# 1 Introduction

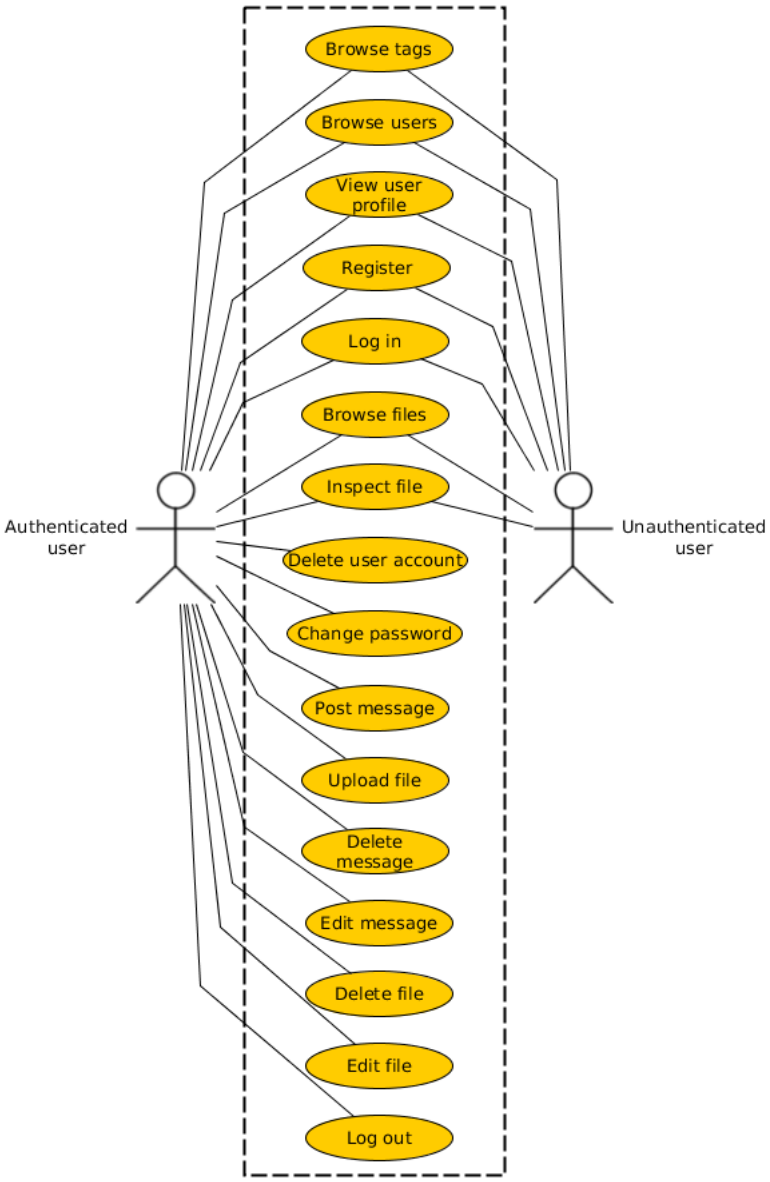
## 1.1 Project description

This project is a web application that enables users to upload files to the server and post comments on them. Uploaded files and messages are available for anyone to view. Uploading and posting messages requires an user account. Registered users can later edit or delete their uploaded files and messages.

## 1.2 Implementation environment

The application is implemented in PHP and runs on the University of Helsinki users-server. It uses a PostgreSQL database and Bootstrap css.

# 2 Use cases




## 2.1 User classes

**Unauthenticated user** Anyone on the web.

**Authenticated user** Someone who has created an account on the server and is currently logged in on that account.

## 2.2 Use cases by user class

### 2.2.1 Unauthenticated user

Unauthenticated users can do the following:

**Browse files** The user lists either all files on the server, or some subset of them by using the search function.

**Inspect file / browse messages** The user selects one file and is shown detailed information about it, in addition to any comments posted on that file.

**Browse users** The user lists all user accounts.

**View user profile** The user views detailed information about a single user account.

**Browse tags** The user lists every tag currently in use in the system.

**Register** The user creates an account on the service.

**Log in** The user logs in to an existing account on the server.

### 2.2.2 Authenticated user

Authenticated users can do everything unauthenticated users can. In addition authenticated users can do the following:

**Upload file** The user uploads a file to the server.

**Post message** The user posts a message related to some file on the server.

**Edit file** The user edits some of the metadata related to a file. It is possible to edit the file's name, description and tags. Other metadata or the file itself are not editable. The file must be originally uploaded by the user trying to edit it.

**Delete file** The user deletes a from the server. All messages related to the file are also deleted. The file must be originally uploaded by the user trying to delete it.

**Edit message** The user edits a message. The message must be originally posted by the user trying to edit it.

**Delete message** The user deletes a message from the server. The message must be originally posted by the user trying to delete it.

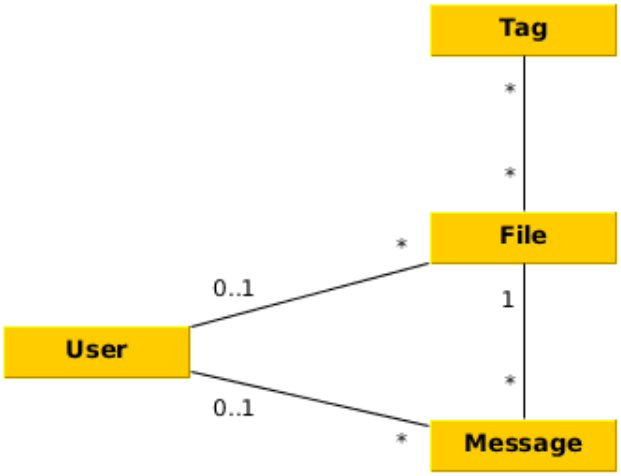
**Change password** The user changes the password for their account.

**Delete user account** The user deletes their account. This also deletes all their uploaded files and messages.

**Log out** The user logs out from the server and becomes and unauthenticated user.

# 3 Data content

## 3.1 Class diagram




## 3.2 Class attribute details

### 3.2.1 User

| Attribute | Type | Description |
|---|---|---|
| User name | String | The name of the user account on the server. |
| Password hash | String | One-way hash of the user's password. |
| Password salt | String | Additional data that is used in the hashing process to increase security. |
| Admin? | Boolean | Whether the user has admin privileges or not. Admins can freely edit and delete files and messages. |

#### 3.2.2 File

| Attribute | Type | Description |
|---|---|---|
| Uploader | User | The user who uploaded the file to the server. |
| File name | String | The name of the file |
| File description | String | Description and context of the file's contents. |
| Timestamp | Timestamp | The time the file was first uploaded to the server. |
| File path | String | Path to the file in the server file system. |
| File size | Integer | The size of the file in bytes. |
| File type | String | The MIME type of the file, for example text/plain. |

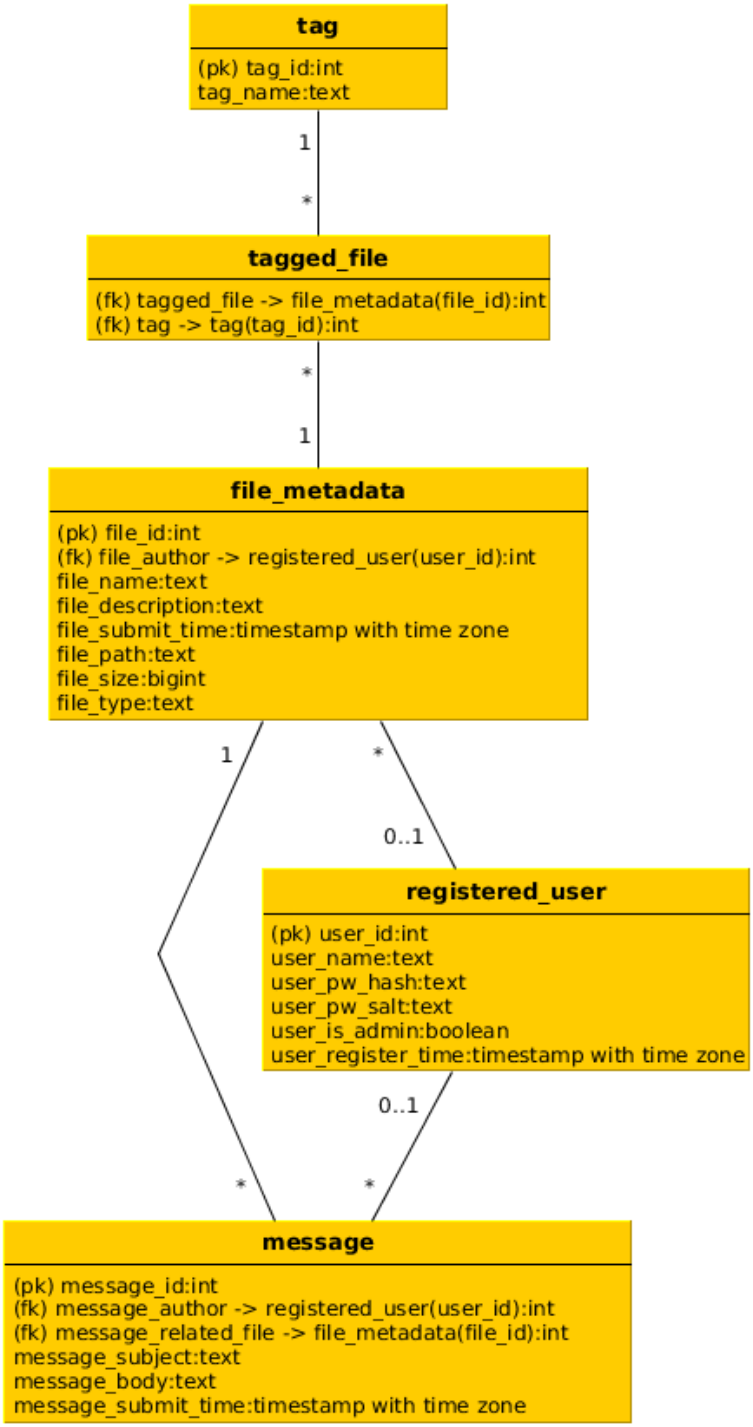
#### 3.2.3 Tag

| Attribute | Type | Description |
|---|---|---|
| Tag name | String | The name of the tag. |

#### 3.2.4 Message

| Attribute | Type | Description |
|---|---|---|
| Message poster | User | The user that posted the message. |
| Related file | File | The file that the message is related to. |
| Subject | String | The subject of the message. |
| Body | String | The content of the message. |
| Timestamp | Timestamp | The time the message was posted on the server. |

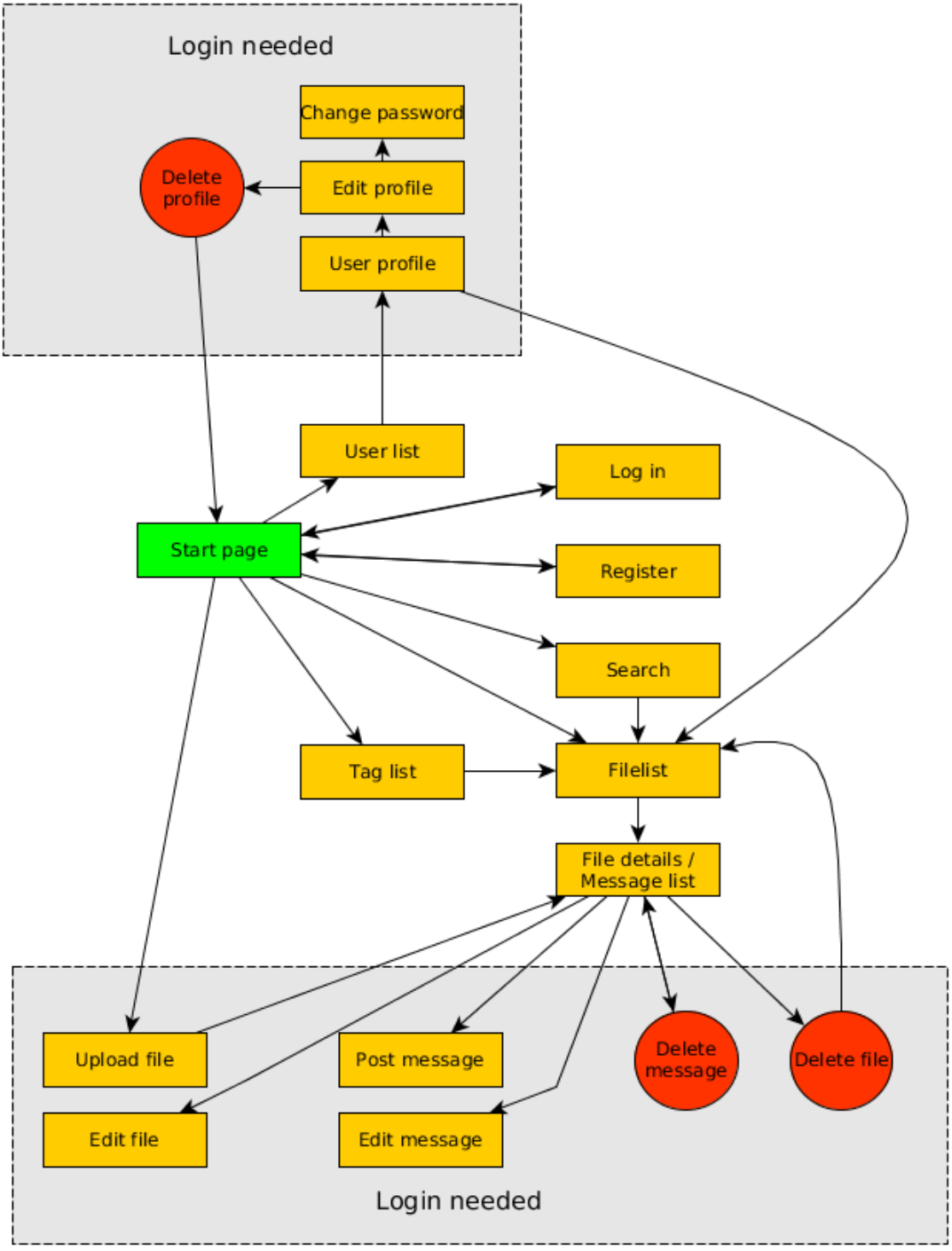
# 4 Database diagram



# 5 Application architecture

The application adheres to the MVC model. Source code is mostly located in the "app" subfolder, with further subfolders for models, views and controllers. Additional libraries are in the folders "lib" and "vendor" in the project root.

# 6 User interface and system components



# 7 Installation instructions

Clone the repository (https://github.com/SSTX/tsoha-fileboard) or otherwise obtain the files. Put them in a directory on your filesystem that your webserver serves to the internet. Change into that directory and run in the shell:

```
php composer.phar dump–autoload
```

# 8 Instructions for use

## 8.1 General

### 8.1.1 Accessing the application

The application is located here (http://ttiira.users.cs.helsinki.fi/tsoha)

### 8.1.2 Test account

This account has admin privileges.

**User name** 4423

**Password** coil

## 8.2 Uploading files

Uploaded files will be linked to your account.

### 8.2.1 Tags

Any tags entered when uploading a file will be linked to that file. Tags that didn't exist before will be created.

## 8.3 Editing files and messages

Editing functions can be accessed from the respective page of the file you want to edit. You must be logged in as the original uploader/poster to edit files/messages. Deleting a file also deletes any messages connected to it.

## 8.4 Managing your user account

User account controls can be accessed from your profile page. Currently supported functions are changing your password and deleting your account.

### 8.4.1 Deleting your user account

The deleting function removes your account from the database, along with any files and messages you have uploaded.

## 8.5 Searching files

Files can be searched based on four fields: file name, file type, tags and uploader name. Results must match ALL of the supplied search fields. Wildcards * match any string.