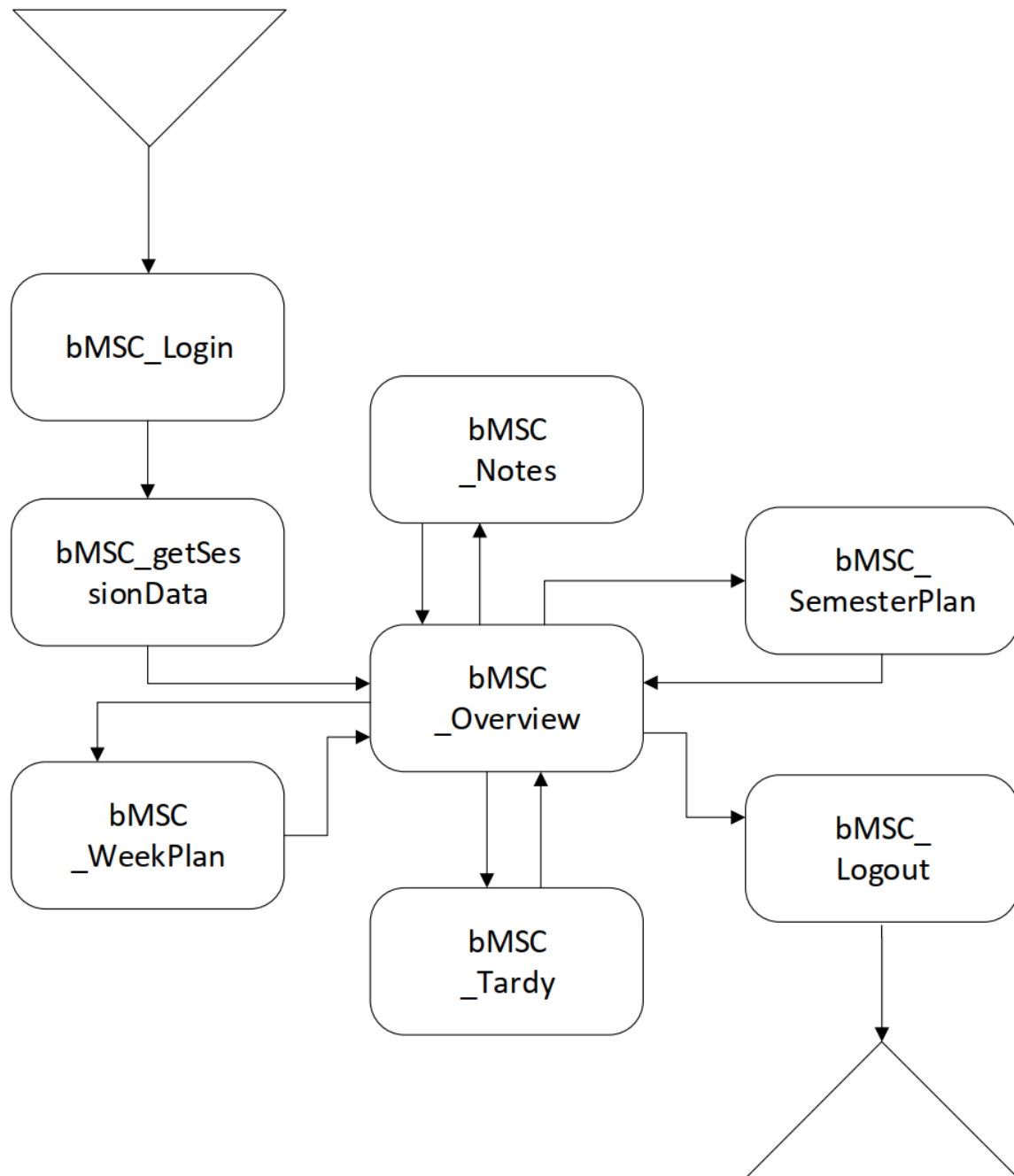


<b>User Story-ID</b>	046
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich zu jedem Semester einen Semesterplan anlegen können, um zeitliche Abläufe innerhalb eines Semesters zu koordinieren.
<b>Geschätzter Realisierungsaufwand</b>	2 Std.
<b>Priorität</b>	Hoch
<b>Autor(en)</b>	Dave Makila
<b>Abhängigkeiten zu anderen User Stories</b>	004,005
<b>Zugehörige Szenarien</b>	bMSC_CreateTermpplan

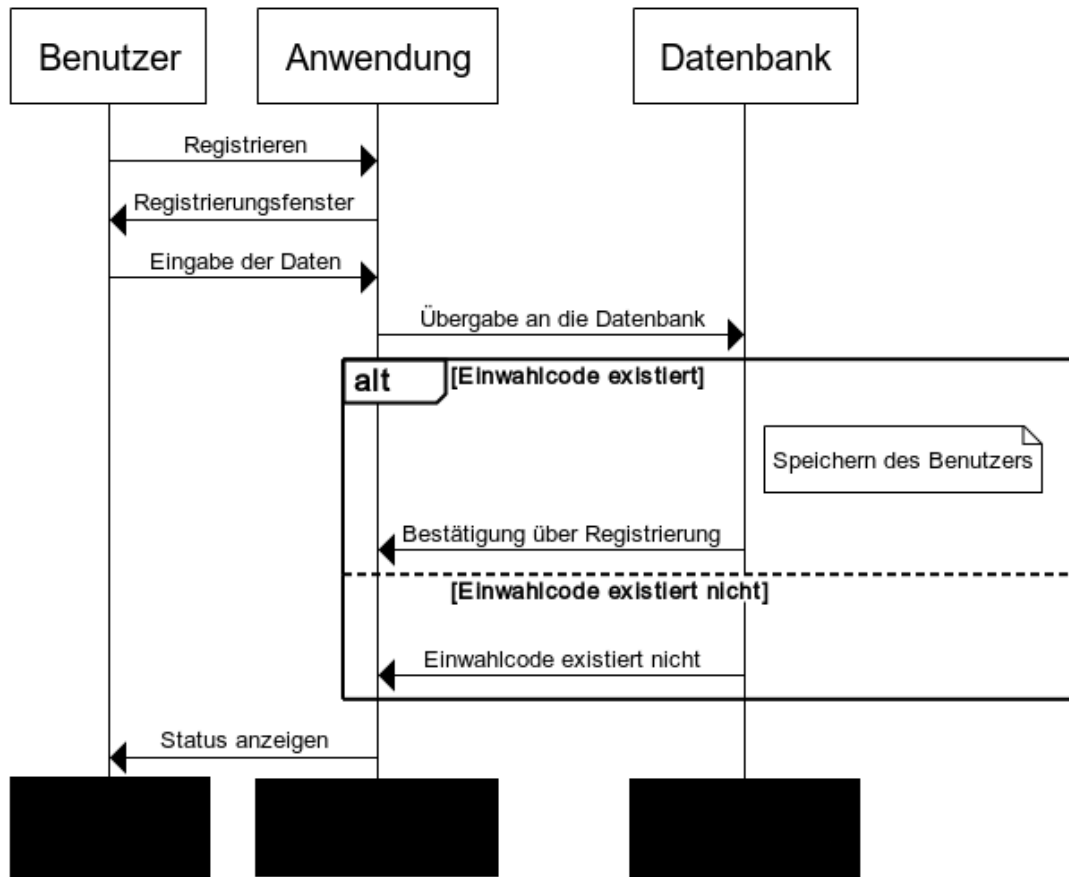
<b>User Story-ID</b>	047
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich vorhandene Semesterpläne bearbeiten können, um Informationen aktuell zu halten.
<b>Geschätzter Realisierungsaufwand</b>	2 Std.
<b>Priorität</b>	Mittel
<b>Autor(en)</b>	Dave Makila
<b>Abhängigkeiten zu anderen User Stories</b>	044
<b>Zugehörige Szenarien</b>	bMSC_EditTermpplan

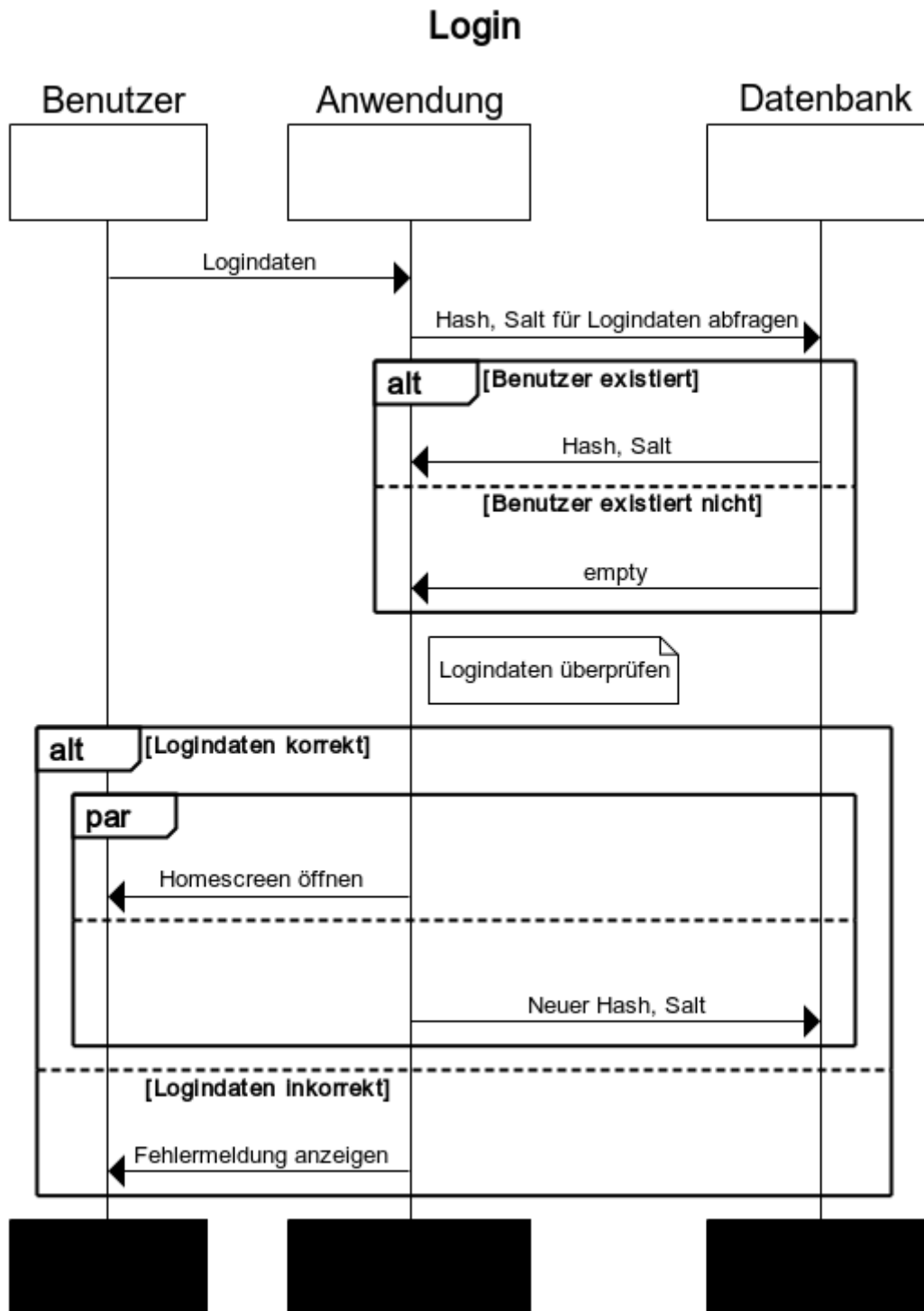
## MSCs

### hMSC\_2

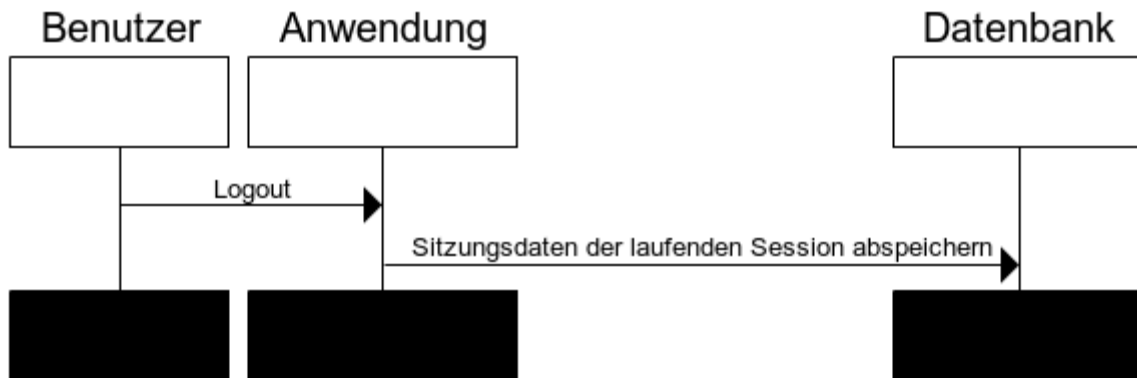


## Registrieren

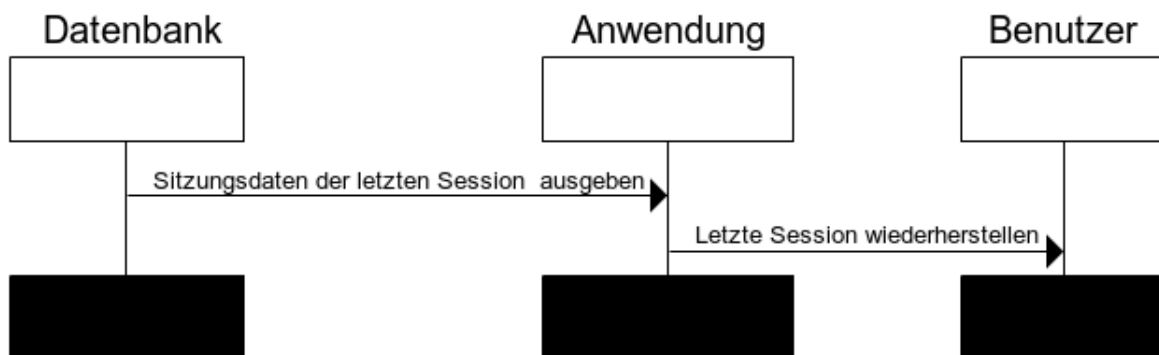




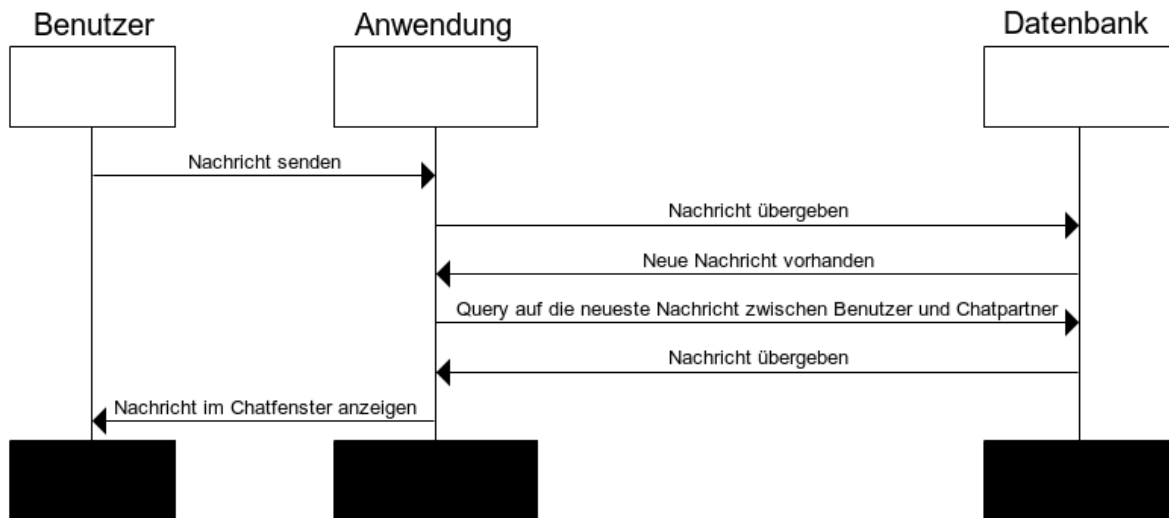
## Session abspeichern



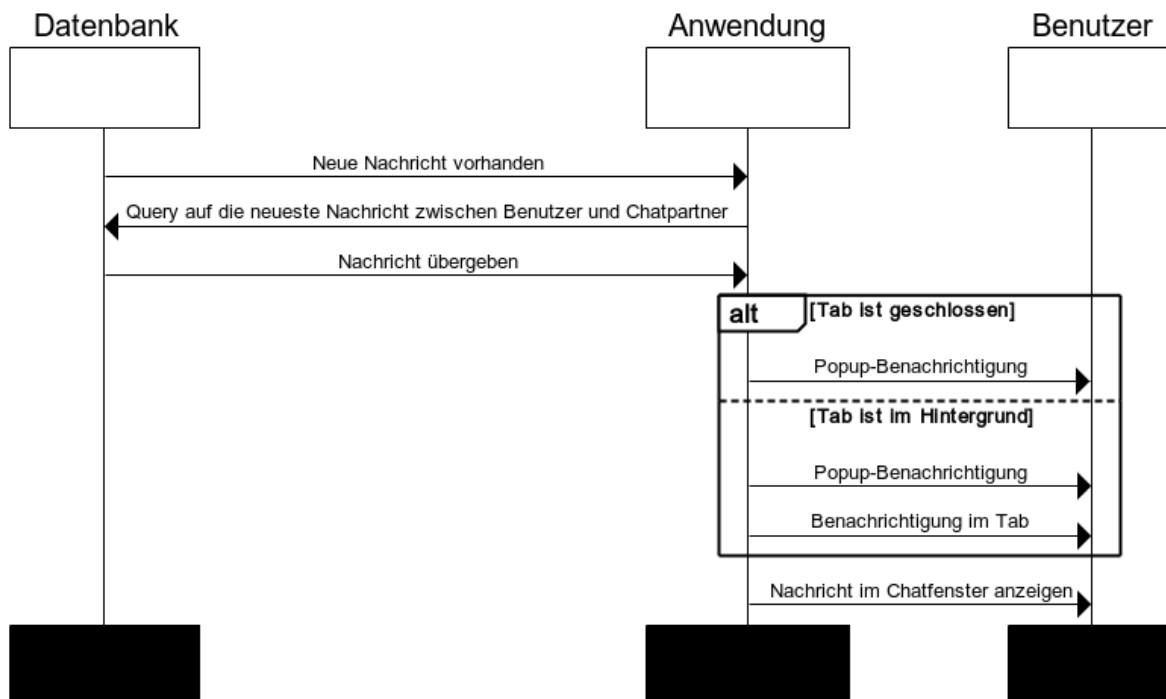
## Session wiederherstellen



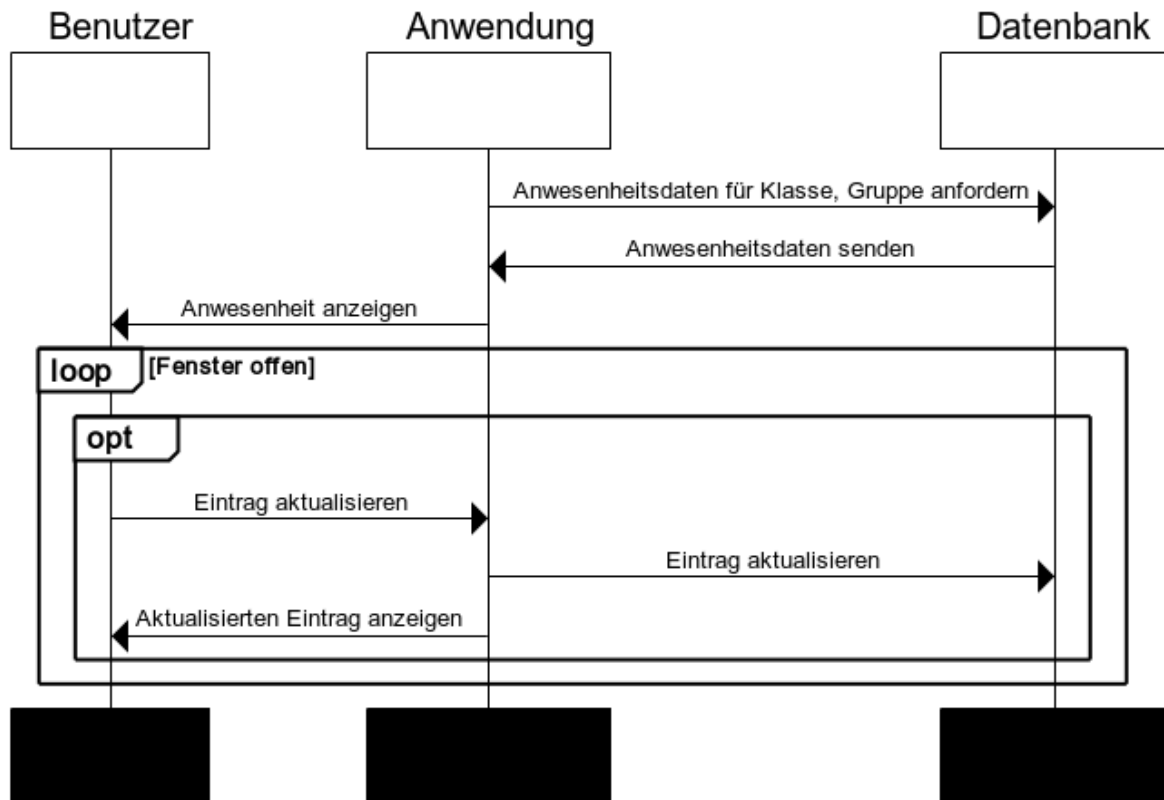
## Nachricht senden



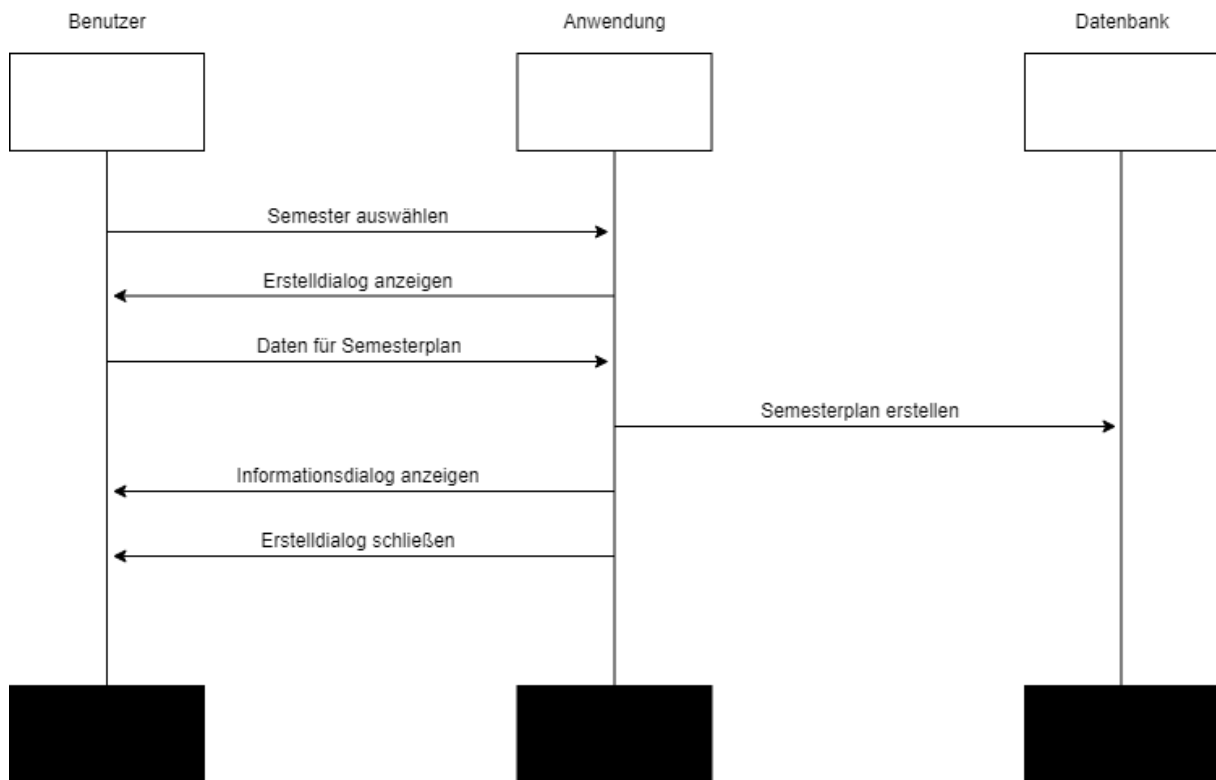
## Nachricht empfangen



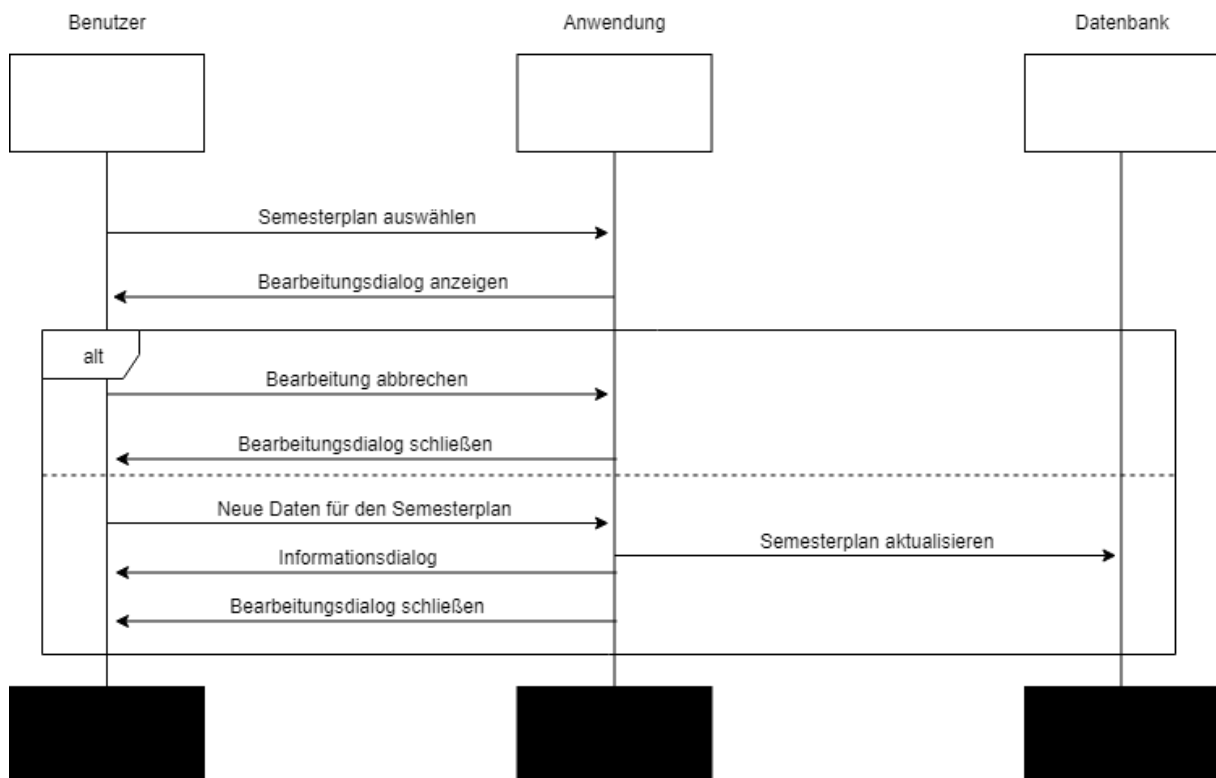
## Tardy



## Semesterplan erstellen

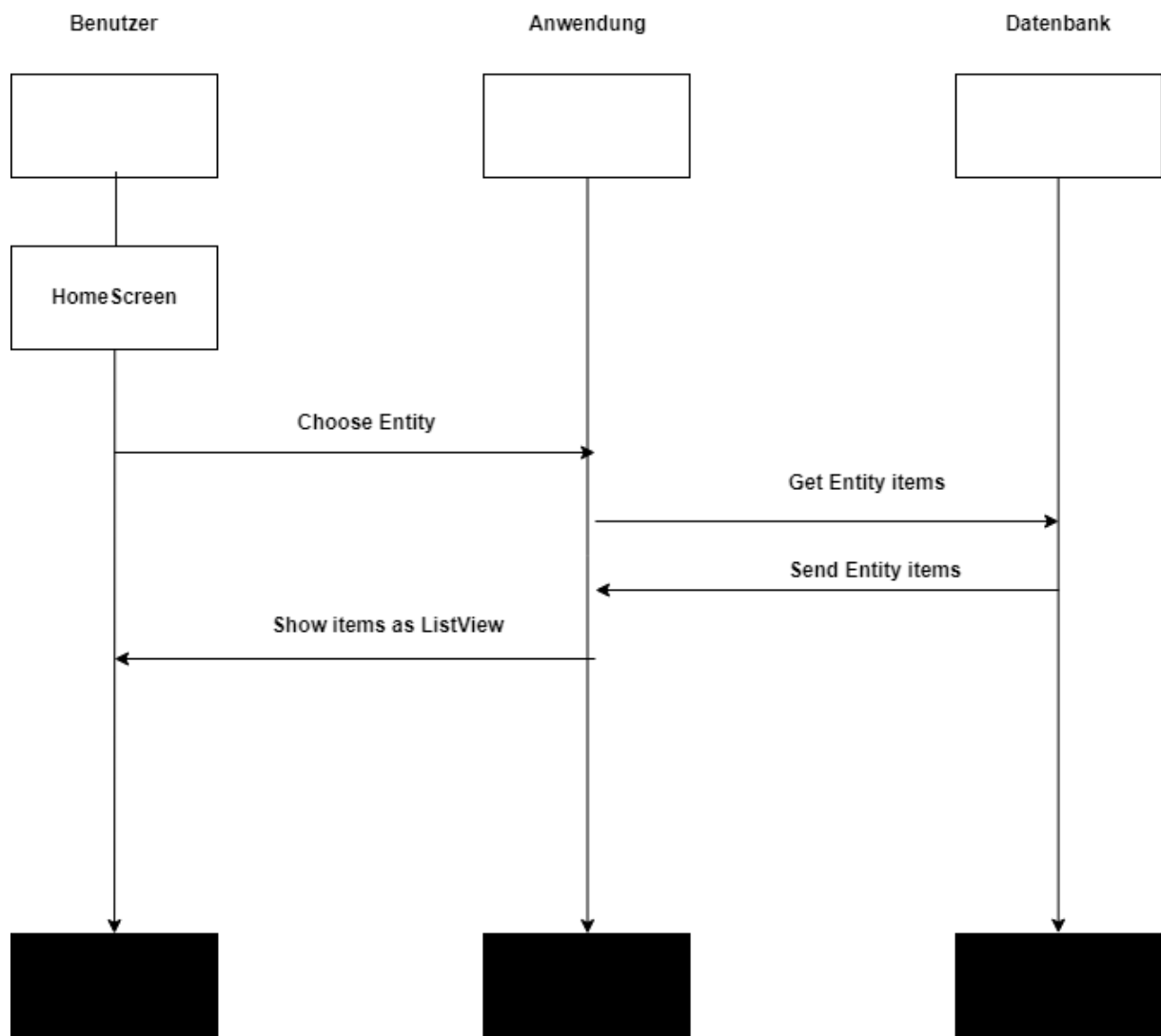


## Semesterplan editieren

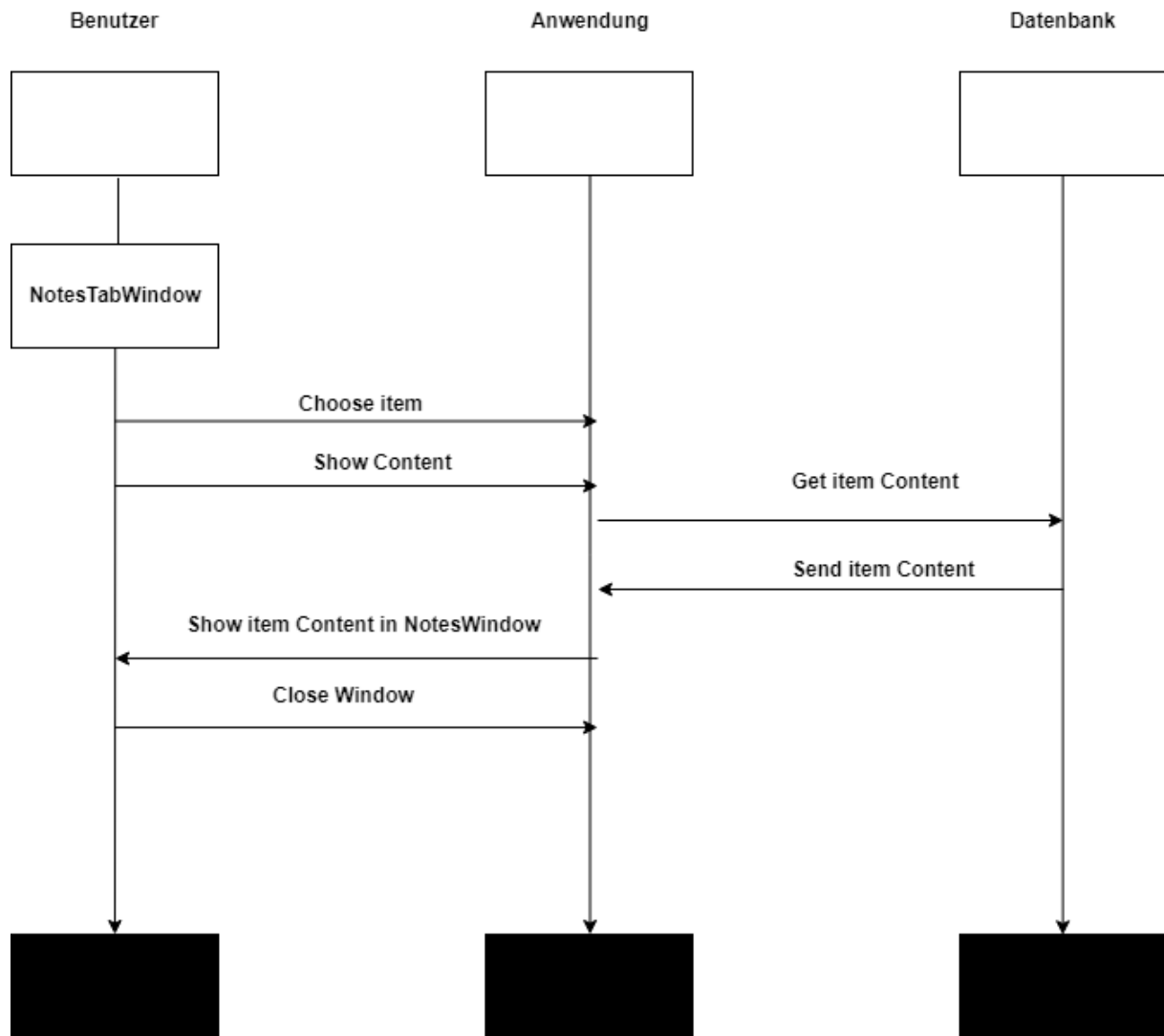




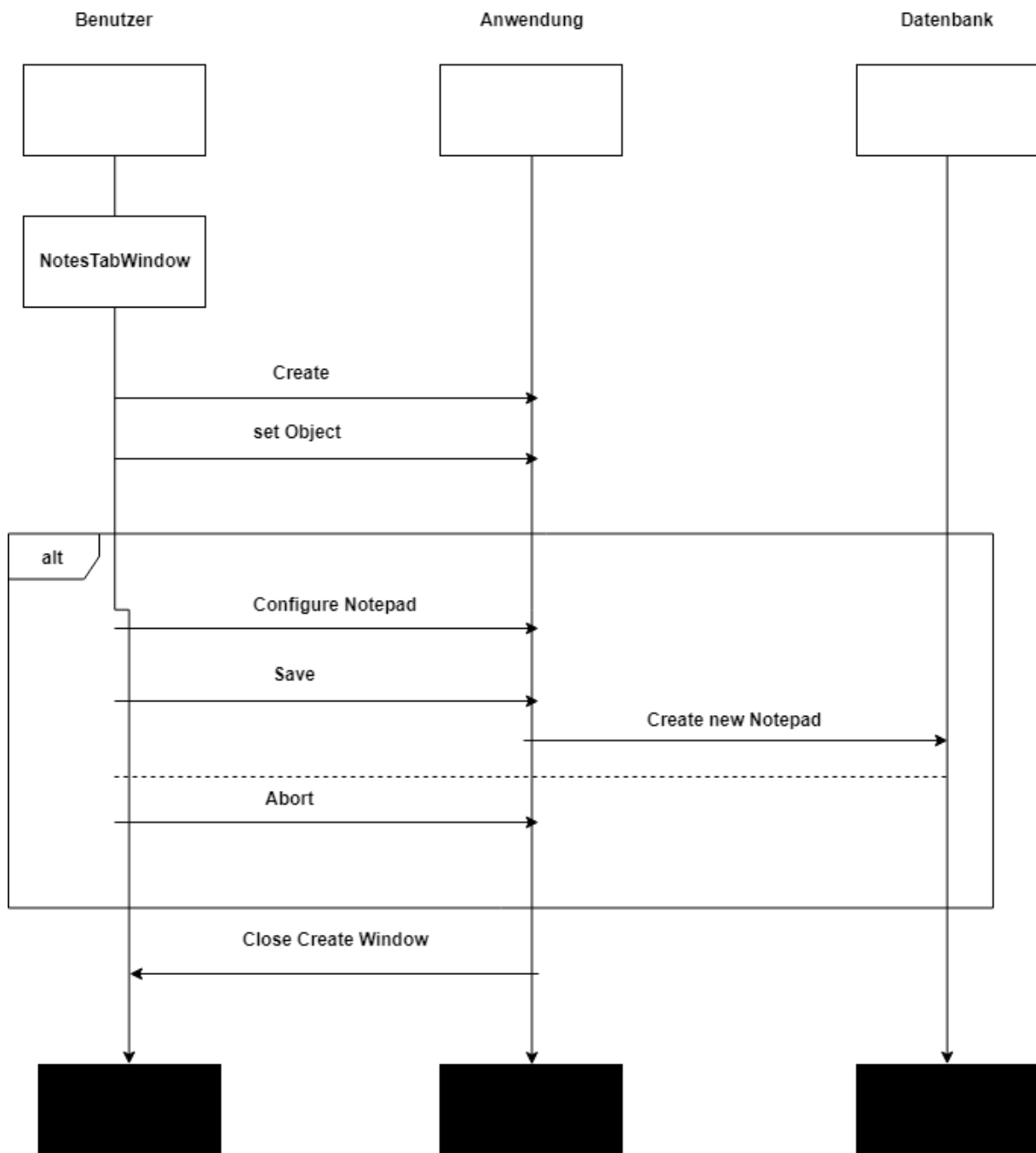
## Notes Tab



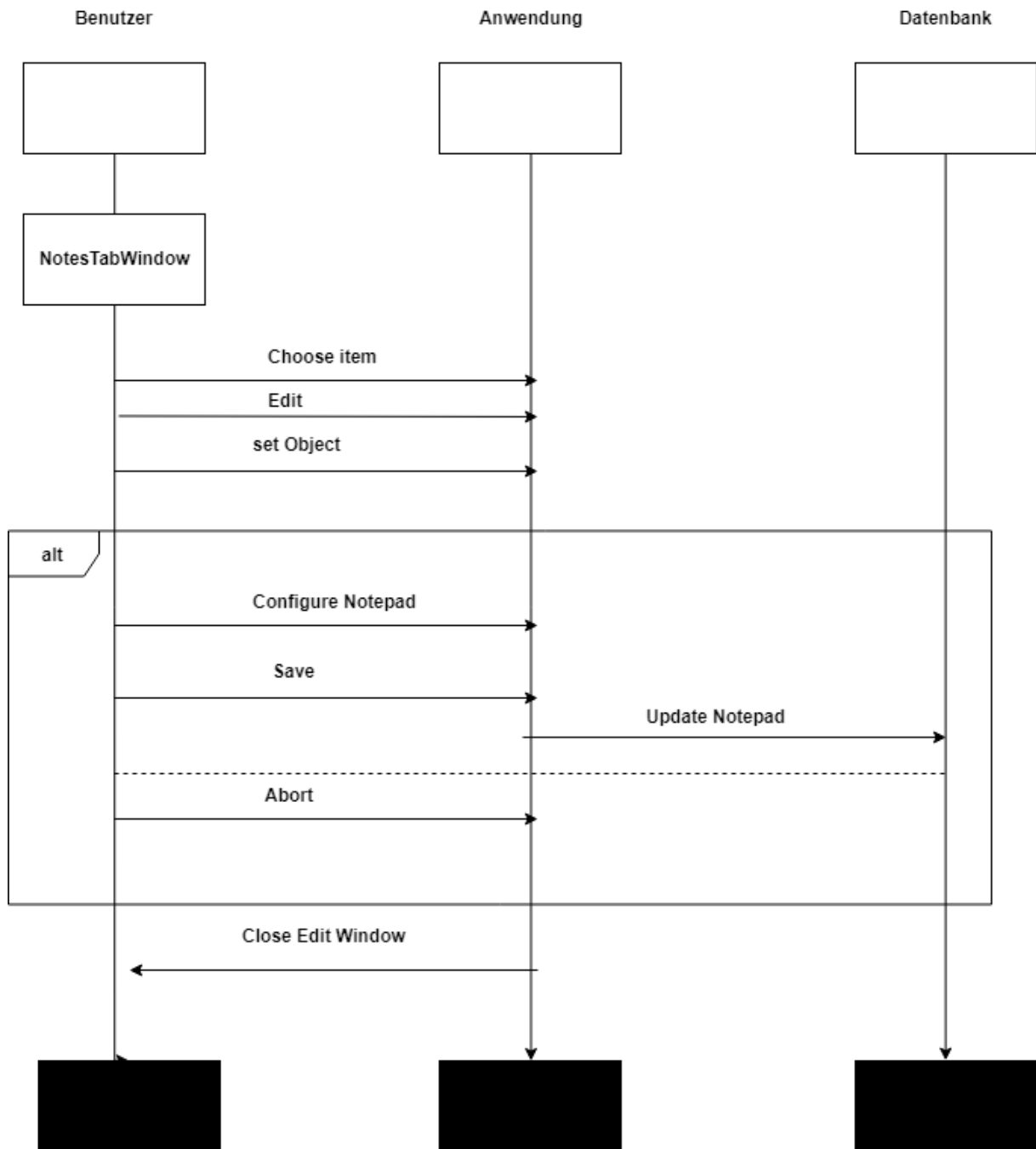
## Show Note



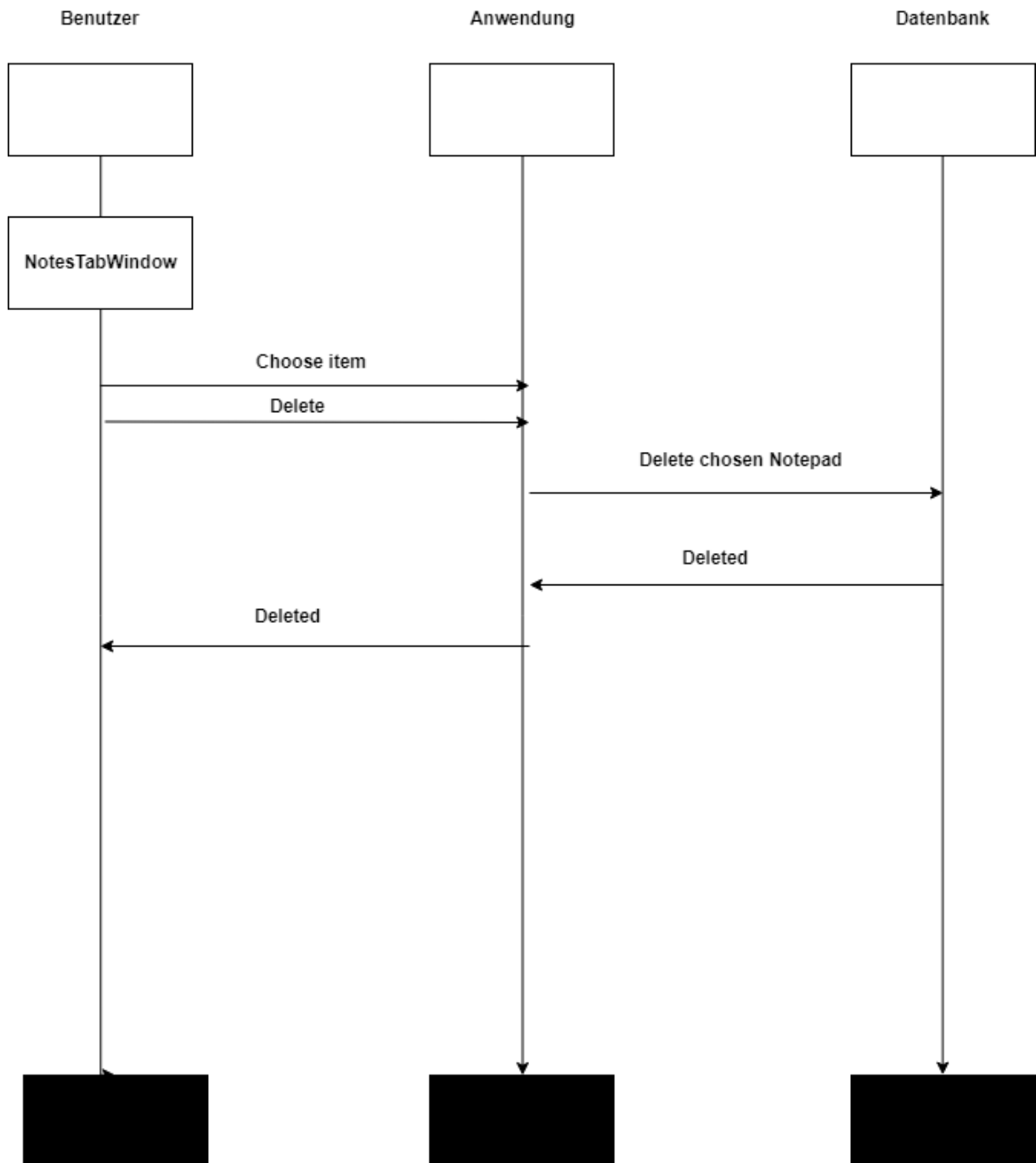
## Create Note



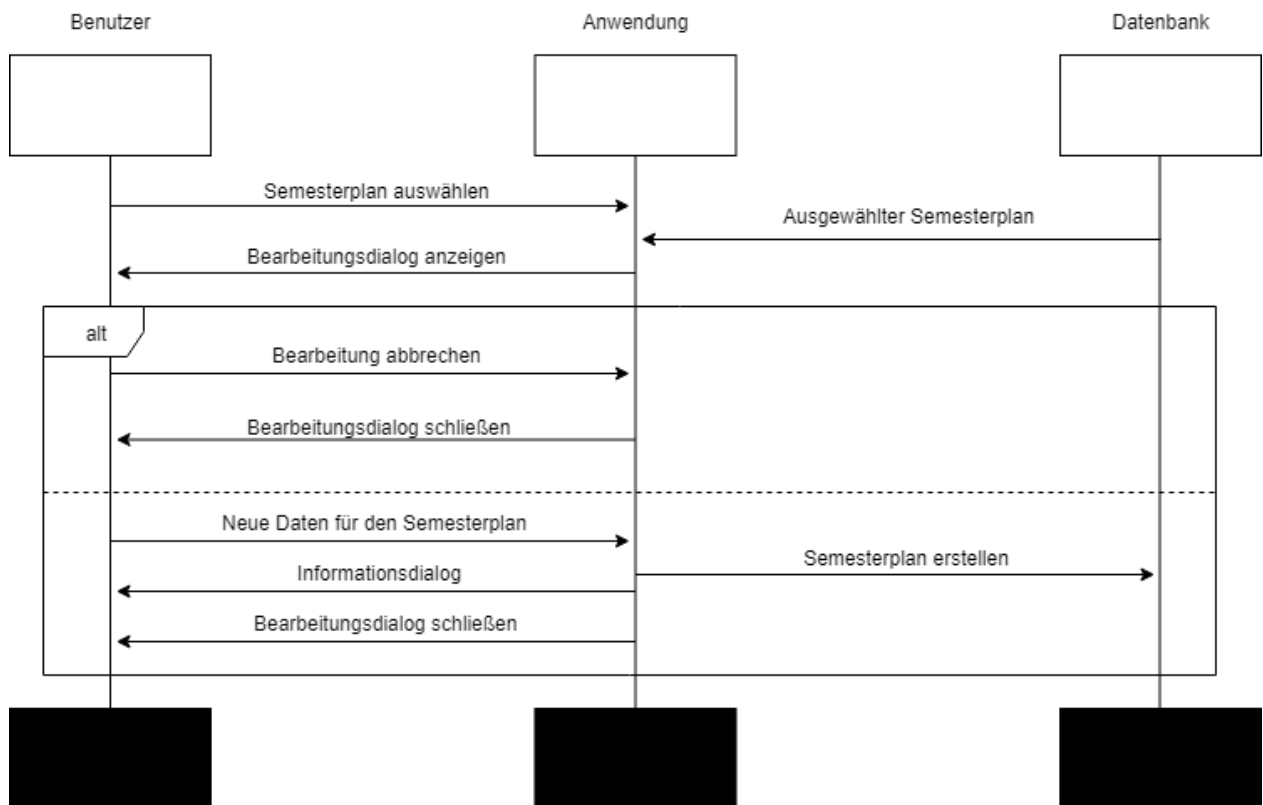
## Edit Note



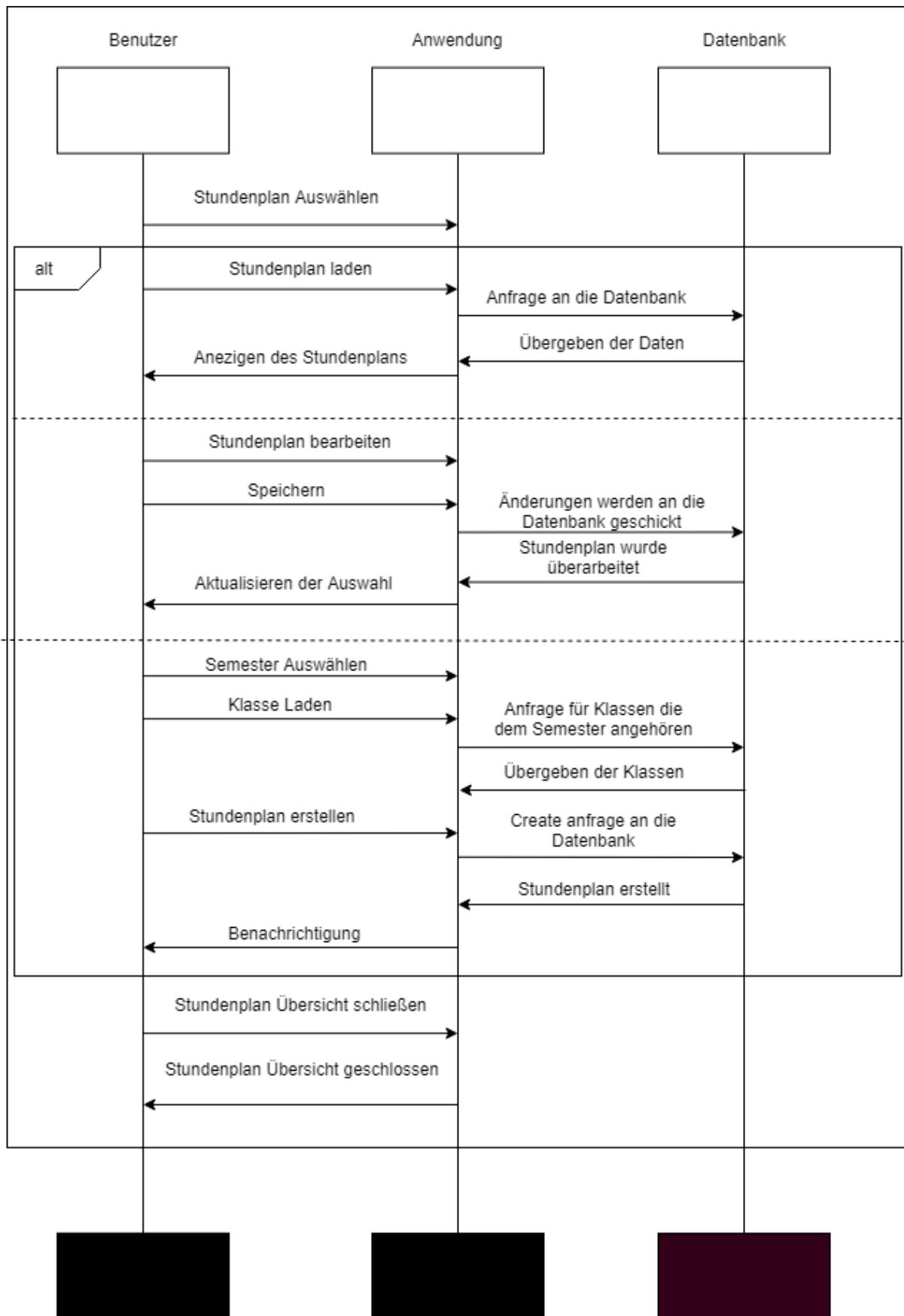
## Delete Note



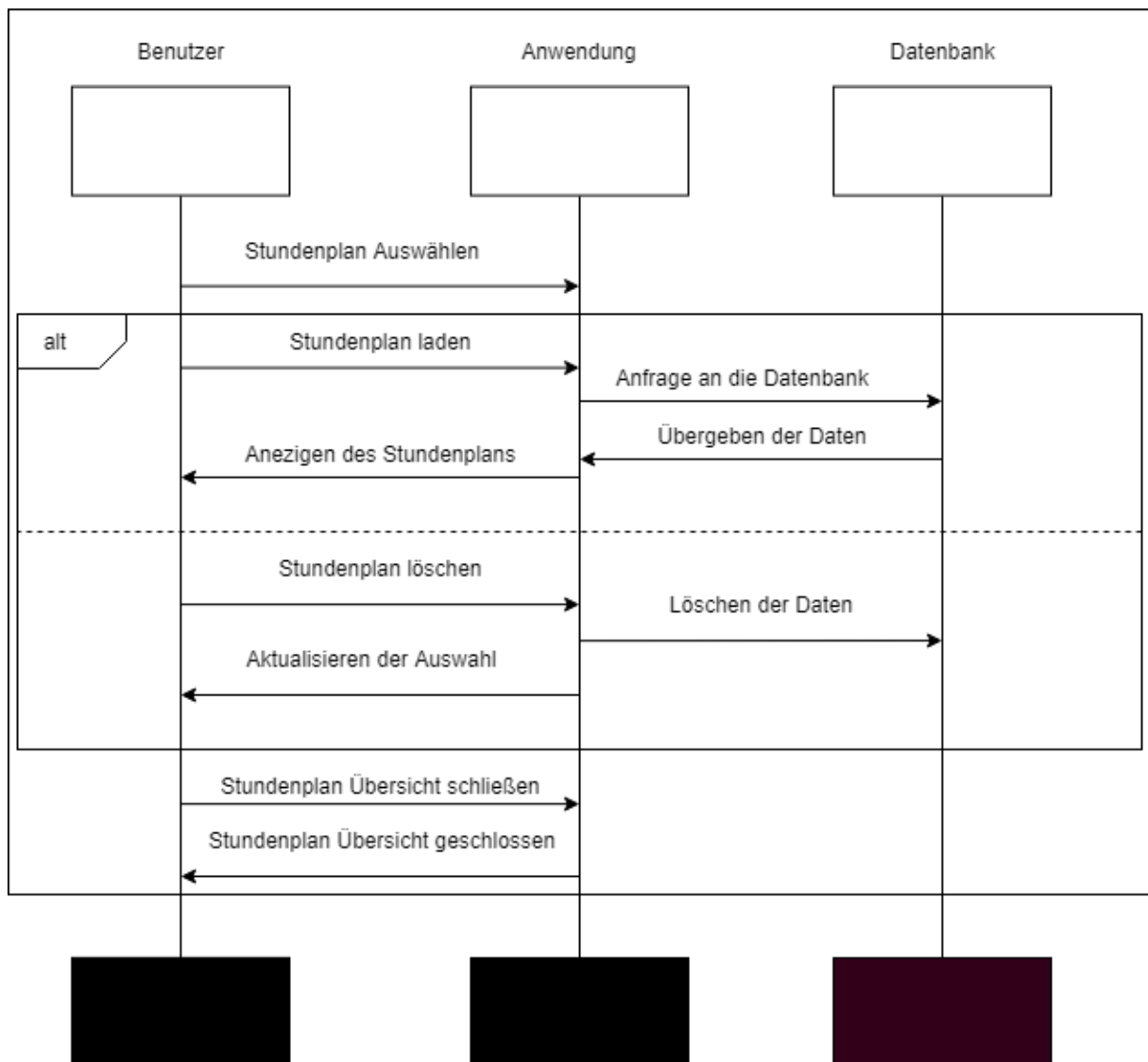
## Edit Termtable



## Create Timetable

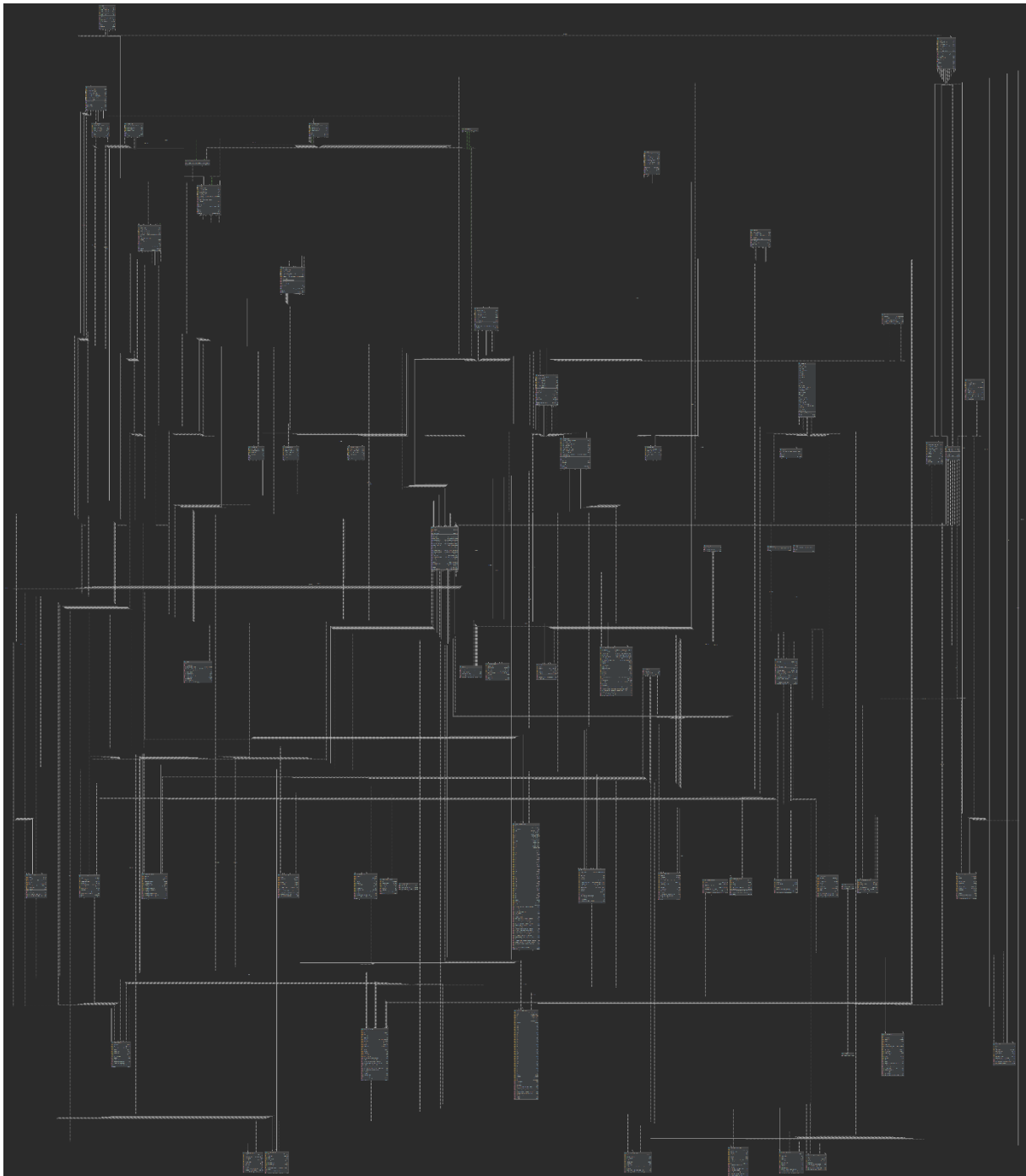


## Timetable View





## UML Klassendiagramm



## Funktionalitätsplanung

Id	Funktionalität	Geschätzter Aufwand	Verantwortlicher	Abhängige Funktionalitäten	Quellcode-referenz	Status
<b>1.</b>	<b>Dokumentation</b>					
1.1	Erstellen der User Stories	3h	Andreas, Johannes, Tobias, Kaan	bMSCs, hMSC		fertig
1.2	Erstellen des hMSCs	2h	Andreas	bMSCs		fertig
1.3	Erstellen der bMSCs	3h	Johannes, Merlin, Tobias, Kaan	Klassendiagramm		fertig
1.4	Erstellen des Klassendiagramms	2h	Andreas, Johannes, Merlin, Tobias			fertig
1.5	Funktionalitätsplanung	1h	Andreas, Tobias			fertig
1.6	Erstellen der Präsentation	2h	Johannes			fertig
<b>2.</b>	<b>Datenbank</b>					
2.1	Notifications	3h	Merlin	Chat, Home	DBNotificationListener	fertig
<b>3.</b>	<b>Controller</b>					
3.1	Registrierung	8h	Tobias	Login	Register-Controller	fertig
3.2	User Administration	6h	Tobias		UserAdministrationController	fertig
3.3	Erstellen des InviteCodes	5h	Tobias	Register, UserAdministration	InviteCodeController	fertig
3.4	Passwort ändern	6h	Tobias	User-Administration	ResetPasswordController	fertig
3.5	Anwesenheitsliste	6h	Merlin	Tardy-Controller		fertig
3.6	Wochenkalender	12h	Alisan		CreateTimetableController, TimetableWindowController	fertig
3.7	Semesterplan	12h	Dave		SemesterPlan-Controller	fertig
3.8	Chat	16h	Johannes		ChatTab-Controller, Chat-Window-Controller	fertig
3.9	Notizen	16h	Kaan		Create Notepad,	fertig

					Edit-Notepad-Controller, NotesTapController, Notes-Window-Controller	
<b>3.10</b>	Anpassen und einbetten in die HomeView	16h	Andreas		Home-View-Controller	fertig
<b>3.11</b>	Letzte Sitzung wiederherstellen	16h	Andreas		Home-View-Controller	fertig
<b>3.12</b>	Wochenkalender	6h	Merlin, Tobias		Week-Calendar-Controller	fertig
<b>3.13</b>	Notizen	6h	Merlin		NoteViewControllor	fertig
<b>4.</b>	<b>Models</b>					
4.1	Favoriten	2h	Merlin	HomeView	FavSemester, FavGroupage, FavGroup, FavStudent	fertig
4.2	LastItem, LastTab Zu User hinzu	1h	Merlin	HomeView	User	fertig
4.3	Kalender	1h	Merlin	Wochen- & Semesterkalender	Calendar, CalendarEntry CalendarEInfo	fertig
4.4	Notepad	1h	Merlin	Notizen	Notepad, -Bridge, -Entity	fertig
4.5	Register	1h	Merlin		InviteCode	fertig
4.6	Anwesenheit	1h	Merlin		Tardy	fertig
<b>5.</b>	<b>Test</b>					
5.1	Erstellung der Tests	5h	Merlin	Gesamtes Projekt	DBTest	fertig

## Unittests

<b>Id</b>	<b>Getestete Funktionalität</b>	<b>Quellcode Referenz</b>	<b>Status</b>
<b>1</b>	Objekt erstellen	DBTest	bestanden
<b>2</b>	Objekte in DB speichern	DBTest	bestanden
<b>3</b>	Objekte aus der DB abrufen	DBTest	bestanden

## Systemtests

<b>Datum</b>	06.12.2018	
<b>Tester</b>	Alexander Lohak	
<b>SW-Version</b>	V 1.7.0	
<b>Vorbedingung(en)</b>	Nutzer „besttutor“ ist am System mit Passwort „changme“ registriert	
<b>Schritt</b>	<b>Aktion (User)</b>	<b>Erwartete Reaktion (System)</b>
1	Der Benutzer gibt den Benutzernamen „besttutor“ auf der Tastatur ein.	Der Username lässt sich in das dafür vorgesehene Feld schreiben.
2	Der Benutzer gibt das Passwort „geheim“ auf der Tastatur ein.	Das Passwort lässt sich in das dafür vorgesehene Feld schreiben und die Anzeige besteht aus „*“.
3	Der Benutzer klickt auf „Anmelden“.	Das System loggt den User ein und die Nutzeroberfläche der Software wird angezeigt.
<b>Nachbedingung(en)</b>	Nutzer ist am System angemeldet.	
<b>Testurteil</b>	Test bestanden.	

<b>Datum</b>	06.12.2018	
<b>Tester</b>	Alexander Lohak	
<b>SW-Version</b>	V 1.7.0	
<b>Vorbedingung(en)</b>	Nutzer „besttutor“ ist im System angemeldet und „moretutor“ ist im System angemeldet.	
<b>Schritt</b>	<b>Aktion (User)</b>	<b>Erwartete Reaktion (System)</b>
1	Der Benutzer „moretutor“ startet den Chat durch Doppelklick auf den Usernamen „besttutor“.	Ein Chatfenster öffnet sich, die History wird geladen und angezeigt.
2	Der Benutzer „moretutor“ schreibt eine Nachricht an „besttutor“,	Die Nachricht wird in den Chatfenstern von beiden Benutzern angezeigt.
3	Der Benutzer „besttutor“ schreibt eine Nachricht an „moretutor“,	Die Nachricht wird in den Chatfenstern von beiden Benutzern angezeigt.
<b>Nachbedingung(en)</b>	Die gesendeten Nachrichten sind in der Datenbank gespeichert.	
<b>Testurteil</b>	Test bestanden.	

<b>Datum</b>	06.12.2018	
<b>Tester</b>	Alexander Lohak	
<b>SW-Version</b>	V 1.7.0	
<b>Vorbedingung(en)</b>	Benutzer „besttutor“ ist angemeldet.	
<b>Schritt</b>	<b>Aktion (User)</b>	<b>Erwartete Reaktion (System)</b>
1	Klick auf „Neues Semester“	Es öffnet sich ein neuer Tab, in welchem die Informationen eingetragen werden können.
2	Informationen werden eingetragen	Die Informationen lassen sich in die dafür vorgesehenen Textfelder eintragen.
3	Klick auf „Speichern“	Das neue Semester wird gespeichert, ist in der Nutzeroberfläche vorhanden und lässt sich auswählen.
<b>Nachbedingung(en)</b>	Das erstellte Semester ist in der Datenbank gespeichert.	
<b>Testurteil</b>	Test bestanden.	

# Iteration III

## User-Stories

<b>User Story-ID</b>	048
<b>User Story-Beschreibung</b>	Als Benutzer möchte ich Emails senden können, um einer Person, die keinen Zugriff auf das System hat, eine Nachricht zu senden.
<b>Geschätzter Realisierungsaufwand</b>	8h
<b>Priorität</b>	hoch
<b>Autor(en)</b>	Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	
<b>Zugehörige Szenarien</b>	bMSC_SendMail

<b>User Story-ID</b>	049
<b>User Story-Beschreibung</b>	Als Benutzer möchte ich verschiedene Kontakte auswählen können, um eine Email an mehrere Studierende senden zu können.
<b>Geschätzter Realisierungsaufwand</b>	6h
<b>Priorität</b>	hoch
<b>Autor(en)</b>	Johannes Morzeck
<b>Abhängigkeiten zu anderen User Stories</b>	048
<b>Zugehörige Szenarien</b>	bMSC_SelectContacts

<b>User Story-ID</b>	50
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich Anhänge an meine Emails heften, um Dateien per Mail verschicken zu können.
<b>Geschätzter Realisierungsaufwand</b>	8h
<b>Priorität</b>	hoch
<b>Autor(en)</b>	Johannes Morzeck
<b>Abhängigkeiten zu anderen User Stories</b>	48
<b>Zugehörige Szenarien</b>	bMSC_AddAttachment

<b>User Story-ID</b>	051
<b>User Story-Beschreibung</b>	Als Betreuer möchte Email empfangen können, um Email empfangen zu können.
<b>Geschätzter Realisierungsaufwand</b>	8h
<b>Priorität</b>	hoch
<b>Autor(en)</b>	Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	048
<b>Zugehörige Szenarien</b>	bMSC_ReceiveMail

<b>User Story-ID</b>	052
<b>User Story-Beschreibung</b>	Als Betreuer möchte Email Template erstellen können, um vorgefertigte Nachrichten zu versenden.
<b>Geschätzter Realisierungsaufwand</b>	8h
<b>Priorität</b>	hoch
<b>Autor(en)</b>	Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	048
<b>Zugehörige Szenarien</b>	bMSC _CreateMailTemplate

<b>User Story-ID</b>	053
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich aus meinen erstellten Templates eins auswählen können, um dieses zu verwenden.
<b>Geschätzter Realisierungsaufwand</b>	8h
<b>Priorität</b>	hoch
<b>Autor(en)</b>	Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	052
<b>Zugehörige Szenarien</b>	bMSC _TemplateList

<b>User Story-ID</b>	054
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich bestehende Templates löschen könne, um redundante Daten zu entfernen.
<b>Geschätzter Realisierungsaufwand</b>	8h
<b>Priorität</b>	hoch
<b>Autor(en)</b>	Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	053
<b>Zugehörige Szenarien</b>	bMSC _DeleteTemplate

<b>User Story-ID</b>	055
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich bestehende Templates bearbeiten können, um bestehende Templates an meine Bedürfnisse anzupassen.
<b>Geschätzter Realisierungsaufwand</b>	8h
<b>Priorität</b>	hoch
<b>Autor(en)</b>	Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	053
<b>Zugehörige Szenarien</b>	bMSC _EditTemplate



<b>User Story-ID</b>	056
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich mein Passwort zurücksetzen können, falls ich meins vergessen habe, um den Zugang zu gewährleisten.
<b>Geschätzter Realisierungsaufwand</b>	8h
<b>Priorität</b>	hoch
<b>Autor(en)</b>	Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	001
<b>Zugehörige Szenarien</b>	bMSC _ResetPassword

<b>User Story-ID</b>	057
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich eine Email mit einem neuen Passwort vom System geschickt bekommen, wenn ich das Passwort zurücksetze, um das neue Passwort zu verwenden.
<b>Geschätzter Realisierungsaufwand</b>	8h
<b>Priorität</b>	hoch
<b>Autor(en)</b>	Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	056
<b>Zugehörige Szenarien</b>	bMSC _ResetPasswordMail

<b>User Story-ID</b>	058
<b>User Story-Beschreibung</b>	Als Betreuer möchte ich, dass das neue Passwort im System nicht als Klartext gespeichert wird, um damit das Passwort geschützt ist.
<b>Geschätzter Realisierungsaufwand</b>	8h
<b>Priorität</b>	hoch
<b>Autor(en)</b>	Tobias Sieber
<b>Abhängigkeiten zu anderen User Stories</b>	056
<b>Zugehörige Szenarien</b>	bMSC _ResetPasswordHashPasswort

<b>User Story-ID</b>	059
<b>User Story-Beschreibung</b>	Als Benutzer möchte ich die Commits von Gitlab auf Klassen- , Gruppen- und Studentenebene einsehen können, um den Projektverlauf beurteilen zu können.
<b>Geschätzter Realisierungsaufwand</b>	8h
<b>Priorität</b>	mittel
<b>Autor(en)</b>	Andreas Häusler
<b>Abhängigkeiten zu anderen User Stories</b>	-
<b>Zugehörige Szenarien</b>	-

<b>User Story-ID</b>	060
<b>User Story-Beschreibung</b>	Als Benutzer möchte ich die Commits von Gitlab als Tortendiagramm erhalten, um den Anteil der entsprechenden Entitäten in Relation zu einander setzen zu können.
<b>Geschätzter Realisierungsaufwand</b>	8h
<b>Priorität</b>	mittel
<b>Autor(en)</b>	Andreas Häusler
<b>Abhängigkeiten zu anderen User Stories</b>	060
<b>Zugehörige Szenarien</b>	-

<b>User Story-ID</b>	061
<b>User Story-Beschreibung</b>	Als Benutzer möchte ich die Commits von Gitlab als Graphen einsehen können, um das Projekt im zeitlichen Verlauf nachvollziehen zu können.
<b>Geschätzter Realisierungsaufwand</b>	8h
<b>Priorität</b>	mittel
<b>Autor(en)</b>	Andreas Häusler
<b>Abhängigkeiten zu anderen User Stories</b>	060
<b>Zugehörige Szenarien</b>	-

<b>User Story-ID</b>	062
<b>User Story-Beschreibung</b>	Als Benutzer möchte ich für jede Gruppe ein Bewertungsformular erstellen können, um gestellte Fragen und zugehörige Antworten speichern und kommentieren zu können.
<b>Geschätzter Realisierungsaufwand</b>	4h
<b>Priorität</b>	mittel
<b>Autor(en)</b>	Dave Makila
<b>Abhängigkeiten zu anderen User Stories</b>	-
<b>Zugehörige Szenarien</b>	bMSC_CreateExamGroup

<b>User Story-ID</b>	063
<b>User Story-Beschreibung</b>	Als Benutzer möchte ich vorhandene Bewertungsformulare für Gruppen bearbeiten können, um Informationen aktuell zu halten.
<b>Geschätzter Realisierungsaufwand</b>	4h
<b>Priorität</b>	mittel
<b>Autor(en)</b>	Dave Makila
<b>Abhängigkeiten zu anderen User Stories</b>	062
<b>Zugehörige Szenarien</b>	bMSC_EditExamStudent

<b>User Story-ID</b>	064
<b>User Story-Beschreibung</b>	Als Benutzer möchte ich für jeden Studenten ein Bewertungsformular erstellen können, um gestellte Fragen zu speichern und diese bewerten und kommentieren zu können.
<b>Geschätzter Realisierungsaufwand</b>	4h
<b>Priorität</b>	mittel
<b>Autor(en)</b>	Dave Makila
<b>Abhängigkeiten zu anderen User Stories</b>	-062
<b>Zugehörige Szenarien</b>	bMSC_CreateExamStudent

<b>User Story-ID</b>	065
<b>User Story-Beschreibung</b>	Als Benutzer möchte ich vorhandene Bewertungsformulare für Studenten bearbeiten können, um Informationen aktuell zu halten.
<b>Geschätzter Realisierungsaufwand</b>	4h
<b>Priorität</b>	mittel
<b>Autor(en)</b>	Dave Makila
<b>Abhängigkeiten zu anderen User Stories</b>	062, 064
<b>Zugehörige Szenarien</b>	bMSC_EditExamStudent

<b>User Story-ID</b>	066
<b>User Story-Beschreibung</b>	Als Benutzer möchte ich alle existierenden Notizen in einer Liste sehen können, um einen besseren Überblick zu haben.
<b>Geschätzter Realisierungsaufwand</b>	3h
<b>Priorität</b>	mittel
<b>Autor(en)</b>	Kaan Erdogan
<b>Abhängigkeiten zu anderen User Stories</b>	039
<b>Zugehörige Szenarien</b>	bMSC_AllNotes

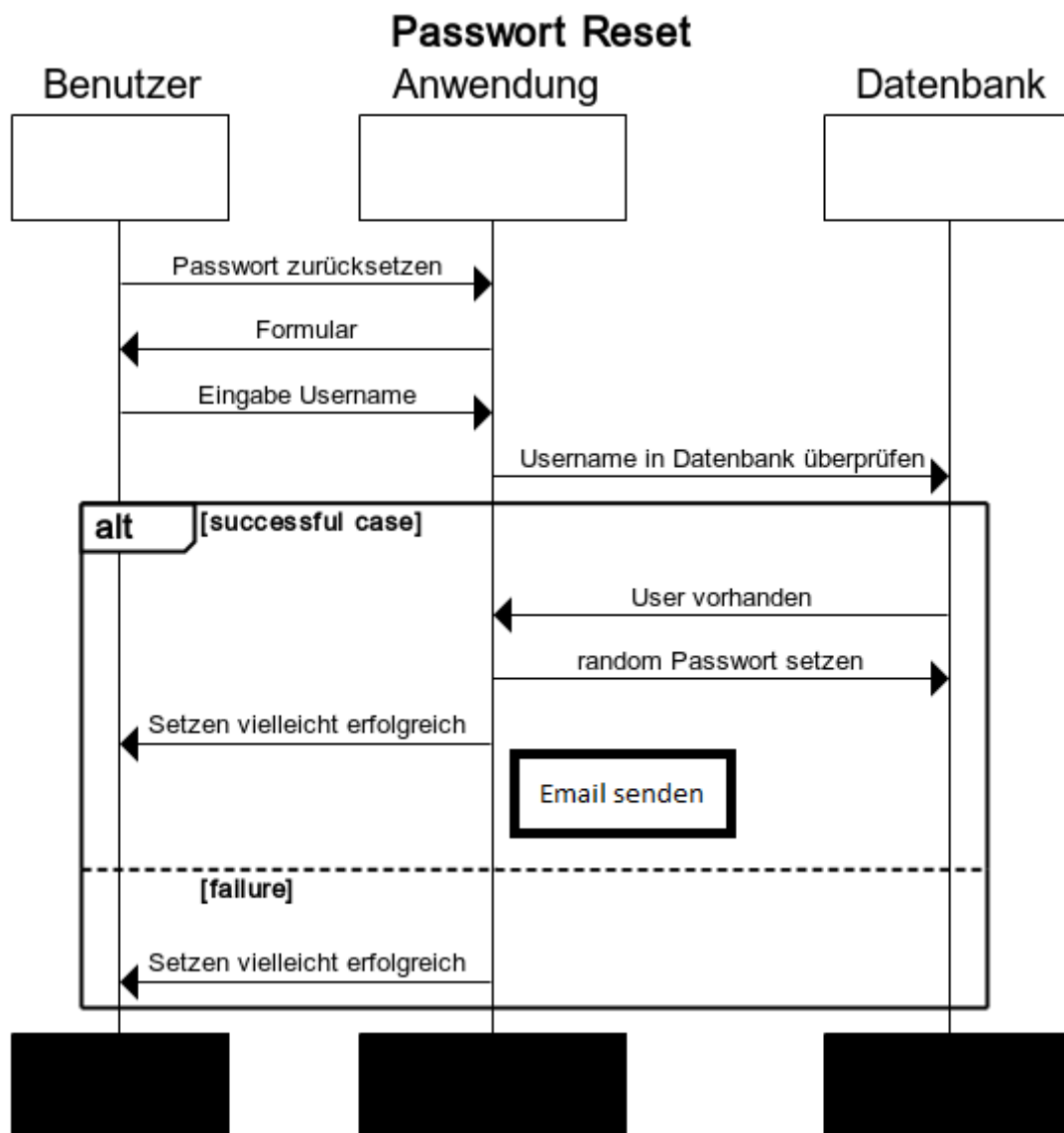
<b>User Story-ID</b>	067
<b>User Story-Beschreibung</b>	Als Benutzer möchte ich in der kompletten Notizliste nach Inhalt und Bezeichnung suchen können, um Notizen schnell finden zu können.
<b>Geschätzter Realisierungsaufwand</b>	3h
<b>Priorität</b>	mittel
<b>Autor(en)</b>	Kaan Erdogan
<b>Abhängigkeiten zu anderen User Stories</b>	064
<b>Zugehörige Szenarien</b>	bMSC_AllNotes

<b>User Story-ID</b>	068
<b>User Story-Beschreibung</b>	Als Benutzer möchte ich eine Liste zur Änderungshistorie aufrufen können, um Änderungen in den Notizen dokumentiert zu haben.
<b>Geschätzter Realisierungsaufwand</b>	6h
<b>Priorität</b>	mittel
<b>Autor(en)</b>	Kaan Erdogan
<b>Abhängigkeiten zu anderen User Stories</b>	064
<b>Zugehörige Szenarien</b>	bMSC_NotesHistory

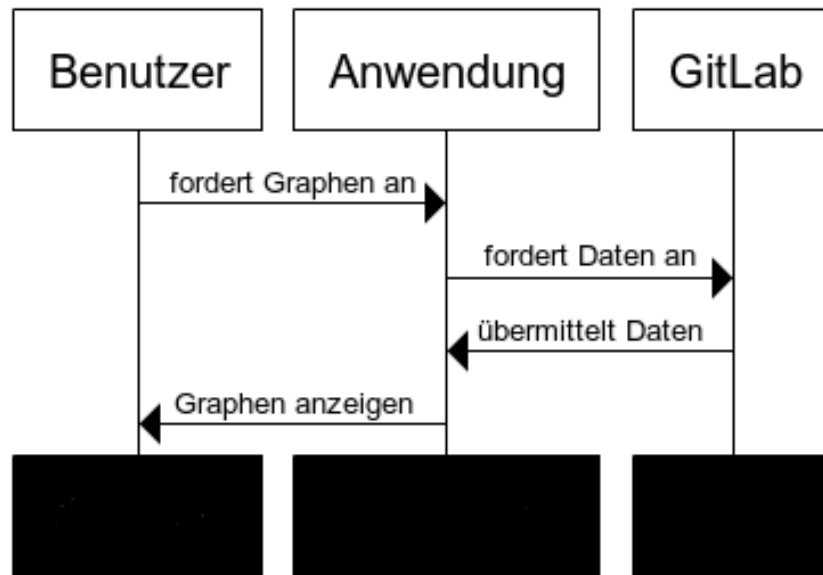
<b>User Story-ID</b>	069
<b>User Story-Beschreibung</b>	Als Benutzer möchte ich, dass die Priorisierungen in der Änderungshistorie als auch in der Notizliste farbig erkennbar sind, um auch dort die Priorisierung erkennen zu können.
<b>Geschätzter Realisierungsaufwand</b>	2h
<b>Priorität</b>	mittel
<b>Autor(en)</b>	Kaan Erdogan
<b>Abhängigkeiten zu anderen User Stories</b>	064, 066
<b>Zugehörige Szenarien</b>	bMSC_AllNotes, bMSC_NotesHistory

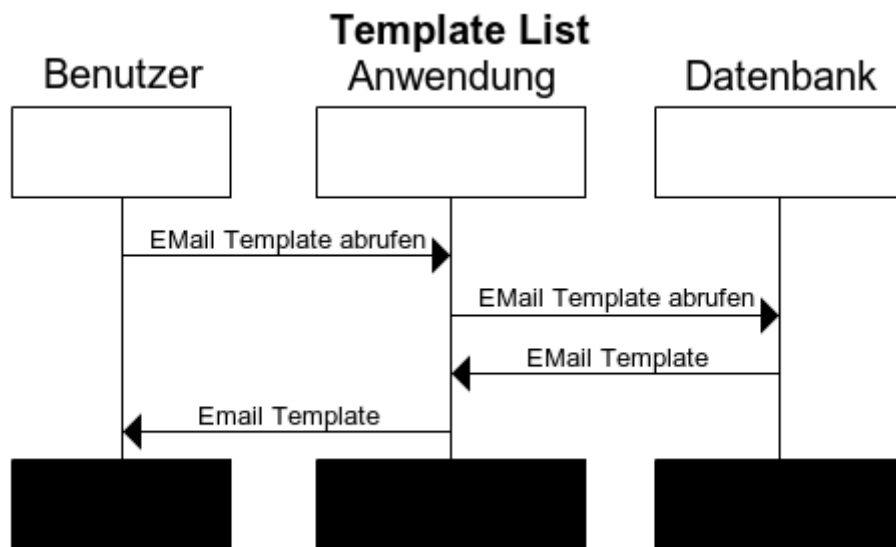
<b>User Story-ID</b>	070
<b>User Story-Beschreibung</b>	Als Benutzer möchte ich die Datenbank zurücksetzen, um einen sauberen Zustand herstellen zu können.
<b>Geschätzter Realisierungsaufwand</b>	2h
<b>Priorität</b>	mittel
<b>Autor(en)</b>	Merlin von Rössing
<b>Abhängigkeiten zu anderen User Stories</b>	N/A
<b>Zugehörige Szenarien</b>	bMSC_ResetDatabase

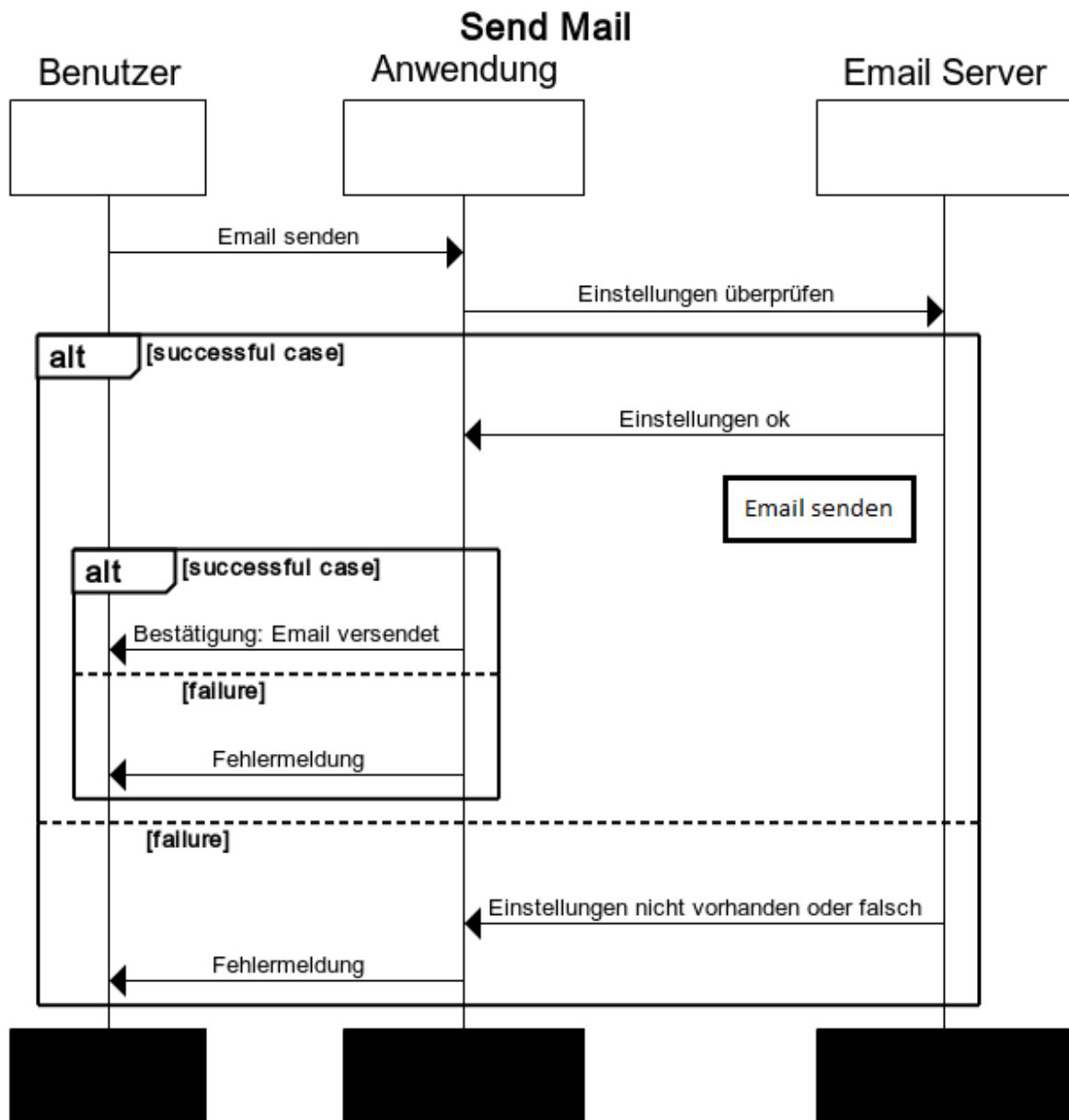
## MSCs



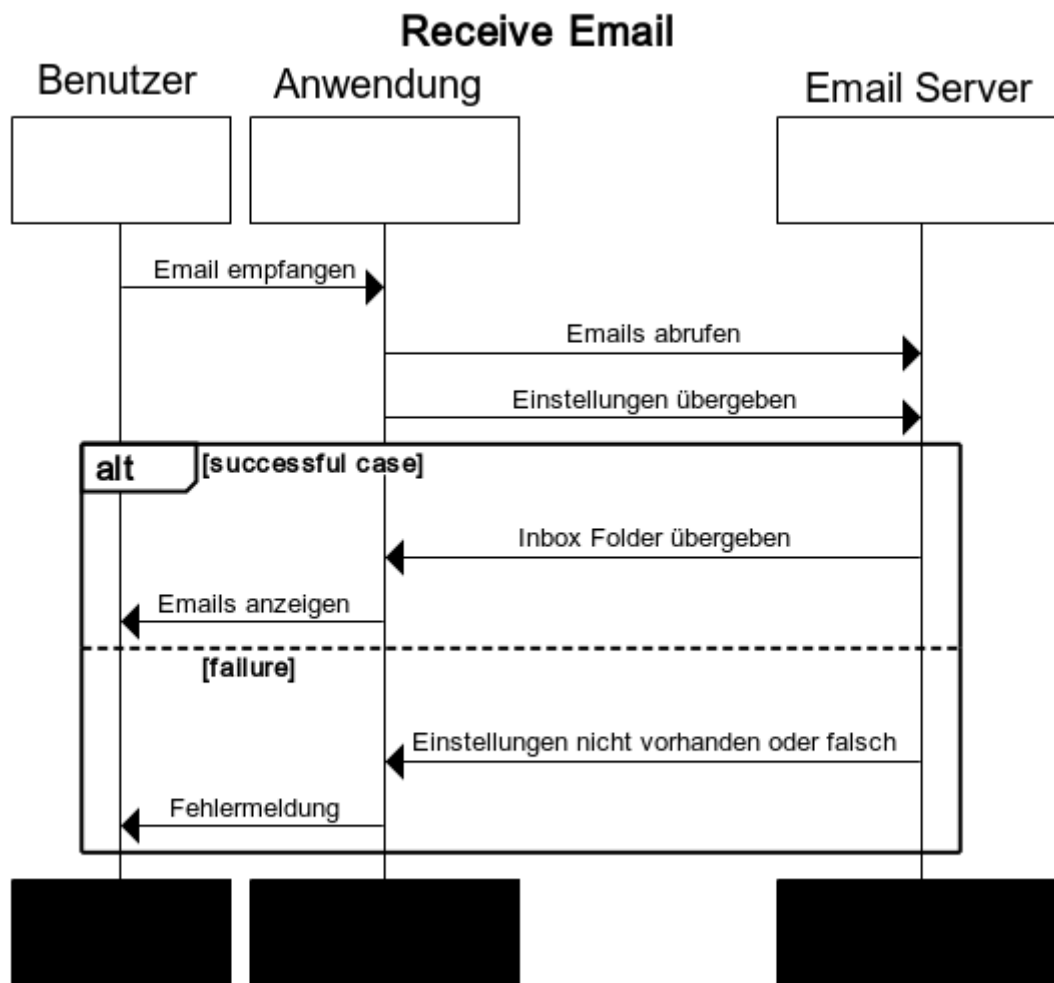
## GitLab Draw Chart



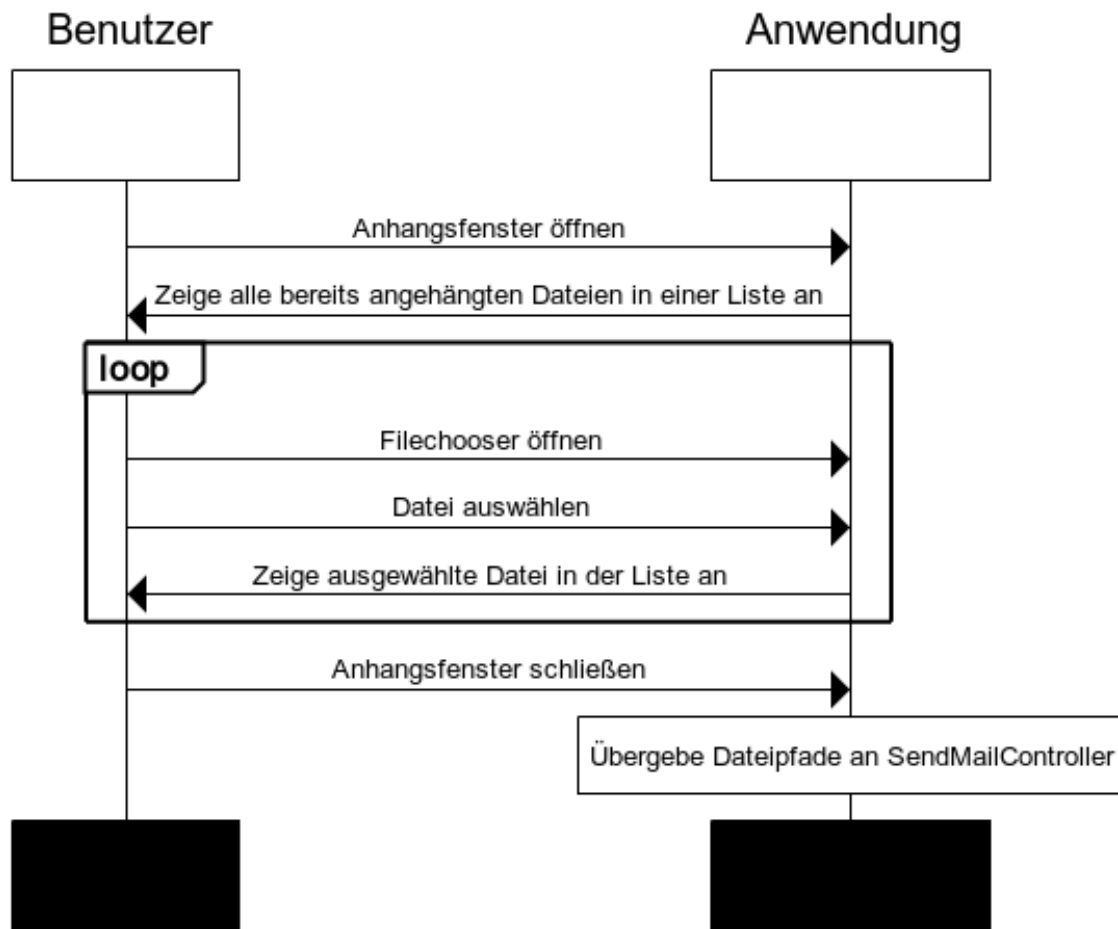




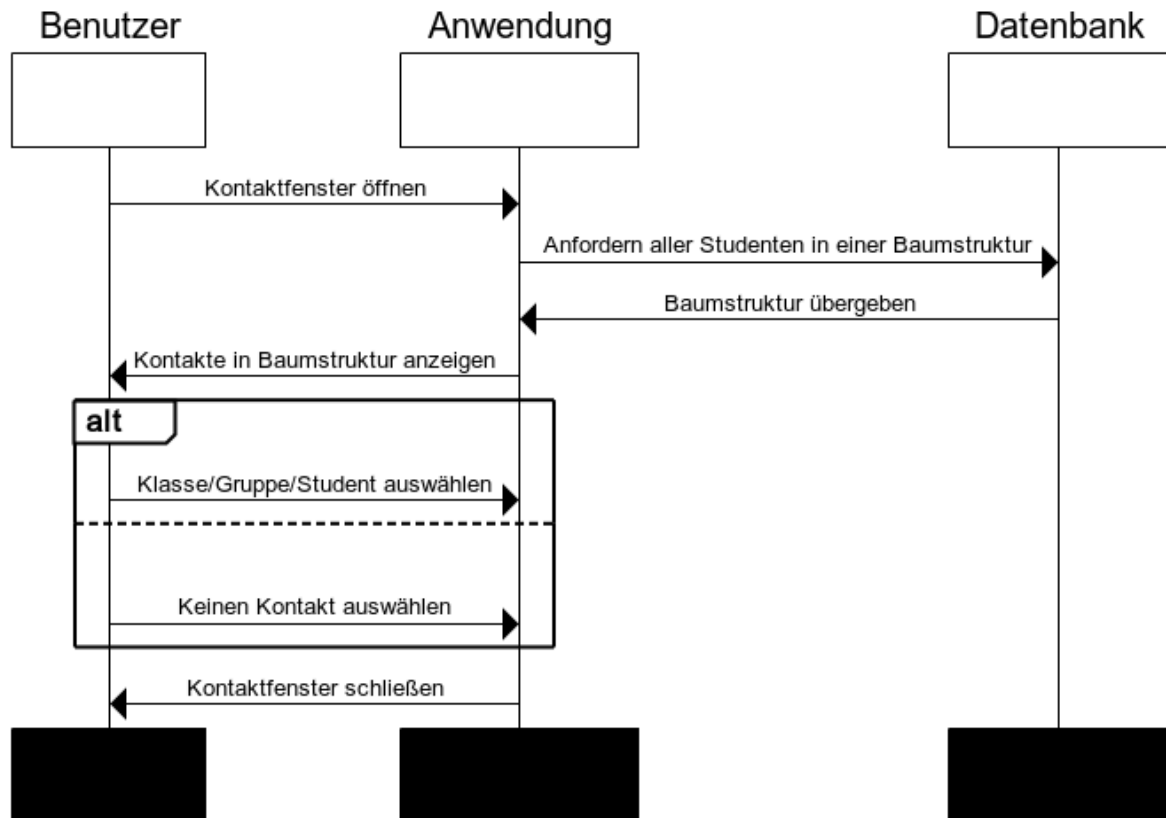




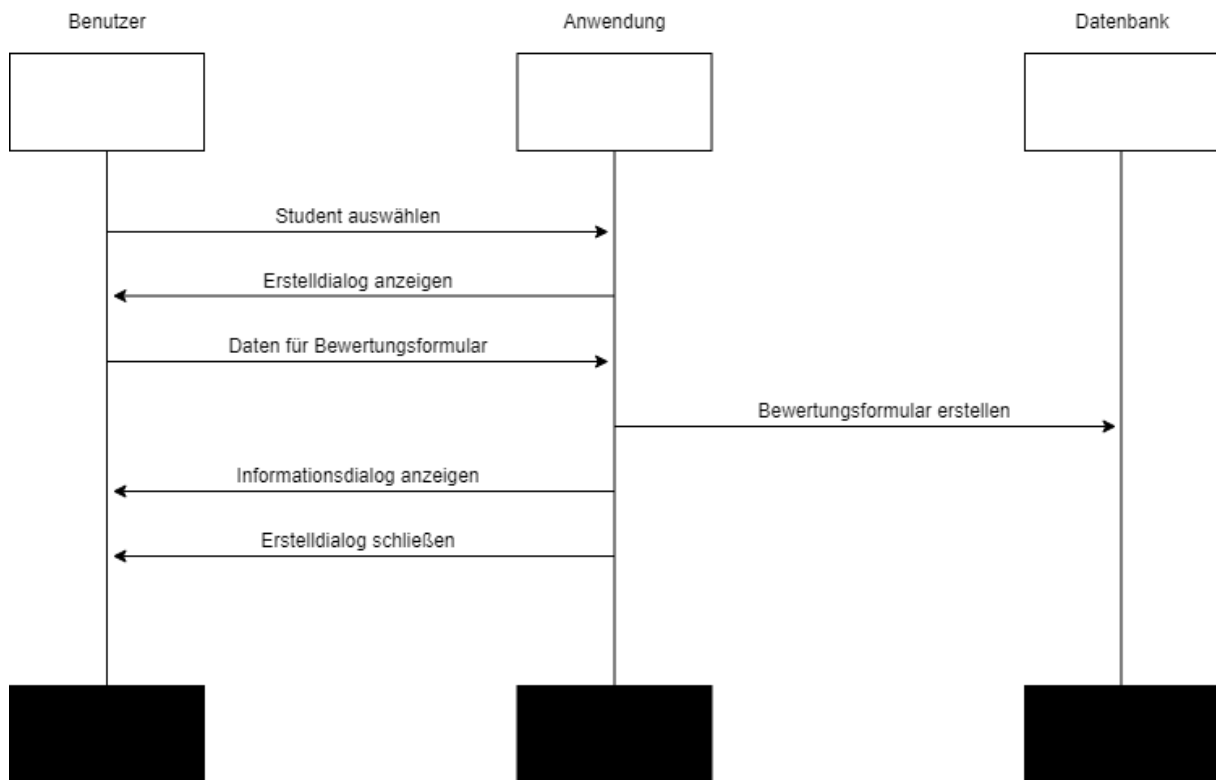
## Anhänge hinzufügen



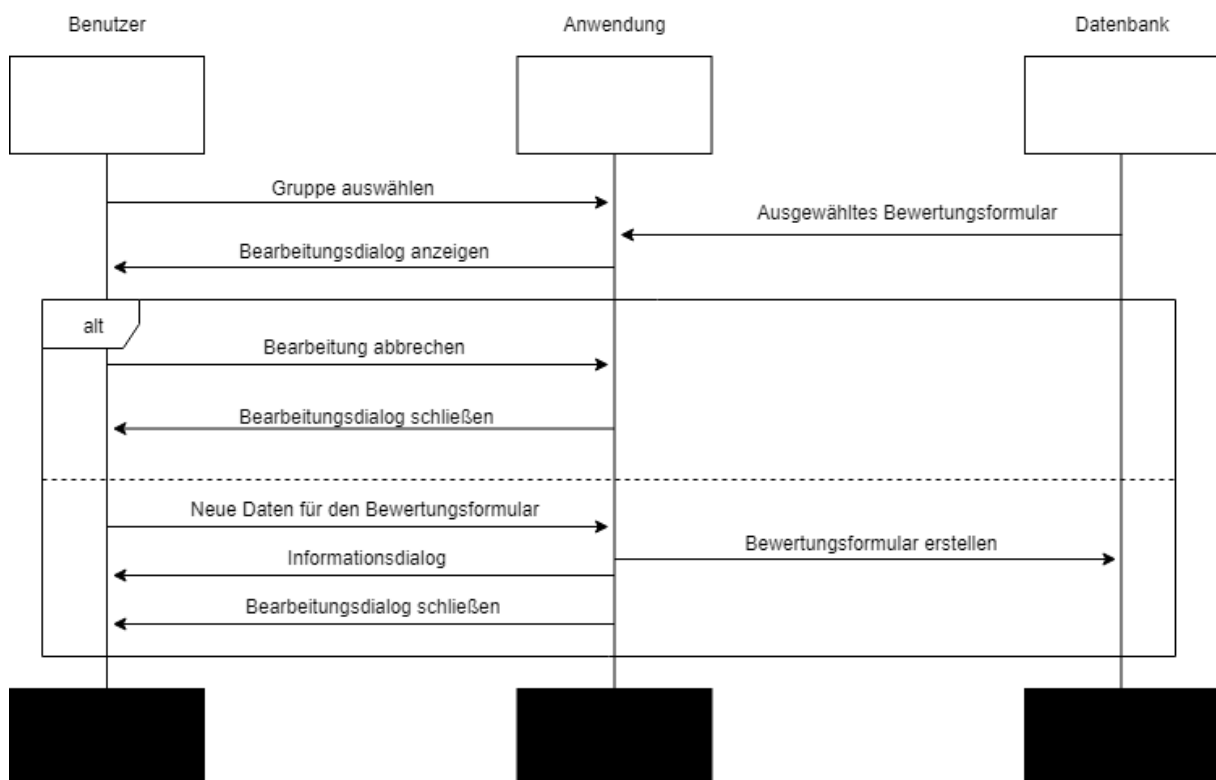
## Kontakte auswählen



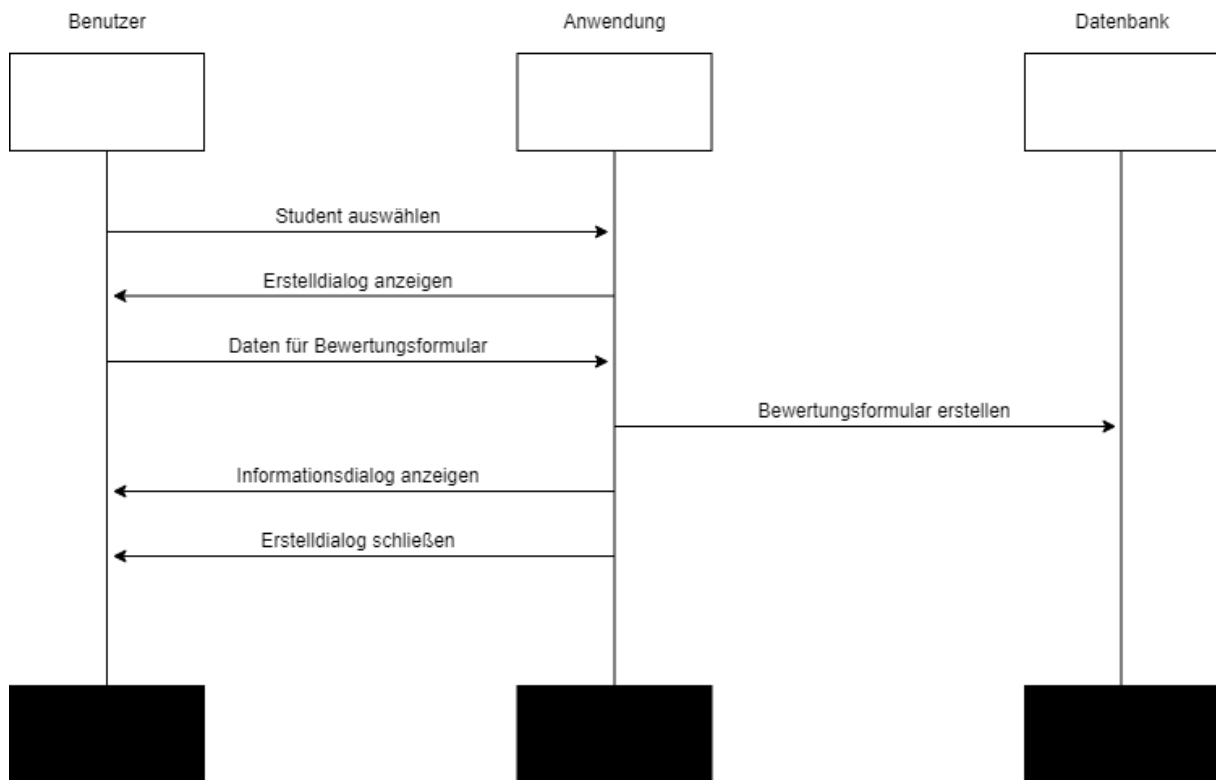
## Gruppenbewertungsformular erstellen



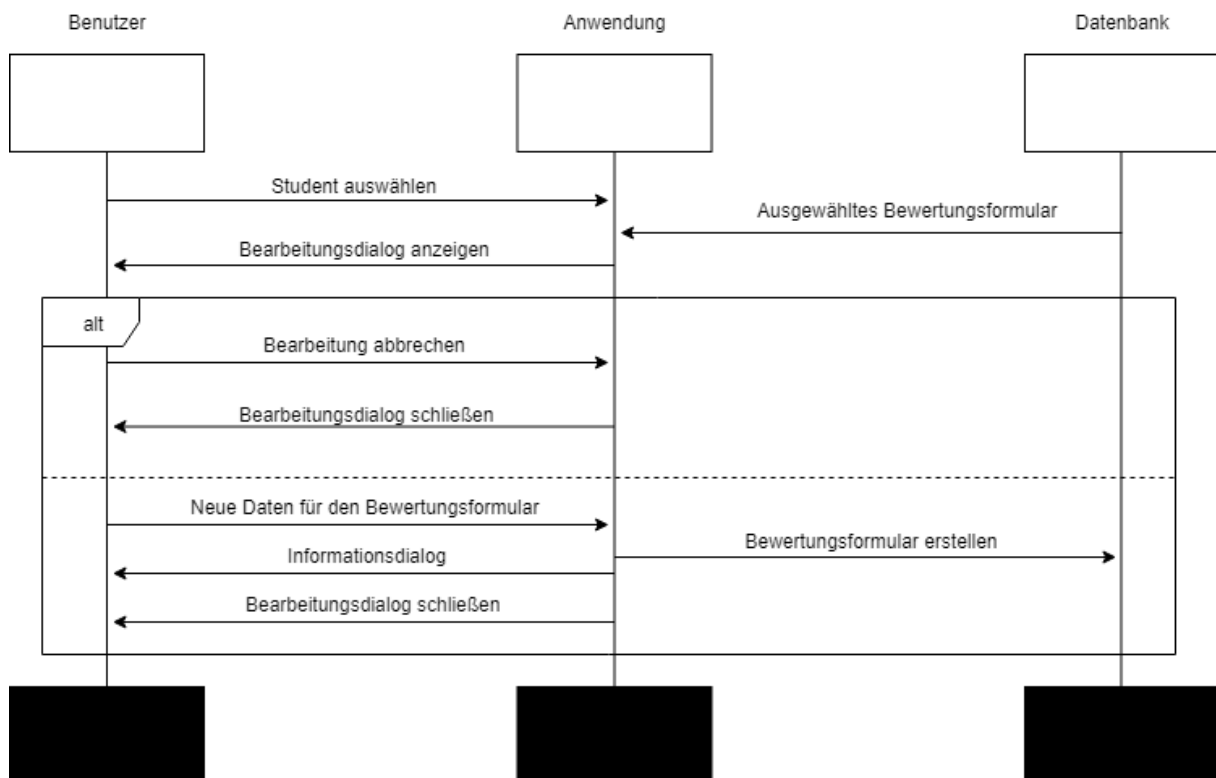
## Gruppenbewertungsformular bearbeiten



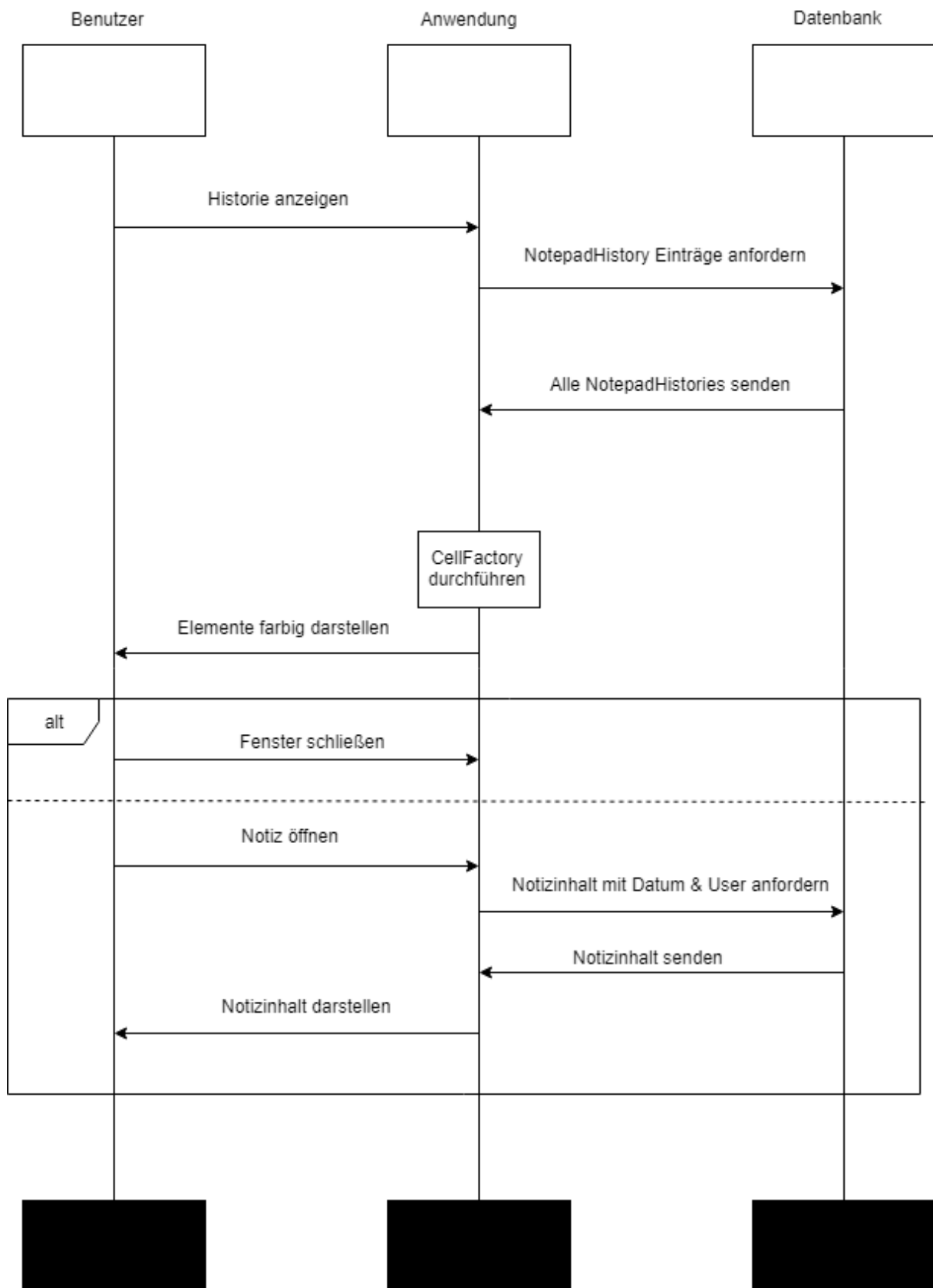
## Studentenbewertungsformular erstellen bewerten



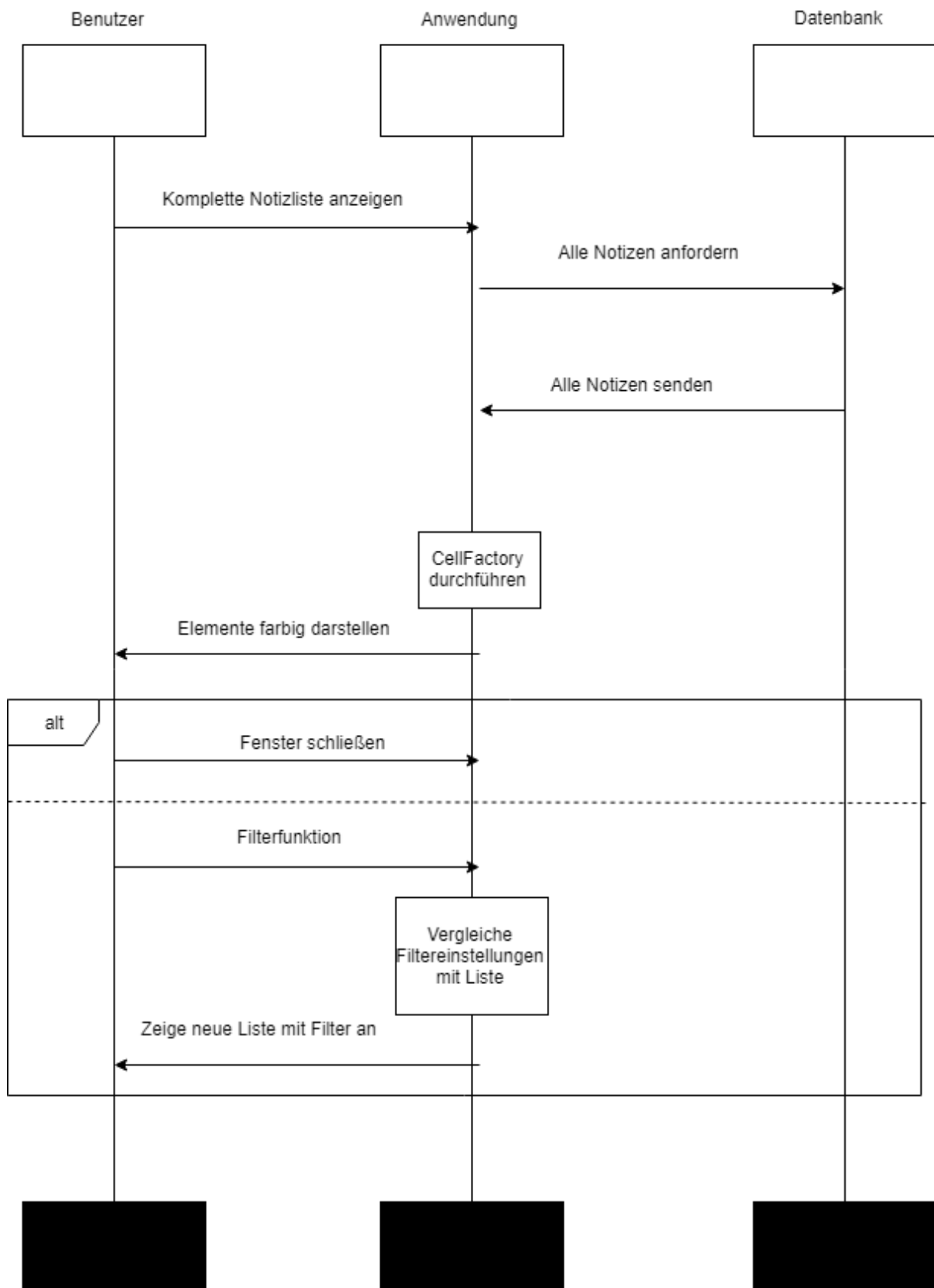
## Studentenbewertungsformular bearbeiten



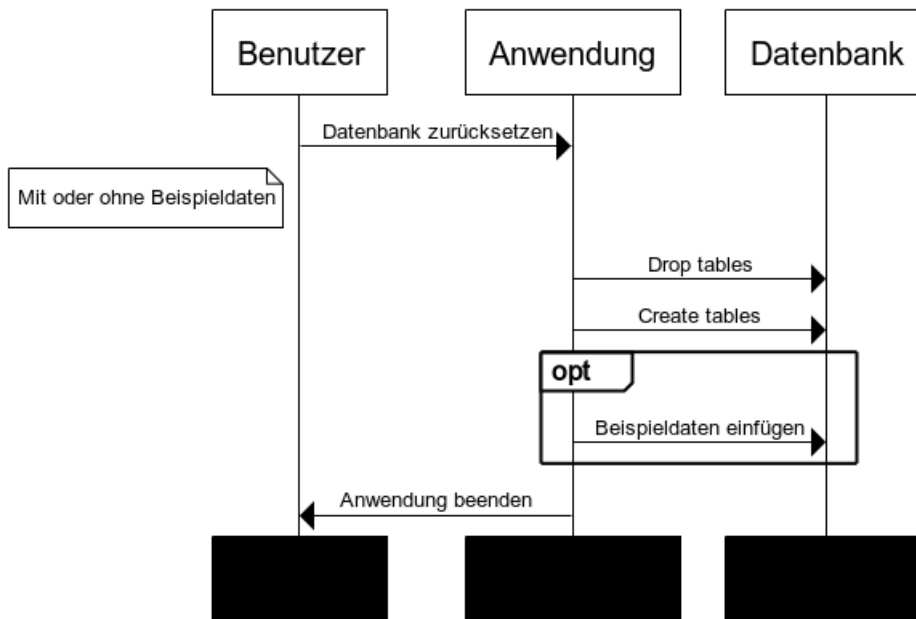
## Notizhistorie anzeigen



## Komplette Notizliste anzeigen

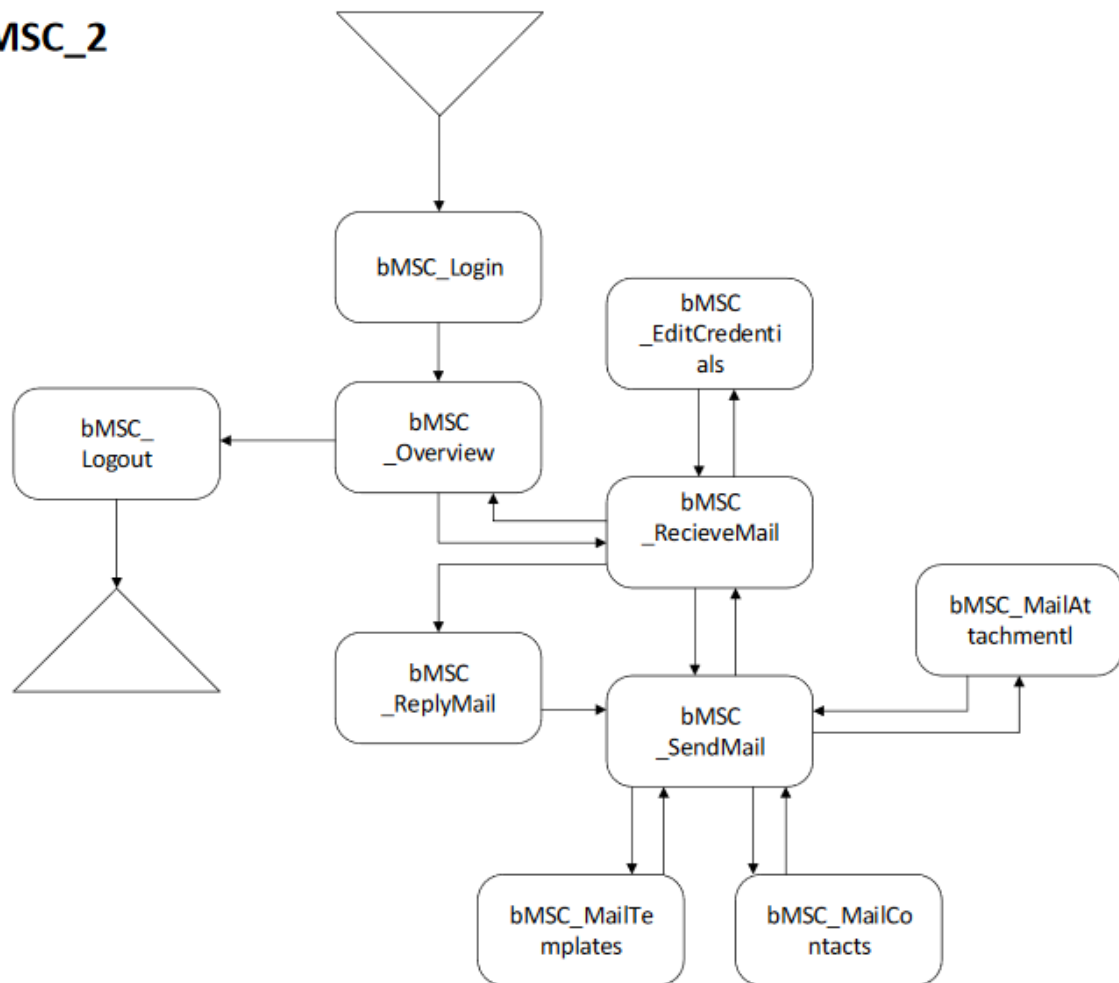


## Datenbank zurücksetzen





## hMSC\_2



# UML Klassendiagramm

Aus Platzgründen wurde das Klassendiagramm der III. Iteration im Ordner „Iteration3“ abgelegt.

## Funktionalitätsplanung

Id	Funktionalität	Geschätzter Aufwand	Verantwortlicher	Abhängige Funktionalitäten	Quellcode-referenz	Status
<b>1.</b>	<b>Dokumentation</b>					
1.1	Erstellen der User Stories	3h	Andreas, Tobias, Kaan, Merlin, Johannes, Dave	bMSCs, hMSC		fertig
1.2	Erstellen des hMSCs	2h	Andreas, Tobias	bMSCs		fertig
1.3	Erstellen der bMSCs	3h	Johannes, Merlin, Tobias, Kaan, Dave, Andreas			fertig
1.4	Erstellen des Klassendiagramms	2h	Merlin, Tobias			fertig
1.5	Funktionalitätsplanung	2h	Andreas, Tobias, Kaan			fertig
1.6	Erstellen der Präsentation	2h	Johannes			fertig
<b>2.</b>	<b>Datenbank</b>					
2.1	Reset-Funktionalität	3h	Merlin	Testdaten	DangerousStufController	fertig
2.2	Testdaten	2h	Merlin, Andreas		DBUtils	fertig
<b>3.</b>	<b>Controller</b>					
<b>3.1</b>	Gitlab Login	8h	Andreas	HomeScreen	GitlabLogin	fertig
<b>3.2</b>	Gitlab Tortendiagramm	8h	Andreas	HomeScreen, GitlabLogin	GitlabChart	fertig
<b>3.3</b>	Gitlab Graphen	8h	Andreas	HomeScreen, GitlabLogin	GitlabChart	fertig
<b>3.4</b>	Mailverkehr	16h	Tobias	-	SendMail, ReadMail, RecieveMail	fertig
<b>3.5</b>	Mailtemplates	6h	Tobias	Mailverkehr	MailTemplate, CreateMailTemplate	fertig
<b>3.6</b>	Mailkonfiguration	12h	Johannes	Mailverkehr	CreateMailCredentials, EditMailCredential	fertig
<b>3.7</b>	Mail Kontakte	8	Johannes	Mailverkehr	MailContacts	fertig

<b>3.8</b>	Mailanhang	12h	Johannes	Mailverkehr	MailAttachments	fertig
<b>3.9</b>	Notizenhistorie	10h	Kaan	Homescreen, Notizliste	NotesHistoryController, NotesHistoryShowNoteController	fertig
<b>3.10</b>	Komplette Notizliste	6h	Kaan	Homescreen, Notizliste	AllNotesController	fertig
<b>3.11</b>	Suchfunktion zu Notizen	3h	Kaan	Komplette Notizliste	AllNotesController	fertig
<b>3.12</b>	Prüfungsvorlagen	16h	Dave	Homescreen	ExamStudent, ExamGroup	fertig
<b>3.13</b>	Passwortmanagement	8h	Merlin	Login, Mail	HashUtils, PasswordModal	fertig
<b>3.14</b>	Homeview anpassen und einbetten	2h	Merlin, Andreas	GitTab, Mail	Week-Calendar-Controller	fertig

## Unittests

<b>Id</b>	<b>Getestete Funktionalität</b>	<b>Quellcode Referenz</b>	<b>Status</b>
<b>4</b>	Speicherung eines Studenten	DBTest	bestanden
<b>5</b>	AES Verschlüsselung	UtilsTest	bestanden
<b>6</b>	String/Date-Konvertierung	UtilsTest	bestanden

## Systemtests

<b>Datum</b>	23.01.2019		
<b>Tester</b>	Semra Burschik		
<b>SW-Version</b>	Commit vom 23.01.2019 „Additional filter functions“		
<b>Vorbedingung(en)</b>	Mail Einstellungen vorhanden.		
<b>Schritt</b>	<b>Aktion (User)</b>	<b>Erwartete Reaktion (System)</b>	<b>✓ / X</b>
1	Der Benutzer klickt auf „Email verfassen“.	Das System öffnet ein neues Fenster.	✓
3	Der Benutzer klickt auf „senden“ .	Das System sendet die Email und gibt eine Bestätigung aus.	✓
<b>Nachbedingung(en)</b>	Das System befindet im Email-Fenster.		✓
<b>Testurteil</b>	Bestanden.		

<b>Datum</b>	23.01.2019		
<b>Tester</b>	Semra Burschik		
<b>SW-Version</b>	Commit vom 23.01.2019 „Additional filter functions“		
<b>Vorbedingung(en)</b>	Mail Einstellungen vorhanden.		
<b>Schritt</b>	<b>Aktion (User)</b>	<b>Erwartete Reaktion (System)</b>	<b>✓ / X</b>
1	Der Benutzer klickt auf „Emails abrufen“.	Das System zeigt die eingegangenen Emails an.	✓
<b>Nachbedingung(en)</b>	Das System befindet im Email-Fenster.		✓
<b>Testurteil</b>	Bestanden.		

<b>Datum</b>	23.01.2019		
<b>Tester</b>	Semra Burschik		
<b>SW-Version</b>	Commit vom 23.01.2019 „Additional filter functions“		
<b>Vorbedingung(en)</b>	Das System befindet sich im Email-Fenster.		
<b>Schritt</b>	<b>Aktion (User)</b>	<b>Erwartete Reaktion (System)</b>	<b>✓ / X</b>
1	Benutzer öffnet das Kontaktfenster.	System öffnet Kontaktfenster.	✓
2	Benutzer wählt eine Klasse aus.	System fügt Klasse als Empfänger hinzu.	✓
<b>Nachbedingung(en)</b>	Das System befindet sich im Email-Fenster.		✓
<b>Testurteil</b>	Bestanden.		

# Nutzerhandbuch

## Technische Anforderungen

Mindestanforderungen:

CPU: Ryzen 1300 oder vergleichbares

RAM: 4 GB

Festplattenspeicher: 2 GB freier Speicher

Grafikkarte: keine Anforderungen

Betriebssystem: Windows 10

Monitor, Tastatur, Maus

JAVA Version 11

PostgreSQL-Datenbank Version 10.1 oder neuer

Für Mail-Funktionalität: Gmail Account

Eine stabile Internetverbindung mit mind. 50/16 Mbit Down/Upload

# Installationsanleitung

## 1. Datenbankinstallation:

Um die Datenbank des SemestervErwaltungsPlans zu installieren muss zuvor eine PostgreSQL-Datenbank vorhanden sei, auf die die Anwendung zugreifen kann.

1. Starten Sie die Anwendung über die Kommandozeile mit dem Befehl „java -jar SEP\_Gruppe\_E-Iteration-3.0“. Die Anwendung wird eine Konfigurationsdatei generieren und eine Fehlermeldung anzeigen.
2. Die Anwendung hat nun eine Konfigurationsdatei „sep.properties“ im Arbeitsverzeichnis generiert. Passen Sie die Einstellungen db\_host, db\_port, db\_db, db\_user und db\_password mit den entsprechenden Parametern an.
3. Starten Sie die Anwendung erneut über die Kommandozeile um zu testen, ob die Verbindung mit der Datenbank erfolgreich ist. Wenn die Verbindung erfolgreich ist sollte die Fehlermeldung „ERROR: Die Datenbank ist nicht installiert“ erscheinen.
4. Starten Sie die Anwendung mit dem parameter „setup“ (z.B. „java -jar SEP.jar setup“)
5. Die Anwendung erstellt nun die benötigte Datenbankstruktur und fügt Beispieldaten ein.
6. Sie können die Anwendung nun wie gewohnt starten und sich mit einem der Standardbenutzer anmelden.

### HINWEIS:

Die Standardbenutzer haben die Namen „besttutor“ und „moretutor“. Die Passwörter für beide Benutzer lauten „changeme“.

## 2. Einrichtung des System-Mailaccounts

Damit die Anwendung E-Mails versenden kann (zum Beispiel, wenn ein Benutzer sein Passwort vergessen hat, müssen die folgenden Einstellungen vorgenommen werden:

1. Die Datenbank muss installiert sein.
2. In der Konfigurationsdatei „sep.properties“ müssen die Einstellungen sys\_mail\_user, sys\_mail\_password, sys\_mail\_host und sys\_mail\_smtp\_port angepasst werden.

### HINWEIS:

Manche Antivirenprogramme blocken die Mailfunktionalität der Anwendung.



# Bedienungsanleitung

## Login:

Hier kann sich der Benutzer mit seinen Anmeldedaten anmelden. Die Standardpasswörter für die Benutzer „moretutor“ und „besttutor“ sind „changeme“.

Über den „Passwort zurücksetzen“-Button kann das Passwort zurückgesetzt, woraufhin der Benutzer eine Email mit einem neuen Passwort erhält.

## HINWEIS:

Wenn das Passwort eines Benutzers zurückgesetzt wurde, muss man in der Benutzerverwaltung das Email-Passwort des Benutzers neu vergeben, bevor die Email-Funktionalität wieder verfügbar ist.

## Registrieren:

Um die Software nutzen zu können, muss sich jeder Benutzer registrieren.

Die Registrierung findet im „Login“ Fenster statt. Mit Betätigung des „Registrierung“ Buttons öffnet sich ein neues Fenster. In diesem Fenster muss der Benutzer seine Informationen eintragen. Hierfür benötigt der Benutzer einen Einwahlcode, der von einem anderen Benutzer geschickt werden muss.

## HINWEIS:

Um die Email Funktionen der Software zu verwenden, muss eine Gmail Adresse verwendet werden.

Um die Email Funktion des Systems erfolgreich nutzen zu können, muss bei dem Gmail Account der Zugriff weniger sicherer Apps aktiviert werden:

1. Melden Sie sich in der Google Admin-Konsole an.
2. Klicken Sie auf Sicherheit --> Grundlegende Einstellungen.
3. Wählen Sie unter „weniger sichere Apps“ die Option „Einstellungen für weniger sichere Apps aufrufen“ aus.
4. Klicken Sie im eingeblendeten Fenster auf das Optionsfeld „Nutzern erlauben“, den Zugriff auf weniger sicherer Apps zu verwalten.

Sobald diese Einstellung übernommen wurde, kann das System die Emails empfangen.

Sollte eine Two-Factor Authentication verwendet werden, so muss ein spezielles Passwort für unsichere Apps erstellt werden. Dafür muss man sich in den Google Account einloggen ( <https://myaccount.google.com/> ) und auf Einstellungen gehen. Unter dem Reiter Sicherheit ( <https://myaccount.google.com/security> ) kann ein App-Passwort vergeben werden. Unter „Bei Google anmelden“ kann das App-Passwort generiert werden. Dafür auf „App-Passwort“ klicken und unter „App“ Email auswählen, dann im danebenstehenden Reiter das Gerät auswählen. Nachdem auf „Weiter“ geklickt wurde, kann der App einen Namen gegeben werden.

Nach Bestätigung des Namens wird durch das System ein Passwort generiert.

Dieses Passwort kann nun in den Email Einstellungen der Software als „Passwort“ verwendet werden.

Sollte es hierbei Probleme geben, bitte schreibe eine Mail an: [tobias.sieber@mail.de](mailto:tobias.sieber@mail.de) oder rufe an unter 0176 436 969 84.

Alternativ kann unter <https://myaccount.google.com/security> auch die Two-Factor Authentication ausgeschaltet werden.

## Hauptbildschirm:

Im Hauptbildschirm ist auf der linken Seite eine Baumstruktur, welche Semester, Klassen, Gruppen und Studenten beinhaltet. Im rechten Teil gibt es Tabs für die Inhalte: Details, Notizen, Wochenplan, Semesterplan und GitLab. Durch eine Auswahl in der Baumstruktur werden die angezeigten Inhalte in den Tabs entsprechend angepasst.

Am unteren Fensterrand befinden sich Buttons zum Anlegen und Löschen von Elementen in der Baumstruktur. Zudem gibt es Buttons zum Bewerten, zum Versenden von Mails, zum Chatten, zum Betreten der Benutzerverwaltung und zum Verlassen der Anwendung. Diese Funktionen werden in ihren jeweiligen Abschnitten näher erläutert

## Benutzerverwaltung:

Mit der Betätigung des Buttons „Benutzerverwaltung“ öffnet sich ein neues Fenster mit fünf verschiedenen Buttons:

- **Dangerous Stuff:** Mit Betätigung des Buttons „Dangerous Stuff“ öffnet sich ein neues Fenster. In diesem Fenster kann die Datenbank zurückgesetzt werden.
- **Bearbeiten:** Über den Button „Bearbeiten“ können Benutzerinformationen bearbeitet werden.  
Zuerst muss der Benutzer ausgewählt werden. Dann öffnet sich mit der Betätigung des Buttons „Bearbeiten“ ein neues Fenster. In diesem Fenster können die Angaben des Benutzers angepasst werden. Der Username kann allerdings nicht verändert werden. Weiterhin können die Email Einstellungen verändert werden, indem der Button „Einstellung“ betätigt wird.  
Zuletzt kann der Benutzer sein Passwort ändern, indem der Button „Passwort ändern“ betätigt wird. Hier kann unter Verwendung des ehemaligen Passwortes ein neues erstellt werden.
- **Löschen:** Mit Betätigung dieses Buttons wird der ausgewählte Benutzer gelöscht. Vor der endgültigen Löschung muss, durch den Nutzer, bestätigt werden, dass der ausgewählte Benutzer gelöscht werden soll.
- **Einwahlcode:** Über den Button „Einwahlcode“ öffnet sich ein neues Fenster in welchem ein neuer Einwahlcode generiert werden kann.  
Über „Code generieren“ wird ein neuer Code generiert und mittels „Zwischenablage“ kann der ausgewählte Code in die Zwischenablage gespeichert werden.  
**Wichtig:** Jeder Einwahlcode kann nur einmal verwendet werden.
- **Zurück:** Schließt die Benutzerverwaltung.

## **GitLab:**

Nach dem ersten Klick auf das GitLab-Tab wird der Benutzer zum Login mit seinen Anmeldedaten von GitLab aufgefordert. Nach erfolgreichem Login werden die Statistiken, je nach gewählter Unterrubrik, in Relation zueinander oder im zeitlichen Verlauf angezeigt. Durch eine Auswahl in der Baumstruktur werden das angezeigte Tortendiagramm bzw. der Graph angepasst.

## **Logout:**

Beim geordneten Beenden der Anwendung werden die aufgeklappten Elemente und die Auswahl in der Baumstruktur sowie das ausgewählte Tab gespeichert, um beim nächsten Login in die Anwendung automatisch wiederhergestellt zu werden.

## **Anwesenheit:**

Im Semesterplan kann über den Button „Anwesenheit“ das Anwesenheitsfenster geöffnet werden. Hierzu muss zuvor eine Klasse aus der zugehörigen ComboBox und ein Termin aus dem Semesterplan ausgewählt werden. Anschließend wird die Anwesenheit für alle Mitglieder der Klasse an diesem Termin angezeigt.

Der Anwesenheitsplan enthält die folgenden Felder: Vorname, Nachname, Anwesend, Fehlzeit (min) und Entschuldigt.

Vorname und Nachname zeigen den Namen des Studenten und sind nicht bearbeitbar.

In die Fehler „Anwesend“ und „Entschuldigt“ können die Werte „Ja“ oder „Nein“ eingetragen werden. In das „Fehlzeit (min)“ kann die Fehlzeit des Studenten angezeigt werden.

## **Wochenplan:**

Im Wochenplan können beliebige Termine erstellt und verwaltet werden. Ein Wochenplan gilt immer pro Klasse. Durch einen Doppelklick kann ein neuer Eintrag erstellt werden. Durch Anklicken eines Eintrags kann dieser bearbeitet werden, alternativ ist dies auch per Rechtsklick -> Information möglich.

Einträge können beliebig durch Drag-and-Drop verschoben werden.

## Chatfunktion:

Das Chatfenster hat auf der rechten Seite eine Liste aller Benutzer, auf der linken Seite öffnen sich die Tabs mit den jeweiligen Chats. Es können mehrere Chats gleichzeitig geöffnet sein. Falls ein Chatter nicht im Vordergrund steht, so bekommt der Nutzer eine Benachrichtigung, wenn er eine Nachricht erhält. Das Chatfenster kann oben rechts geschlossen werden.

Um einen Chat zu starten, kann man auf einen Benutzernamen in der Liste doppelklicken, oder einen Namen markieren und den „Chat starten“-Button betätigen.

Weiterhin kann der Gesprächsverlauf zwischen dem ausgewählten und dem eingeloggten Nutzer exportiert werden. Darauf klickt man auf den „History exportieren“-Button, nachdem ein Benutzer ausgewählt wurde. Daraufhin öffnet sich ein Filechooser, mit dem man den Speicherort und den Namen der Textdatei mit dem Gesprächsverlauf festlegt.

Innerhalb eines Chatfensters wird im oberen Teil der aktuelle Gesprächsfaden angezeigt. Im unteren Teil kann die zu sendende Nachricht geschrieben werden. Um die Nachricht dann zu senden, kann der „Senden“-Button, oder die Eingabetaste getätigt werden. Der jeweilige Chat kann mit „Abbrechen“ oder im Tab am oberen Rand geschlossen werden.

## Notizfunktion:

Die Notizfunktion erlaubt dem Nutzer, Notizen zu „Klasse“, „Gruppe“ oder „Schüler“ zu erstellen. Die Notizfunktion ist in das Hauptfenster integriert und wird durch einen Klick auf den Reiter „Notizen“ aufgerufen. Da Notizen zu bestimmten Entitäten existieren, muss vor oder nach dem Klick auf den Reiter, eine Entität des Typs „Klasse“, „Gruppe“ oder „Schüler“ ausgewählt werden.

Das Notizfenster bietet eine Übersicht aller Notizen der ausgewählten Entität in Form einer Liste. Notizen können ebenso priorisiert werden. Die Priorisierung ist in allen Fenstern, welche Notizen darstellen, farblich erkennbar. Grün steht für „Gut“, gelb steht für „Mittel“, rot steht für „Schlecht“ und grau steht für „Ohne Zuordnung“.

Das Notizfenster bietet folgende Buttons:

- **Anzeigen:** Bei einem Klick auf diesen Button wird die ausgewählte Notiz angezeigt. Es öffnet sich ein Fenster mit nicht-editierbarem Notizinhalt, der Bezeichnung sowie der Priorität rechts in Textform als auch farblich. Dieses Fenster kann mit einem Klick auf den „Schließen“ Button geschlossen werden.
- **Neu:** Mit diesem Button können Notizen für die ausgewählte Entität erstellt werden. Ein Klick auf diesen Button öffnet ein neues Fenster, worin Bezeichnung, Inhalt sowie Priorität festgelegt werden können. Der „Speichern“ Button speichert die konfigurierte Notiz. Der „Abbrechen“ Button schließt das Fenster.
- **Editieren:** Der „Editieren“ Button ermöglicht das Bearbeiten der ausgewählten Notiz. In diesem Fenster können die Bezeichnung, der Inhalt sowie die Priorität geändert werden. Ein Klick auf den „Speichern“ Button speichert die Veränderungen und hält den Inhalt sowie Datum und Änderer in der Historie fest. Der „Abbrechen“ Button schließt das Fenster.
- **Löschen:** Der „Löschen“ Button löscht die ausgewählte Notiz aus der Datenbank.
- **Historie:** Der „Historie“ Button öffnet ein Fenster mit allen Notizen, die verändert wurden. Ein Klick auf den „Anzeigen“ Button öffnet ein neues Fenster, wo Notizinhalt sowie Änderer

und Änderungsdatum eingetragen sind. Dieses Fenster kann mit „Schließen“ geschlossen werden. Das Historienfenster kann ebenso mit einem Klick auf dessen „Schließen“ Button geschlossen werden.

- **Komplette Notizliste:** Dieser Button öffnet eine Liste mit allen Notizen in der Datenbank. Dieses Fenster ermöglicht folgende zusätzliche Funktionen:
  - Filtern der Notizliste nach Bezeichnung und/oder Inhalt.
  - Filtern nach dem Objekt bzw. der Entität.
  - Filtern nach Prioritäten.
  - Beliebige Kombination der drei Filterfunktionen.
  - Ein Klick auf den „Filtern“ Button wendet die Filtereinstellungen an.
  - Der „Anzeigen“ Button zeigt die ausgewählte Notiz an.
  - Der „Aktualisieren“ Button ermöglicht das aktualisieren der Liste.
  - Der „Editieren“ Button ermöglicht das editieren der ausgewählten Notiz.
  - Der „Schließen“ Button schließt das „Komplette Notizliste“ Fenster.

## Semesterplan:

Um einen Semesterplan erstellen bzw. bearbeiten zu können, muss ein Semester ausgewählt werden.

Anhand des Semesterplans ist es möglich eine Übersicht, über die gesamte Veranstaltung SEP zu erhalten. Diese Übersicht beinhaltet folgendes:

- **SEP-Termine:** Termine an denen SEP-Veranstaltungen stattfinden
- **KW:** Die Kalenderwoche in der die SEP-Veranstaltungen stattfinden
- **Termin-Nr.:** Die fortlaufende Anzahl an SEP-Veranstaltungen
- **Iteration:** Die Iteration in der sich die Gruppen befinden
- **Vorlesungsinhalte:** Die Inhalte der Folien, die der Betreuer während einer SEP-Veranstaltung vorträgt
- **Arbeitsphase:** Anstehende Arbeitsphase der Gruppen, bspw. das Erstellen von User Stories.

Die Buttons besitzen folgende Funktionen:

- **Hinzufügen:** Fügt der Tabelle eine neue Reihe hinzu.
- **Löschen:** Löscht eine Reihe einer Tabelle.
- **Anwesenheit:** Öffnet den Anwesenheitsplan

## Bewertungsformular:

Das Bewertungsformular dient als Unterstützung der Zwischen- und Endabnahmen. Es gibt auf Gruppen- und Studierendenebene jeweils ein Bewertungsformular.

### Gruppenebene:

Zur Erstellung und Bearbeitung des Formulars bedarf es, die Auswahl einer Gruppe aus der zugehörigen ComboBox.

Darauffolgend ist es nun möglich in einer tabellarischen Übersicht allgemeine Fragen, die eine Gruppe als ganzes betreffen, einzufügen. Des Weiteren können diese Fragen mit Ja oder Nein beantwortet werden und bei Bedarf, ist es möglich, zu jeder Frage ein Kommentar abzugeben.

Anhand eines Countdowns ist es möglich zu überprüfen, ob eine Gruppe bei ihrer Präsentation eine vorgegebene Zeit einhält.

Die Buttons besitzen folgende Funktionen:

- **Hinzufügen:** Fügt der Tabelle eine neue Reihe mit einer Frage, Ja/Nein-Antwort und Kommentar hinzu.
- **Löschen:** Löscht eine Reihe einer Tabelle.
- **Set:** Nach der Eingabe von Minuten und Sekunden in den entsprechenden Textfeldern wird die Zeit für den Countdown festgelegt.
- **Start:** Startet den Countdown.
- **Pause:** Pausiert den Countdown.
- **Fragerunde:** Öffnet das Bewertungsformular der Studierenden einer Gruppe

#### **Studierendenebene:**

Ausgehend von einem Bewertungsformular einer Gruppe, öffnet sich das Bewertungsformular der Studierenden dieser Gruppe.

Zur Erstellung und Bearbeitung des Formulars bedarf es, die Auswahl eines Studierenden aus der zugehörigen ComboBox.

Darauffolgend ist es nun möglich in einer tabellarischen Übersicht Fragen, an die jeweiligen Studierenden, einzufügen. Des Weiteren wird jede Antwort zu einer Frage bewertet und optional können Kommentare zu den Fragen oder Antworten der Studierenden eingetragen werden.

Es kann eine Auswertung der Bewertungen vollzogen werden, indem der Durchschnitt aller Bewertungen berechnet wird. Die Bewertungen basiert auf einer Skala von Dezimalzahlen zwischen 1-5, wobei 1 die schlechteste Bewertung und 5 die beste Bewertung ist, die abgegeben werden kann. Sollte der Durchschnitt einen Wert von mindestens 2,5 erreicht haben, ist die Prüfung bestanden. Andernfalls ist der Studierende durchgefallen.

Anhand eines Countdowns ist es möglich zu überprüfen, ob ein Studierender beim Codereview eine vorgegebene Zeit einhält.

Die Buttons besitzen folgende Funktionen:

- **Hinzufügen:** Fügt der Tabelle eine neue Reihe mit einer Frage, Bewertung und Kommentar hinzu.
- **Löschen:** Löscht eine Reihe einer Tabelle.
- **Set:** Nach der Eingabe von Minuten und Sekunden in den entsprechenden Textfeldern wird die Zeit für den Countdown festgelegt.
- **Start:** Startet den Countdown.
- **Pause:** Pausiert den Countdown.
- **Auswerten:** Errechnet den Durchschnitt alle Bewertungen der Fragen und gibt anschließend an, ob der erreichte Durchschnitt zum Bestehen der Prüfung ausreicht oder nicht.

## Emails:

Beim Starten der Email-Funktion muss zunächst das Passwort erneut eingegeben werden. Wurde die Email-Funktion das erste Mal aufgerufen, muss man zunächst seine Verbindungsdaten eingeben, um seinen Email-Client einzurichten.

Danach öffnet sich der Email-Posteingang. Hier sind alle Emails aufgelistet, die sich im Posteingang des Clients befinden.

Die Buttons besitzen die folgenden Funktionen:

- **Schließen:** Die Email Anwendung wird geschlossen.
- **Einstellungen:** Hier können die Verbindungsdaten geändert werden.
- **Antworten:** Auf eine im Posteingang markierte Email in einem neuen Fenster antworten.
- **Refresh:** Der Posteingang wird neu geladen und neue Email werden abgerufen.
- **Mail verfassen:** Eine neue Email schreiben. Wird in einem neuen Fenster geöffnet.

## Verbindungsdaten ändern:

Hier können die neuen Verbindungsdaten eingegeben und gespeichert werden. Die Bearbeitung kann durch „Abbrechen“ geschlossen werden.

## Antworten:

Ein neues Fenster öffnet sich, um eine Antwortmail zu senden. Der Sender der Ursprungsmail wird als Empfänger gesetzt, der Betreff beinhaltet „Awd:“ und den Ursprungsbetreff. Weiterhin wird der ursprüngliche Text zitiert. Alle Felder können immer noch verändert werden. Für die weiteren Funktionen, siehe „Mail verfassen“.

## Mail verfassen:

Ein neues Fenster öffnet sich. Hier müssen Empfänger und Betreff in das obere, bzw. zweite Textfeld eingetragen werden. Der Text der Mail wird in das große untere Textfeld eingetragen.

Die Buttons besitzen die folgenden Funktionen:

- **Kontakte:** Einen oder mehrere Empfänger in einer Baumstruktur auswählen.
- **Anhang:** Einen oder mehrere Dateien als Anhang hinzufügen.
- **Abbrechen:** Das Fenster wird geschlossen.
- **Templates:** Eine vorgefertigte Email wird aus einer Liste ausgewählt und zum Bearbeiten geladen.
- **Entwurf:** Die Email wird als Template gespeichert.
- **Senden:** Die Email senden.

## Kontakte:

Nachdem sich das Kontaktfenster öffnet, können Klassen, Gruppen oder einzelne Studenten in der Baumstruktur ausgewählt werden. Die Markierung wird durch einen Doppelklick übernommen, oder durch den „Auswählen“-Button. Das Fenster kann über den „Abbrechen“-Button geschlossen werden.

## Anhang:

Hier kann durch den „Hinzufügen“-Button ein Filechooser geöffnet, über den man eine Datei (Ordner sind nicht erlaubt) auswählen kann, die als Anhang hinzugefügt werden soll. Diese Datei wird

dann in der Listenansicht angezeigt. Eine in der Liste markierte Datei kann über „Entfernen“ wieder entfernt werden. Um die Anhänge der Email hinzuzufügen, klickt man auf den „Übernehmen“-Button.

### **Templates:**

Im Fenster „Email verfassen“ befindet sich im unteren Bereich der Button „Templates“.

Mit Betätigung dieses Buttons öffnet sich das Fenster „Mail Templates“. Hier gibt es vier Buttons, die betätigt werden können:

- **Erstellen:** mit Betätigung dieses Buttons öffnet sich ein neues Fenster. In diesem Fenster kann ein neues Template erstellt werden. Sollte der Betreff oder der Inhalt leer sein, öffnet sich ein Pop-Up Fenster, welches darüber informiert, dass alle Felder besetzt werden müssen. Durch den Button „Speichern“ wird das Template gespeichert. Durch „Abbruch“, wird das Fenster geschlossen.
- **Löschen:** mit Betätigung dieses Buttons wird das ausgewählte Template gelöscht. Vor der endgültigen Löschung muss, durch den Nutzer, bestätigt werden, dass das Template gelöscht werden soll.
- **Benutzen:** mit Betätigung dieses Buttons wird das ausgewählte Template verwendet. Es öffnet sich das „Email verfassen“ Fenster und das Template wird übernommen.
- **Abbruch:** schließt das Fenster.

### **Entwurf:**

Im Fenster „Email verfassen“ befindet sich im unteren Bereich Button „Als Entwurf speichern“.

Mit Betätigung des Buttons wird der aktuelle Betreff und der Inhalt als Template gespeichert.

Sollte der Betreff oder der Inhalt der Email leer sein, so wird ein Pop-Up angezeigt, welches darüber informiert, dass alle Felder besetzt werden müssen.

### **Senden:**

Die Email wird hiermit gesendet. Sollte einer der gesetzten Email Adressen formal nicht korrekt sein, so wird ein Pop-Up Fenster angezeigt, welches darüber informiert, welche Email Adresse formal nicht korrekt ist.