Fakebook Manual

Features (User):

- create posts with text and images

- create comments to posts

- reacting to posts with likes

- reacting to posts with dislikes

- report posts

- edit profile (change profile picture, **bio**)

- send friend invitations

- accept or decline friend invitations

- **see other profiles**

- **search other user**

- chat with other users

- click on advertisements

Features (Researcher):

- host the website (see chapter Setup)

- see user information (name, email, but NOT passwords)

- manage users (add, remove, edit)

- see all content (including posts, comments, likes, chats, friendlists, etc.)

- edit content (posts, profiles, comments, number of likes, dislikes, etc.)

- enable advertisements (with image, text and link)

- see the statistics of advertisements (number of total clicks, list of users who clicked on the advertisement)

- download all data as xlsx file with selected tables or the entire database

Setup

tba

User Management for researchers

Everyone can sign up the website by simply registering on the site.

After loggin in user can post, comment and have full access to all the features described in the chapter Features (User)

A researcher can log into the admin site by appending a „/admin“ to the url and loggin in with the superuser credentials that were set in the chaper Setup. This will lead to the admin homepage.

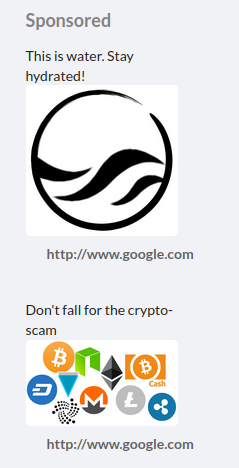
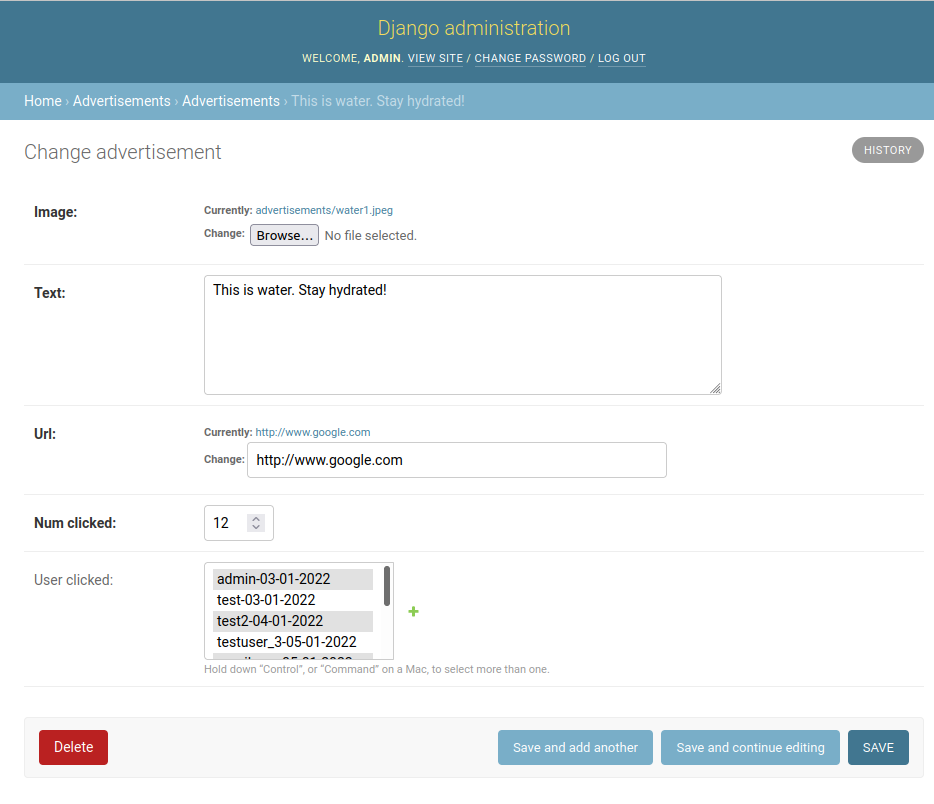
On the admin homepage all modules can be inspected and modified (add, remove, edit). Artificial users can be created and posts, comments and other activities can be done under their authorship. This can be done before and during the experiment.

This chapter describes all the sections of the admin‘s homepage.

Advertisements

The advertisement will be shown on the post site on the upper right corner. There is no limit on how many advertisements can be uploaded and they will all be shown in a vertical list with the oldest advertisement on the top. To keep the site clear it is recommended to upload no more than a few advertisements and to limit the text to no more than 35 characters. The pictures will keep their size ratio and the pictures should be chosen accordingly.

To add advertisements to the website go to Advertisements -> Add Advertisement and choose an image, a text and a URL.



Here is the description of the fields that can be edited:

Image: The image that will be shown on the website

Text: The text that will be displayed above the image

Url: The adress of the site that will be opened after clicking on the URL. The image and the text are no links.

Num clicked: The total amount of times this add has been clicked by users. The number will only be displayed on the admin website, not on the user website. It can be set to any value, but it is strongly advided against doing so because it will falsify the information about this ad.

User clicked: The list of all users that have registered. The users, who clicked on this advertisement have light grey background (in the example above the add was clicked on by admin-03-01-2022, test2-04-01-2022). IMPORTANT: To add another user hold down the Crtl key while clicking on a user to add or remove them from the list. If the Ctrl key is not pressed during the click all users will be removed from the list and only the newly clicked user is added to the list.

Authentication and Authorization -> Users

This list contains the user that are registered on the site. Note that this is not the list of profiles which are used to post, comment etc, however there is a link between the userlist and the list of profiles, e.g. the username is the same. A notable difference is that the email in the user entry is not visible to other users, but if users decide to enter an email adress to the profile that will be shown in the profile page.

Use this site to view the users and generate artifficial users that can be used by a researcher to post content.

The list gets automatically updated whenever a user registers and shows the username, which is the same as the username of their profile .

To add an user manually go to to Authentication and Authorization -> Users and click on „Add User +“ on the top right. Here only the username and the password can be set. For any further settings the user has to be edited after creation.

To edit an user, go to the user list and click on the username.

Here is a description of the fields that can be edited:

Username: The name of the user, it was initially chosen by the user at the registration at the site.

Password: For security reasons the password is not stored in cleartext, but in a hashed form. The hash algorithm, the amount of iterations, the salt and the corresponding hash of the password. Since the password is not stored directly it cannot be seen, however it can be reset by the linked form.

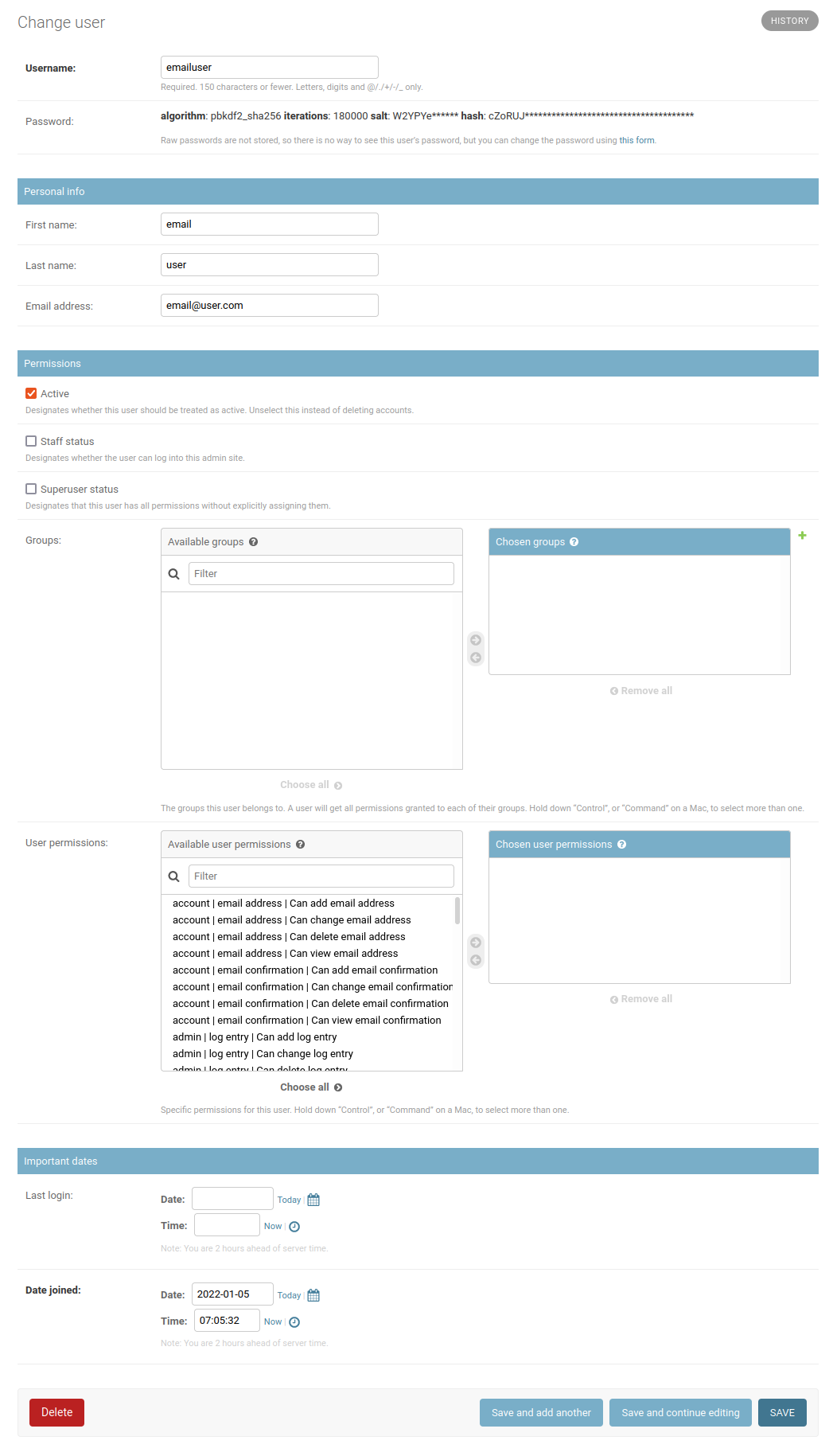
First name: The first name of the user

Last name: The last name of the user

Email address: The email address of the user

Beneath that is the section for the permissions that the user has. A regular user only has the lowest possible permissions, which only allow the user to use the site. To elevate the user to an superuser or staff, tick the corresponding box. A staff or superuser can access the admin homepage and perform all the actions that is permitted to their status.

Below the permission section the last login as well as the joined date can be seen and edited. This can be helpful to disguise the artificial as a normal user when other user check the last login date.



Chat

tba

Posts, Comments, Likes, Dislikes, Reports

This section contains all the content that the user post on the website. That includes the posts and the commens, as well as the reactions (likes, dislikes and reports)

Posts:

To see the list of posts go to Posts, Comments, Likes, Dislikes, Reports -> Posts. Here all the posts are listed, ordered by the creation date. To add a post click on the „Add Post +“ button on the top right.

To edit a post go to the post list and click on the ID of the post.

The post edit/creation fields are described here:

Content: The text of the post. Every post must have a text. It supports new lines, but not emojis, links or other text enrichments like bold or cursive text.

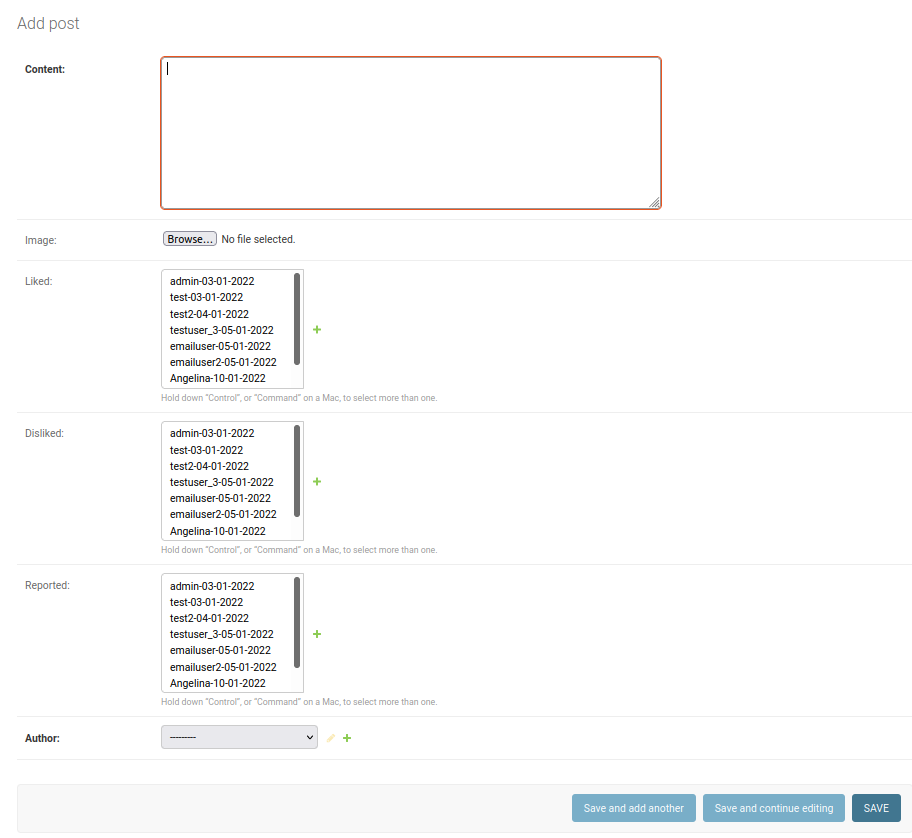
Image: To upload an image that will be shown below the text. Supported formats are .png, .jpg/.jpeg. There is no size limit to the pictures.

Liked: The list of users that liked the post. IMPORTANT: To add another user to the liked list hold down the Crtl key while clicking on a user to add or remove them from the list. If the Ctrl key is not pressed during the click all users will be removed from the list and only the newly clicked user is added to the list. When a user is added to the list (by either clicking on the tumbs up button on the site or by getting added by an administrator manually) an entry in the database is created with the value „Liked“ (The database is described in more detail in the section „Download Data“). When the user is logged on the corresponding posts are marked by a filled blue thumbs up. The user can click on that button again to revoke the like. In this case the entry in the database will be changed to „Unlike“. However, removing the Like manually as an administrator by editing the liked list will not create the database entry. To correctly remove the like and have the corresponding entry in the database go to Posts, Comments, Likes, Dislikes, Reports -> Likes.

The green plus right next to the Liked list can be used to create a new profile.

Dislikes and Reports: Dislikes and Reports function the exact same way as the Likes do.

Author: This is a drop down menu that shows the list of all profiles. To contribute a post to an author select the desired author. This can be used to create a post by an artificial user controlled by the researcher.



Comments:

To see the list of comments go to Posts, Comments, Likes, Dislikes, Reports -> Comments. To add a post click on the „Add Comment +“ button on the top right.

To edit a comment go to the comment list and click on the ID of the comment.

The comment edit/creation fields are described here:

User: The profile that created the comment.

Post: The post that the comment belongs to. Only the first few words of the posts are shown. To see more information about the post click on the pen button right next to the drop down menu. To create a new post that the comment will belong to click on the green plus button. For more information about how to create a post go to the section Posts.

Body: The text of the comment. It supports new lines, but not emojis, links or other text enrichments like bold or cursive text.

Like, Dislike, Report:

Like, Dislike and Reports are very similar in their application and are described here in one section. Likes, Dislikes and Reports are used to express the reaction of users to posts.

Likes, Dislikes and Reports have the following fields:

User: The profile that issued the reaction.

Post: The post that the reaction belongs to. User can not react to comments.

Value: This field indicates that the user reacted to the post. User can react to a post by clicking the thumbs up button to like, the thumbs down button to dislike and the flag to report. After reacting to the post an entry is created with the values Like/Dislike/Report accordingly. Also the button (thumbs up, thumbs down and the flag) will change from lined out to filled. The user can click the button again to undo the reaction. In this case the value will change to Unlike/Undislike/Unreport.

Profiles:

Profiles -> Profiles:

This list stores all the profiles. They are linked to a user. The difference from a profile to a user is that everything on the profile will be visible to other users on the website. Users can change some entries of their profile. To see all the profiles go to Profiles -> Profiles. To add a profile click on the „Add profile +“ button in the top right. To edit a profile click on the ID of the profile.

The profile edit/creation fields are described here:

First name: The first name of the profile. This field can be changed by the user.

Last name: The last name of the profile. This field can be changed by the user.

User: The user that the profile is linked to. This links the registering email to the user as well as the username. It can not be changed by the user.

Bio: A short text that can be changed by the user to write a short description of themselves.

Email: The email of the profile. It can not be changed by the user.

Country: The country of the profile. It can not be changed by the user.

Avatar: The profile picture of the user. When not changed there is a default profile picture.

Friends: A list of all users. If the user is a friend of the current profile the name will have a slight grey background. IMPORTANT: To add another user as friend hold down the Crtl key while clicking on a user to add or remove them from the list. If the Ctrl key is not pressed during the click all users will be removed from the list and only the newly clicked user is added to the list.

Slug: A unique identifier of the profile in string format. It is generated automatically and should not be changed. The site of the profile can be accessed by the following URL: <website-url>/profiles/<slug>.

Profiles -> Relationship:

This is a list of all friend requests. It shows the friend requests and their status, which is either „send“ or „accepted“.

To add another relationship go to Profiles -> Relationship and click on the „Add Relationship +“ button in the top right. To edit a relationship click on the ID of the relationship in the list.

A relationship has the following fields:

Sender: The profile that has send the friend request

Receiver: The profile that has received the friend request

Status: The status of the friend request. It can bei either „send“ or „accepted“. If the status is „accepted“ the two profiles will be regarded as friends, their number of friends field on their profile page will be updated as well as an entry in the database will be created.

Download Data

To download all the content and statistics go from the admin homepage an go to the Download section. There are two ways to download the data.

1. Select and download the tables as a .xlsx file

Select all the desired tables and click on the dowload button. Appendix A describes the tables with their columns as well as the sql statements that were used to create those tables from the original database.

1. Download the entire raw database

If the raw database is required and the technical knowledge is present the researcher can download the .sqlite

Click on the download button to download the database as .sqlite3.

Important: The structure of the database is not the same as in the .xlsx file. Appendix B details the database.

APPENDIX A

This appendix describes the tables and columns of the .xlsx file the researcher can download in the download section.

It also lists the SQL statements that were used to get the information from the database and create the tables for the .xlsx file. The database can also be downloaded in the download section.

Table User:

user\_id: The ID of the profiles. Note that there is a difference between the User and the Profile. Although the column is named user\_id it shows the ID of the profiles, as the main identifier of the uses will be with their profiles.

username: The username chosen by the user at the registration

first\_name: The first name of the profile

last\_name: The last name of the profile

avatar: The path to the profile picture. The profile pictures can either be downloaded as .zip as described in the section „Download Data“ or accessed directly on the server in the folder „data/media/profile\_pictures“

bio: The bio of the profile. This is a text written by the user, usually for used for a self-desctiption.

date\_joined: The date that the user joined the website

last\_login: The last time that the user was logged in.

SQL statement:

SELECT profiles\_profile.id AS "user\_id", auth\_user.username, profiles\_profile.first\_name, profiles\_profile.last\_name, profiles\_profile.avatar, profiles\_profile.bio, auth\_user.date\_joined, auth\_user.last\_login

FROM profiles\_profile, auth\_user

WHERE profiles\_profile.user\_id = auth\_user.id

Table friends:

sender\_id: the profile ID of the profile that has sent the friend request. It matches the ID in the table „user“

receiver\_id: the profile ID of the profile that has received the friend request. It matches the ID in the table „user“

status: the status of the friend request

updated: The time of the last update of the relationship. This occurs when the friend request has been accepted.

created: The time the friend request has been sent

SQL statement:

SELECT sender\_id, receiver\_id, status, updated, created

FROM profiles\_relationship

Table chats:

tba

Table posts:

post\_id: The unique identifier of the post

author\_id: The ID of the profile that has created the post. It matches theuser\_id in the table „user“.

content: The text of the post.

created: The time the post has been created.

image: The path to the image, if a picture has been uploaded in the post. The images of the posts can either be downloaded as .zip as described in the section „Download Data“ or accessed directly on the server in the folder „data/media/posts“.

comments: The ammount of comments on this post

likes\_received: The ammount of likes this post has received by users

dislikes\_received: The ammount of dislikes this post has received by users

reports\_received: The ammount of reports issued by users

SQL statement:

SELECT posts\_post.id AS "post\_id", posts\_post.author\_id, posts\_post.content, posts\_post.created, posts\_post.image,

(SELECT COUNT(\*)

FROM posts\_comment

WHERE posts\_post.id = posts\_comment.post\_id) AS "comments",

(SELECT COUNT(\*)

FROM posts\_post\_liked

WHERE posts\_post.id = posts\_post\_liked.post\_id) AS "likes\_received",

(SELECT COUNT(\*)

FROM posts\_post\_disliked

WHERE posts\_post.id = posts\_post\_disliked.post\_id) AS "dislikes\_received",

(SELECT COUNT(\*)

FROM posts\_post\_reported

WHERE posts\_post.id = posts\_post\_reported.post\_id) AS "reports\_received"

FROM posts\_post

Table comments:

comment\_id: Unique identifier of the comment

post\_id: ID of the post that this comment belongs to. This matches the ID in the post\_id in the table „posts“.

user\_id: The ID of the profile that has created the post. It matches the user\_id in the table „user“.

content: The text of the comment

created: The time the comment was created.

updated: The last time the comment was updated.

SQL statement:

SELECT id AS "comment\_id", post\_id, user\_id, body AS "content", created, updated

FROM posts\_comment

Table likes:

post\_id: ID of the post that this like belongs to. This matches the ID in the post\_id in the table „posts“.

user\_id: The ID of the profile that has liked or revoked the like of the post. It matches the user\_id in the table „user“.

value: „Like“ if the user has liked the post. „Unlike“ if the user has revoked the like of the post.

created: The time the like has been created.

updated: The last time the like has been updated

SQL statement:

SELECT post\_id, user\_id, value, updated, created

FROM posts\_like

Table dislikes:

post\_id: ID of the post that this dislike belongs to. This matches the ID in the post\_id in the table „posts“.

user\_id: The ID of the profile that has disliked or revoked the dislike of the post. It matches the user\_id in the table „user“.

value: „Dislike“ if the user has disliked the post. „Undislike“ if the user has revoked the dislike of the post.

created: The time the dislike has been created.

updated: The last time the dislike has been updated

SQL statement:

SELECT post\_id, user\_id, value, updated, created

FROM posts\_dislike

Table reports:

post\_id: ID of the post that this report belongs to. This matches the ID in the post\_id in the table „posts“.

user\_id: The ID of the profile that has reported or revoked the report of the post. It matches the user\_id in the table „user“.

value: „Reported“ if the user has reported the post. „Unreport“ if the user has revoked the report of the post.

created: The time the repport has been created.

updated: The last time the report has been updated

SQL statement:

SELECT post\_id, user\_id, value, updated, created

FROM posts\_report

Table advertisements:

This table differs from all the other tables since it list the advertisement multiple times; one time for every unique user that has clicked on the advertisement. The column „num\_clicked“ shows the ammount of times the advertisement has been clicked in total. There is no way to see which user has clicked how many times on the add

advertisement\_id: The ID of the advertismenent

text: The text of the advertisement

url: The URL that the advertisement leads to when clicked on the site.

num\_clicked: The total amount of times the advertisement has been clicked.

image: The path to the image. The images of the advertisements can either be downloaded as .zip as described in the section „Download Data“ or accessed directly on the server in the folder „data/media/advertisements“.

profile\_id: The IDs of the profiles that have clicked on the advertisement. It matches the user\_id in the table „user“. Every user is listed only once. Multiple clicks on the advertisement will not be accounted for other than in the increased number in „num\_clicked“.

SQL statement:

SELECT advertisement\_id, text, url, num\_clicked, image, profile\_id FROM advertisements\_advertisement, advertisements\_advertisement\_user\_clicked

WHERE advertisements\_advertisement.id = advertisement\_id"