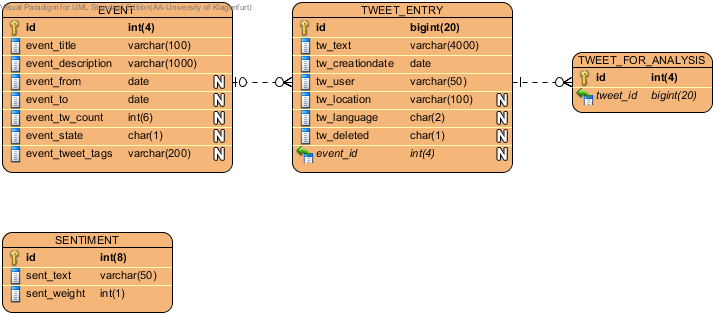
# DOCUMENTATION

## Database

### Entity Relationship Diagram

The following Entity Relationship Diagram shows all tables that were created in Sprint 1 and Sprint 2 (last update of table changes: 25. 04. 2014, 14:00).



Tweets can be collected for all created events. The table TWEET\_ENTRY contains all tweets that were collected for a certain event. The table TWEET\_FOR ANALYSIS marks the subset of all collected tweets that should be used for the analysis.

The table SENTIMENT is not used yet. It contains keywords that determine if the text of a certain tweet has a positive or negative meaning.

### Description of the tables

The table EVENT contains the information needed for each created event.

|  |  |  |
| --- | --- | --- |
| table EVENT | Anmerkungen | Beschreibung |
| id | AUTO\_INCREMENT | unique identifier of an event |
| event\_title |  | title of the event |
| event\_description |  | short description of the event |
| event\_from | DEFAULT NULL | date to specify the beginning of the tweet collection |
| event\_to | DEFAULT NULL | date to specify the ending of the tweet collection |
| event\_tw\_count | DEFAULT NULL | specifies the number of tweets to collect |
| event\_state | DEFAULT '0' | 0: initial state  1: collection in progress  2: collection finished |
| event\_tweet\_tags | DEFAULT NULL | all tweet tags which should be considered, separeted with a special delimiter |

The table TWEET\_ENTRY holds all collected tweets and their relevent information. Each tweet is assigned to an event because each collection of tweets is started for a certain event.

|  |  |  |
| --- | --- | --- |
| table TWEET\_ENTRY | Anmerkungen | Beschreibung |
| id | AUTO\_INCREMENT | unique identifier of a tweet |
| tw\_text |  | content/text of the tweet |
| tw\_creationdate |  | creation date of the tweet |
| tw\_user |  | user/author of the tweet |
| tw\_location | DEFAULT NULL | Location of the user/author |
| tw\_language | DEFAULT NULL | language of the tweet text |
| tw\_deleted | DEFAULT '0' | marks the tweet as deleted |
| event\_id |  | foreign key, each tweet is assigned to a certain event |

The entries in the table TWEETS\_FOR\_ANALYSIS marks which of the collected tweets should be considered in the analysis.

|  |  |  |
| --- | --- | --- |
| table TWEETS\_FOR\_ANALYSIS | Anmerkungen | Beschreibung |
| id | AUTO\_INCREMENT | unique identifier |
| tweet\_id |  | foreign key, id of the considered tweet |

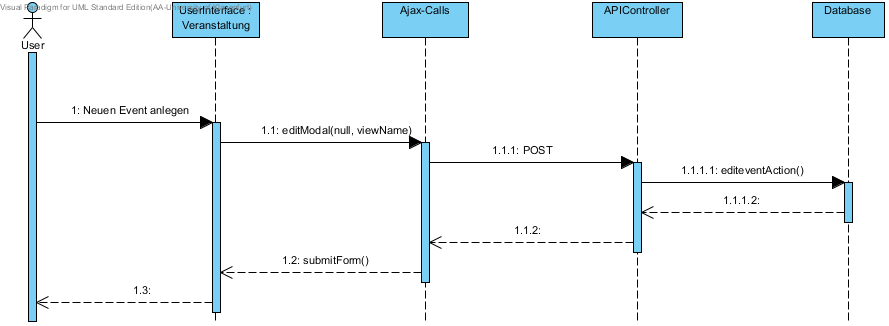
The table SENTIMENT is not used in Sprint 2 but was created as a preparation for a following sprint.

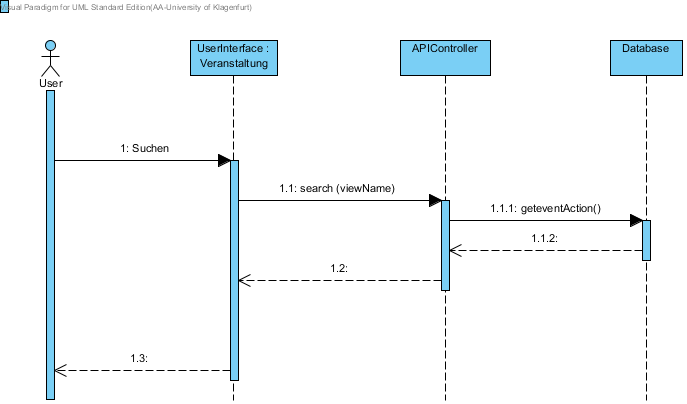
|  |  |  |
| --- | --- | --- |
| table SENTIMENT | Anmerkungen | Beschreibung |
| id | AUTO\_INCREMENT | unique identifier of a sentiment |
| sent\_text |  | keyword for sentiment analysis |
| sent\_weight |  | weight of the keyword |

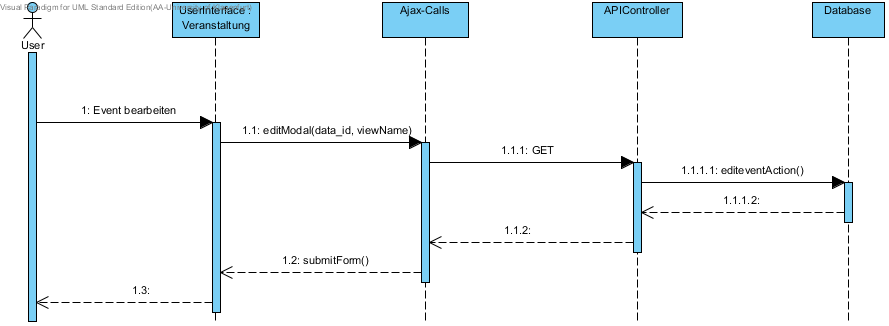
## Realized sequences

### Status after Sprint 1

Sprint 1 focussed on the management of events: new events can be created or existing events can be searched and edited:







### Status after Sprint 2

The collected tweets are shown in a list. It is possible to filter tweets for a certain event and to filter tweets by defining keywords.

