**Федеральное государственное бюджетное образовательное учреждение высшего образования «МГТУ «СТАНКИН»**

**Кафедра информатика и вычислительная техника**

**Отчёт**

по творческой работе

дисциплина

**«WEB-программирование»**

Выполнили: студенты группы ИДБ-16-02 Косырев Г.И

Федотов И.А.

Проверил: преподаватель Ефромеев Н.М.

**Москва 2019**

**Сроки выполнения**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Этап | Содержание этапа | Сроки и объем | Предварительные результаты | Отметка о выполнении | Дата/Подпись |
| 1 | Изучение теоретической части, разработка технического задания и проекта программы. |  | Полное описание работы, техническое задание. |  |  |
| 2 | Выбор структур данных, разработка интерфейса пользователя, разработка иерархии объектов, подготовка для тестирования. |  | Методы выполнения технического задания, описание структур данных, проектированный интерфейс. |  |
| 3 | Разработка алгоритмов и реализация на выбранном языке программирования. |  | Результаты в виде работающей программы. |  |
| 4 | Подготовка расчетно-пояснительной записки. |  | Полностью заполненная записка в печатном варианте. |  |
| 5 | Защита работы. |  | Полностью заполненная записка в печатном варианте. |  |

«\_\_» апреля 2019г.

Студенты группы ИДБ-16-02 Косырев Г.И\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Федотов И.А.\_\_\_\_\_\_\_\_\_\_\_\_\_

**Задание на работу по дисциплине «WEB-программирование»**

**Задание на творческую работу по дисциплине: “WEB-программирование”**  
Студентам 3 курса учебной группы ИДБ-16-02 Косыреву Глебу Игоревичу и Федотову.

**Тема работы**: “Онлайн-система коллективного рисования изображений и просмотра результата”.  
**Цель работы**: Разработка системы коллективного рисования изображений, в которой каждый зарегистрированный пользователь на 30 секунд может получить случайное изображение с возможностью его дорисовки, а также обладающей возможностью просмотра “Галереи” полученных изображений со списком авторов. В ходе работы студент должен изучить теоретические основы построения WEB-приложений и на практике продемонстрировать свои знания.  
**Функции приложения** (задачи, которые необходимо реализовать в данном проекте):  
Кейс 1: Просмотр главной страницы, содержащей кнопку старта  
Кейс 2: Подгрузка изображения при старте  
Кейс 3: Создание и подгрузка нового изображения в случае отсутствия в базе изображений либо “занятости” всех изображений  
Кейс 4: Обработка изображения пользователем  
Кейс 5: Автоматизированная отправка новой версии изображения по завершению 30-ти секундного временного интервала  
Кейс 6: Регистрация пользователя в системе  
Кейс 7: Логин пользователя  
Кейс 8: Просмотр основной страницы галереи с поиском   
Кейс 9: Просмотр страницы рисунка с его авторами  
Кейс 10: Добавление “запроса на удаление”  
Кейс 11: Удаление изображения (функция администрации)  
Кейс 12: Логаут  
Дополнительные функции, не заявленные изначально:  
Кейс 13: Вывод подсказки при поиске изображений с ожидаемым количеством результатов.  
Кейс 14: Установка “лайка” на изображение.  
Кейс 15: Выбор сортировки при поиске по изображений: либо по дате последнего обновления, либо по популярности, “лайкам”.  
Кейс 16: Просмотр страницы “своих работ” автором.  
Кейс 17: Предложение нового тэга.  
Кейс 18: Предложение изменения описания существующего тэга.  
Кейс 19: Добавление тэгов к изображению.  
Кейс 20: Добавление “запроса на смену тэгов”  
Кейс 21: Удаление тэгов. (функция администрации)  
Кейс 22: Работа с панелью администрации, принятие и отклонение предложений. (функция администрации)  
Кейс 23: Поиск по тэгам.  
Кейс 24: Просмотр страницы тэга.  
Кейс 25: Добавление комментария к изображению.  
Кейс 26: Возможность пометить комментарий как спам.  
Кейс 27: Удаление комментариев. (функция администрации)  
Кейс 28: Просмотр страницы лайкнутых изображений.  
Кейс 29: Валидация аккаунта через email.  
Кейс 30: Смена пароля через email.  
Кейс 31: Просмотр наиболее популярных в текущие сутки изображений и количества лайков за 24 часа к ним.

Сроки исполнения: 10 апреля 2019 года

**Техническое задание по работе**

Cистема коллективного рисования изображений

Необходимо разработать систему коллективного рисования изображений.  
Основанием для разработки служит желание разработчиков создать ресурс, позволяющий пользователям создавать совместные художественные произведения. Это позволит каждому пользователю находить единомышленников, самовыражаться, развивать свои способности и продвигать свои работы.  
Назначение разработки: создание художественных произведений, их поиск, классификация и оценка.  
Требования к программе: в программе должны быть использованы основные теоретические основы построения Web-приложения, полученные в ходе обучения по дисциплине Web-программирование.

**Методы выполнения технического задания**

Для выполнения технического задания мы будем пользоваться следующими инструментами:  
Для тестирования артефактов разработки нам потребуется установить на компьютер специальное программное обеспечение:  
1. Web-сервер Apache- программное обеспечение, отвечающее за отображения документов, запрашиваемых при наборе URL-адреса в Web-браузере.  
2. Интерпретатор PHP – для выполнения программ, написанных на языке PHP.  
3. MySQL - реляционная система управления базами данных.  
4. phpMyAdmin - веб-интерфейс для администрирования СУБД.

Всё указанное ПО можно получить бесплатно с сайтов производителей.  
Языки, используемые в проекте:  
1. HTML – язык разметки.  
2. CSS – язык стилей.  
3. JavaScript – язык, позволяющий сделать страницу динамической.  
4. PHP – язык, позволяющий взаимодействовать с БД, работать с cookies и файлами на сервере.  
5. SQL – язык, позволяющий создавать запросы к БД.  
Также в проекте использовались следующее артефакты:  
Bootstap - фреймворк для разработки пользовательского интерфейса.  
PHPMailer – библиотека с простой отправкой почты.

**Описание структур данных**

**База данных**

users

|  |  |  |
| --- | --- | --- |
| Название колонки | Тип данных | Применение |
| id | int(11) | Уникальный идентификатор пользователя. |
| login | varchar(25) | Уникальный логин. |
| email | varchar(360) | email пользователя. Используется для восстановления пароля. |
| passwordHash | varchar(60) | Хэш пароля. Хэш генерируется с помощью алгоритма CRYPT\_BLOWFISH. |

authors

|  |  |  |
| --- | --- | --- |
| Название колонки | Тип данных | Применение |
| id\_picture | int(11) | Идентификатор изображения, к которому приписывается художник. |
| id\_author | int(11) | Идентификатор художника, работавшего над id\_picture. Используется для вывода авторов на странице изображения в соответствующей графе. Также определяет может ли пользователь добавлять теги к изображению (если он автор этого изображения или модератор – может, иначе – нет). |
| number\_of\_change | int(3) | Порядковый номер вклада в изображение. Используется на странице изображения для определения z-index части изображения. |

likes

|  |  |  |
| --- | --- | --- |
| Название колонки | Тип данных | Применение |
| id\_picture | int(11) | Идентификатор лайкнутого изображения. Используется в поиске по популярности. |
| id\_user | int(11) | Идентификатор пользователя, поставившего лайк. Уникален для каждого id\_picture. |
| number\_for\_user | int(32) | Порядковый номер лайка, поставленного пользователем. Используется для сортировки по времени на странице лайкнутых изображений. |

sessions

|  |  |  |
| --- | --- | --- |
| Название колонки | Тип данных | Применение |
| user\_id | int(11) | Идентификатор пользователя, зашедшего на сайт. |
| tokenHash | varchar(65) | Хэш токена сессии. Токен записывается в куки пользователя, его хэш, полученный методом HMAC и алгоритмом sha256 хранится в базе данных. При произведении пользователем действий, требующих установления личности, токен хэшируется и ищется среди значений этой колонны. |
| submit\_date | timestamp | Время и дата входа. Используется для автоматического удаления сессий по истечению действия куки токена событием reset\_expired\_tokens. Значение по-умолчанию: CURRENT\_TIMESTAMP. |

pics

|  |  |  |
| --- | --- | --- |
| Название колонки | Тип данных | Применение |
| id | int(11) | Уникальный идентификатор изображения. |
| count\_of\_changes | int(3) | Хранит количество редакций изображения. Изображение считается завершённым, если количество изменений равно 6. |
| last\_update | timestamp | Хранит дату последнего действительного обновления изображения. По-умолчанию: 0. |
| filepath | varchar(60) | Хранит путь до изображения на сервере. |

drawing\_sessions

|  |  |  |
| --- | --- | --- |
| Название колонки | Тип данных | Применение |
| id\_user | int(11) | Идентификатор пользователя, производящего редактирование. |
| id\_pic | int(11) | Идентификатор изображения, которое редактируется. Может хранить -1, если отсутствует изображение, которое можно отредактировать. В этом случае после успешного редактирования в pics будет добавлена запись о редактируемом изображении с настоящим идентификатором. |
| date\_of\_start | timestamp | Время начала редактирования. Используется событием reset\_expired\_drawing\_sessions для автоматического прекращения незавершённых сессий рисования. |

moderators

|  |  |  |
| --- | --- | --- |
| Название колонки | Тип данных | Применение |
| moderator\_id | int(11) | Идентификатор модератора. Предоставляет доступ к admin\_panel, удалению и добавлению тегов без запроса на это. |

tags

|  |  |  |
| --- | --- | --- |
| Название колонки | Тип данных | Применение |
| id | int(11) | Уникальный идентификатор тэга. |
| tag | varchar(32) | Название тэга. |
| description | text | Описание тэга. |

tags\_of\_pics

|  |  |  |
| --- | --- | --- |
| Название колонки | Тип данных | Применение |
| id\_picture | int(11) | Идентификатор изображения, для которого установлен тэг. |
| id\_tag | int(11) | Идентификатор тэга, установленного изображению. |

suggestions

|  |  |  |
| --- | --- | --- |
| Название колонки | Тип данных | Применение |
| id | int(11) | Уникальный идентификатор предложения. |
| type | varchar(8) | Тип предложения. Доступные на данный момент типы: new tag – запрос на добавление нового тэга, edit tag – запрос на изменение описания тэга, tag chan – запрос на замену установленного авторами набора тэгов, delete – запрос на удаление изображения, spam – запрос на удаление спама в комментариях. |
| id\_user | int(11) | Идентификатор пользователя, от которого пришло предложение. |
| id\_pic | int(11) | Идентификатор изображения. Заполнено если тип delete или tag chan. |
| id\_comment | int(11) | Идентификатор комментария. Заполнено если тип spam. |
| id\_tag | int(11) | Идентификатор тэга. Заполнено если тип edit tag. |
| text\_arg\_32 | varchar(32) | Короткий текстовый аргумент. Заполнено если тип new tag (хранит название нового тэга). |
| text\_arg\_text | text | Длинный текстовый аргумент. Заполнено если тип new tag или edit tag (хранит описание тэга). |

comments

|  |  |  |
| --- | --- | --- |
| Название колонки | Тип данных | Применение |
| id | int(11) | Уникальный идентификатор комментария. |
| user\_id | int(11) | Идентификатор автора комментария. |
| pic\_id | int(11) | Идентификатор изображения, которое было прокомментировано. |
| comment\_text | text | Текст комментария. |
| time | timestamp | Дата и время отправки комментария |
| is\_anon | int(1) | Желает ли пользователь оставить комментарий анонимно. Тем не менее, user\_id всё равно будет хранить идентификатор этого пользователя, что поможет в модерации. |

not\_validated\_users

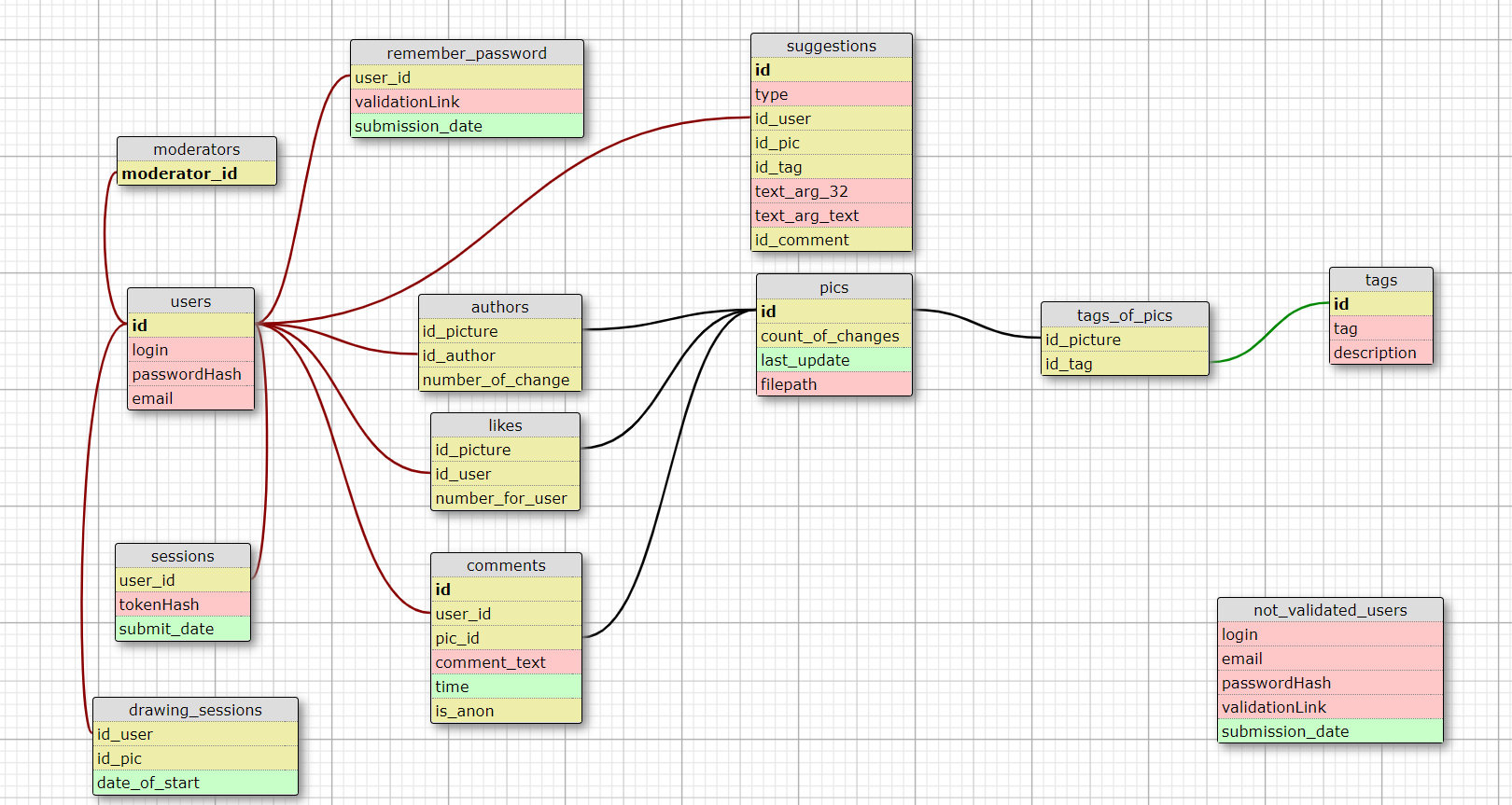
|  |  |  |
| --- | --- | --- |
| Название колонки | Тип данных | Применение |
| login | varchar(25) | Логин неподтверждённого валидацией по email пользователя |
| email | varchar(360) | email неподтверждённого валидацией по email пользователя |
| passwordHash | varchar(60) | Хэш пароля неподтверждённого валидацией по email пользователя |
| validationLink | varchar(32) | Значение, передаваемое как значение в query string при обращении к странице account\_validation.php для подтверждения email и пользователя в целом. Оно отправляется на почту пользователю. |
| submission\_date | timestamp | Время, используемое событием reset\_not\_validated\_users\_and\_password\_reset\_requests для удаления устаревших неподтверждённых пользователей. Срок подтверждения – около 24 часов. |

remember\_password

|  |  |  |
| --- | --- | --- |
| Название колонки | Тип данных | Применение |
| user\_id | int(11) | Идентификатор пользователя, запросившего смену пароля. |
| validationLink | varchar(32) | Значение, передаваемое как значение в query string при обращении к странице forgot\_password.php для подтверждения email и пользователя в целом. Оно отправляется на почту пользователю. |
| submission\_date | timestamp | Время, используемое событием reset\_not\_validated\_users\_and\_password\_reset\_requests для удаления устаревших ссылок на смену пароля. Срок подтверждения – около 24 часов. |

24\_hours\_likes

|  |  |  |
| --- | --- | --- |
| Название колонки | Тип данных | Применение |
| id\_pic | int(11) | Идентификатор изображения |
| count\_of\_likes | int(32) | Количество лайков, поставленных ему в текущие сутки. Содержимое таблицы удаляется каждый раз в 0:00 событием reset\_daily\_top. |

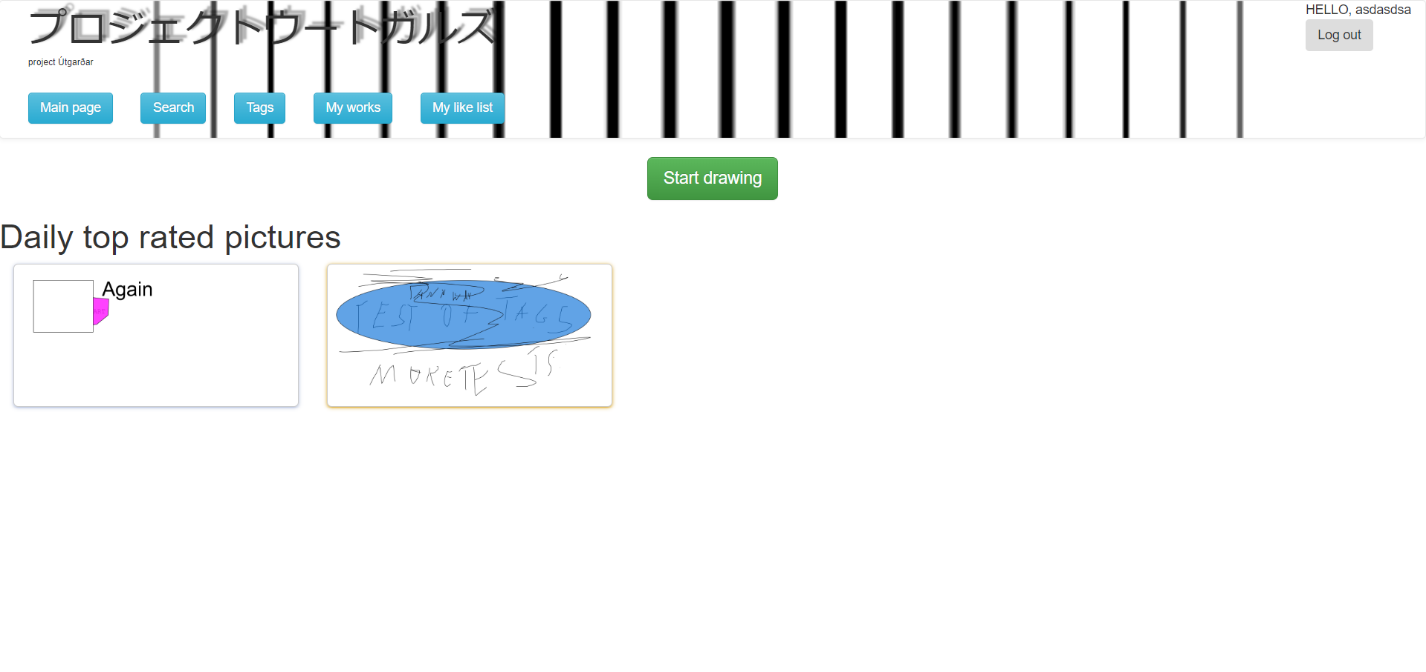


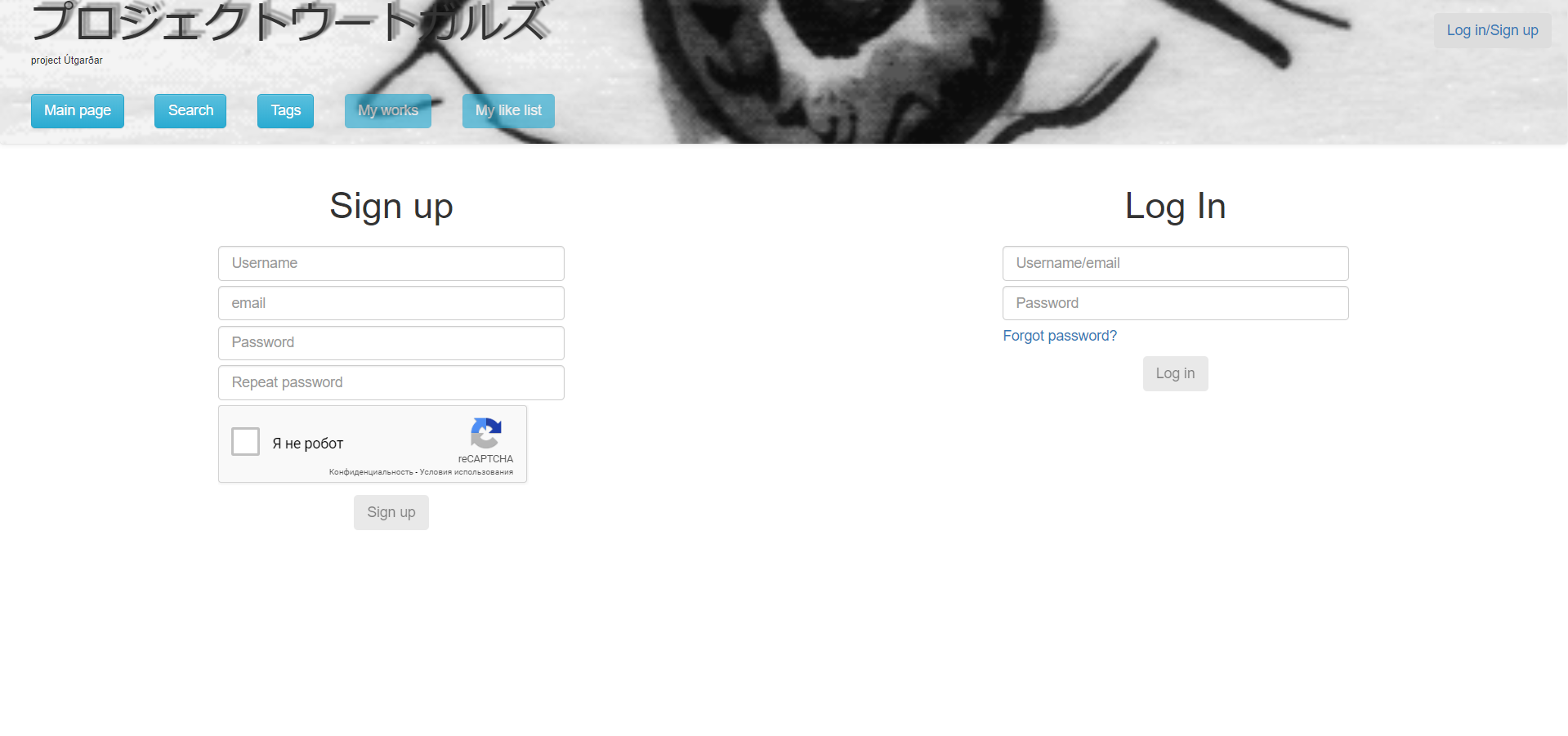
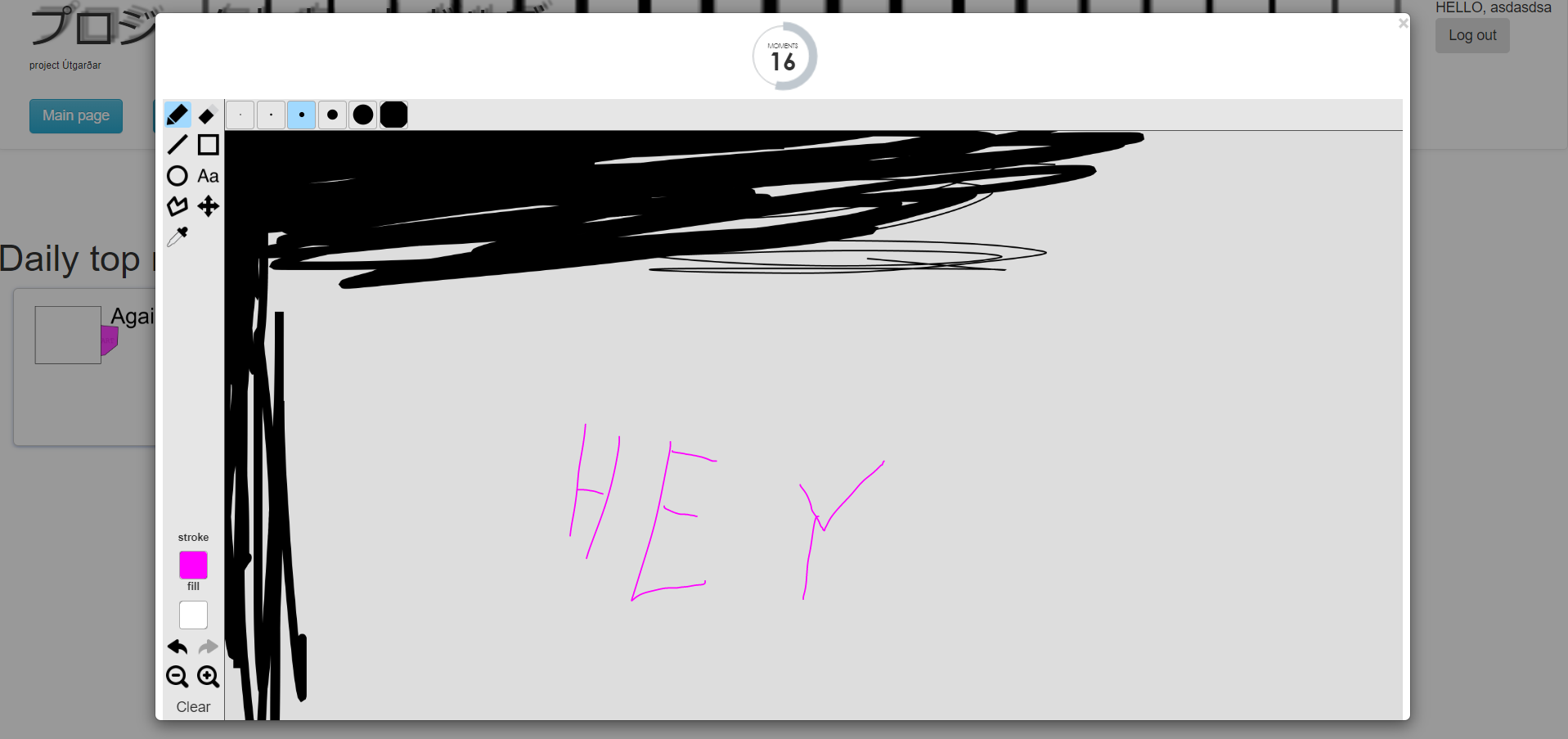
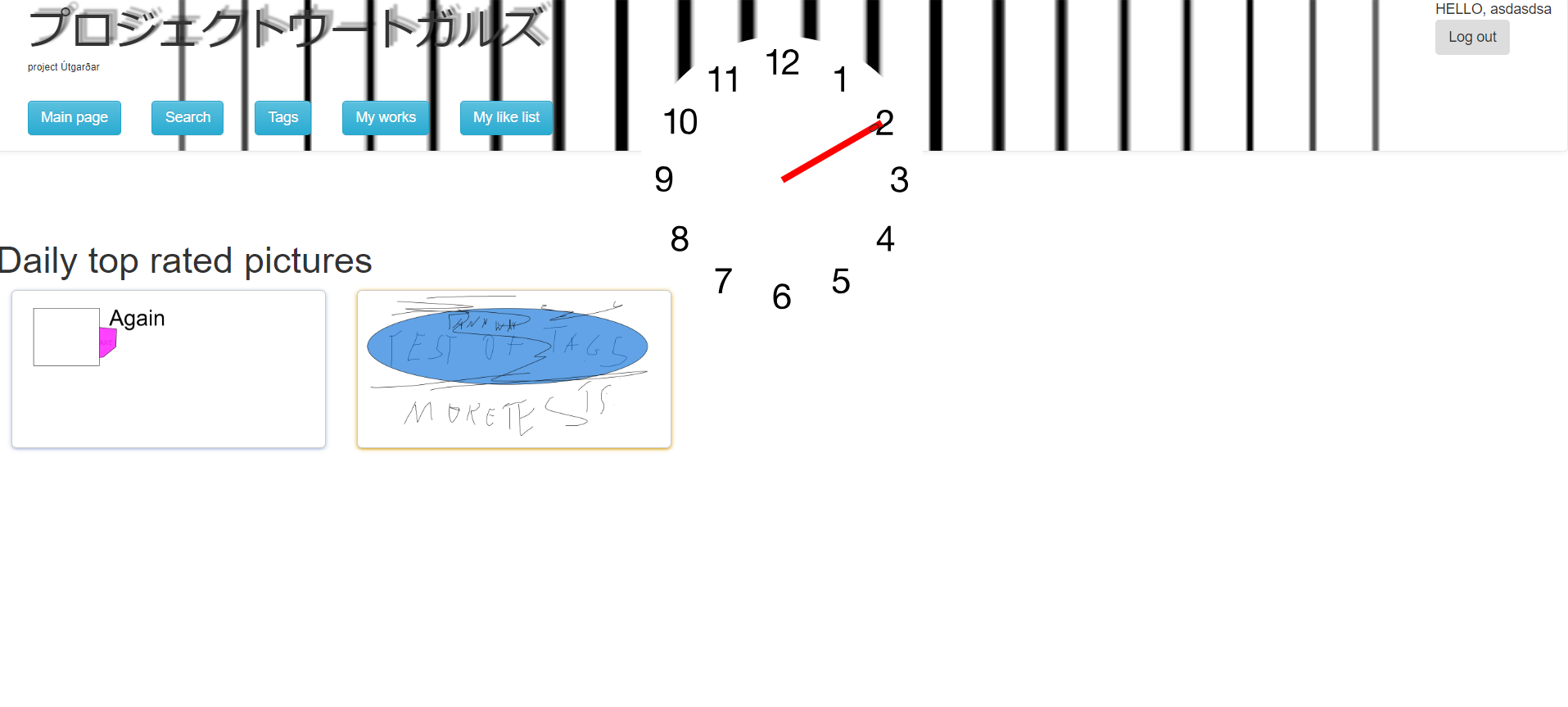
Также на сервере, в директории utgard.pro/db/img хранятся изображения, созданные пользователями. В img хранятся .png изображения общего результата рисования, названия этих файлов генерируются по шаблону [id изображения]\_[uniqid()]. Этот файл перезаписывается каждый раз, когда в изображение вносятся изменения. Также в img существуют поддиректории, именуемые как id изображения. Данные директории хранят части соответствующих изображений отдельно, то есть с каждым редактированием в такую директорию добавляется новое. Их названия генерируются по шаблону [id изображения]\_[id художника]\_[uniqid()].

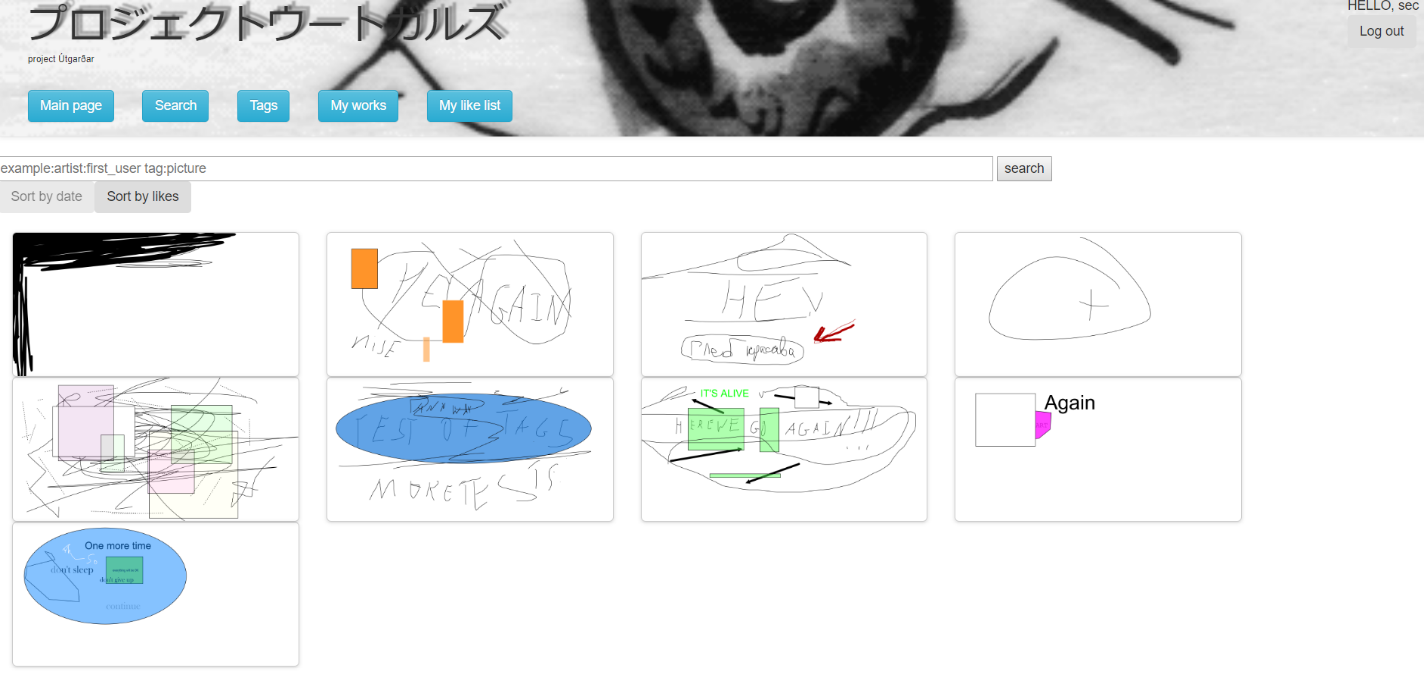
**Проектированный интерфейс**

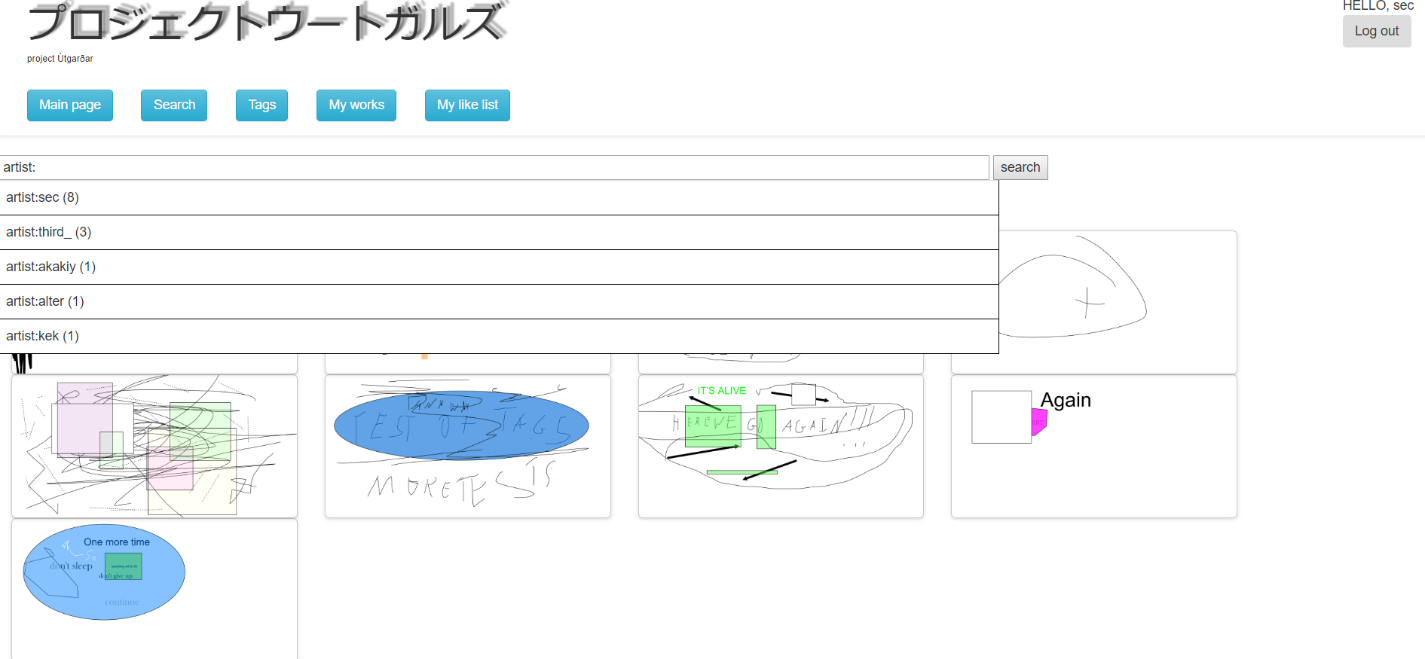
**Страницы, доступные всем пользователям:**index – главная страница  
img\_page – страница изображения  
login\_and\_registration – страница входа и регистрации  
search – страница поиска изображений с подсказками  
tag\_search – страница поиска тэгов  
tag\_page – страница с названием и описанием тэга  
**Страницы, доступные только авторизированным пользователям:**my\_works – страница с работами пользователя  
tag\_creation – страница предложения нового тэга  
my\_like\_list – страница просмотра лайкнутых изображений  
**Страницы, доступные только модераторам:**admin\_panel – страница принятия или отказа от предложений пользователей.  
**Страницы специального назначения:**  
account\_validation – используется для подтверждения пользователей по email  
forgot\_password – используется для смены пароля

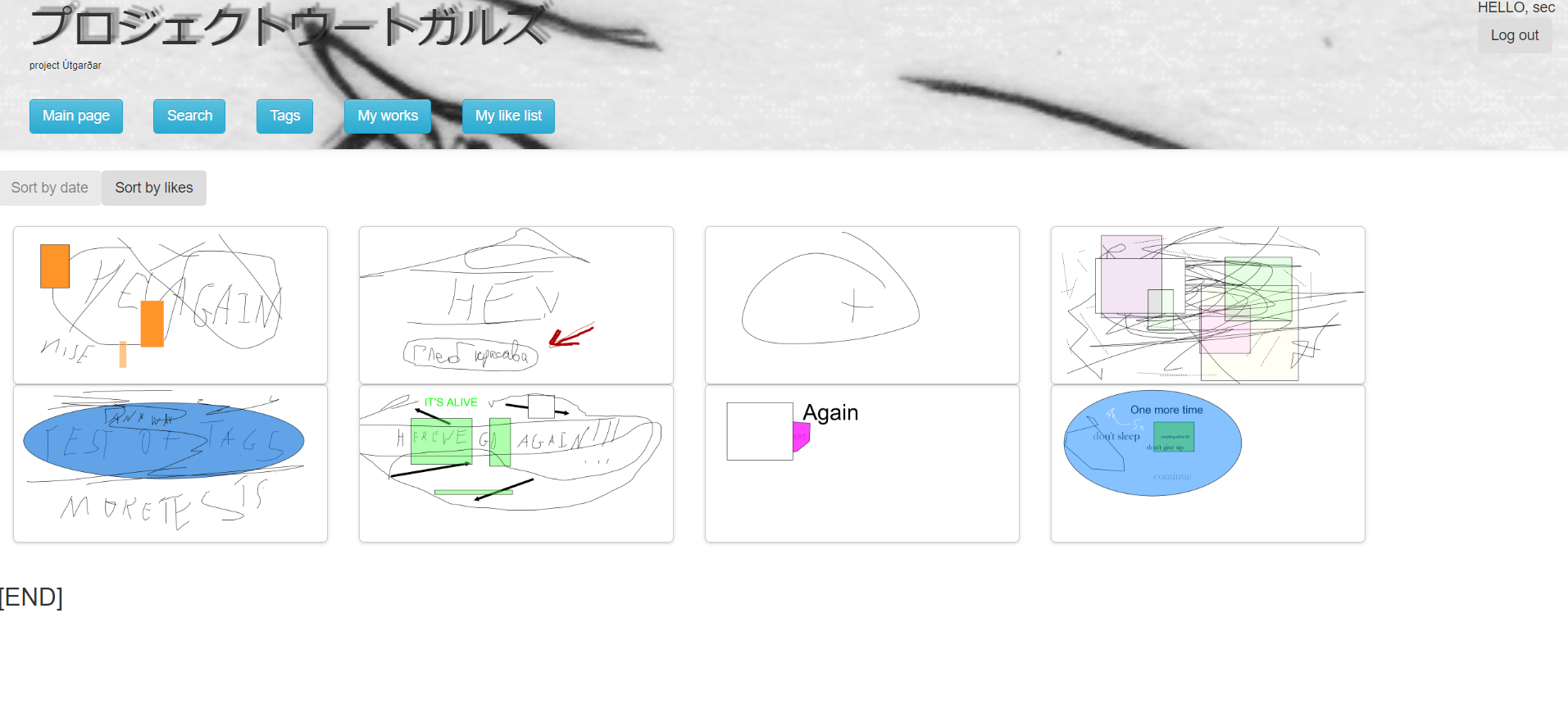
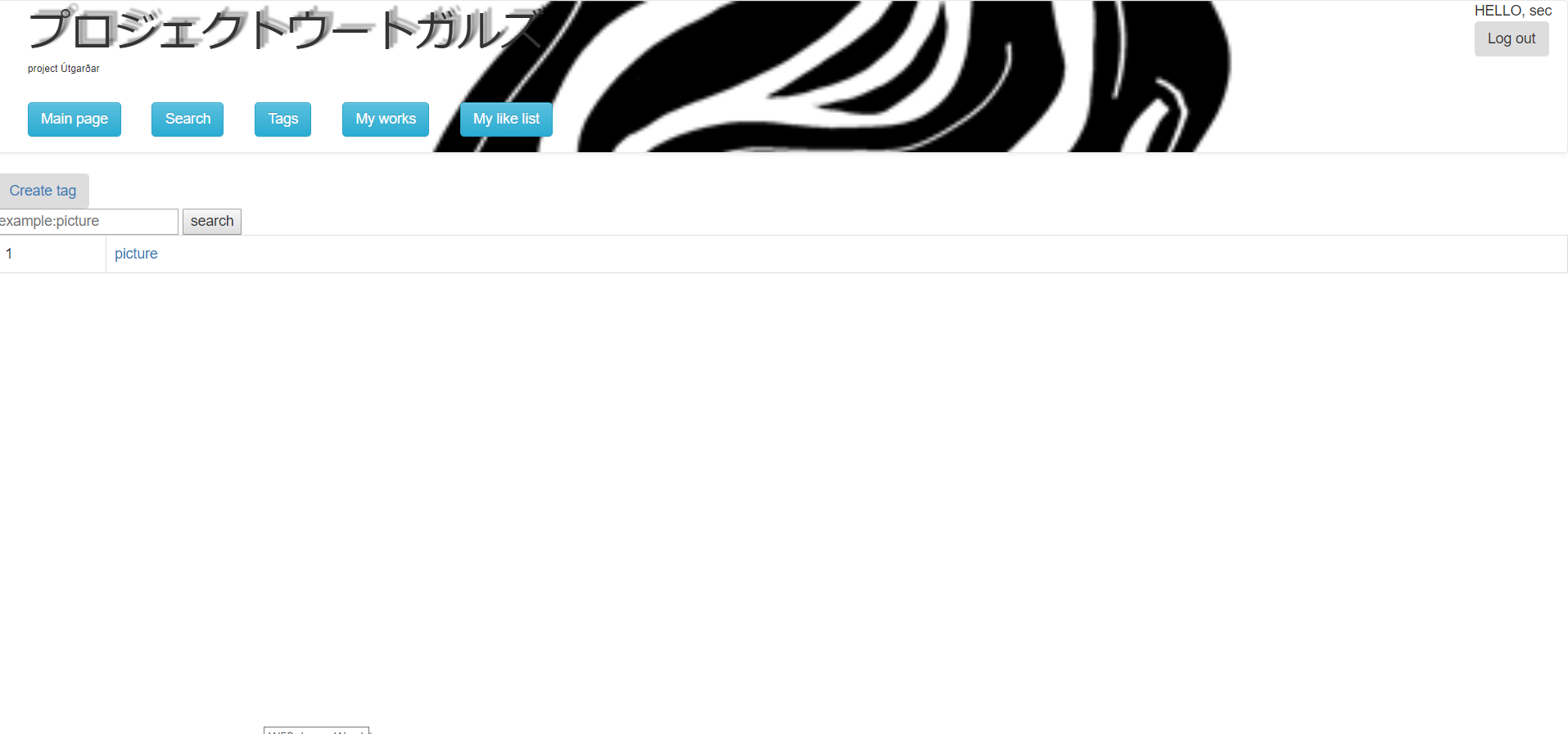
**Действия, требующие определённой роли:**Доступно всем авторизированным пользователям:  
Рисование, добавление флагов на удаление, смену тэгов изображения, установка флагов спама, предложение изменения описания тэга, предложение нового тэга, комментирование изображения.  
Доступно всем авторизированным пользователям, один раз для одного изображения:  
Установка лайка.  
  
Доступно художникам изображения и модераторам:  
Добавление тэга к изображению.  
  
Доступно модераторам:  
Удаление тэгов с изображения, удаление изображения по предложению в admin\_panel, добавление нового тэга в словарь тэгов по предложению в admin\_panel, изменение описания тэга по предложению в admin\_panel, просмотр предложений изменения тэгов в admin\_panel, удаление спама по предложению в admin\_panel, удаление комментариев на странице изображения.

**Приложение**

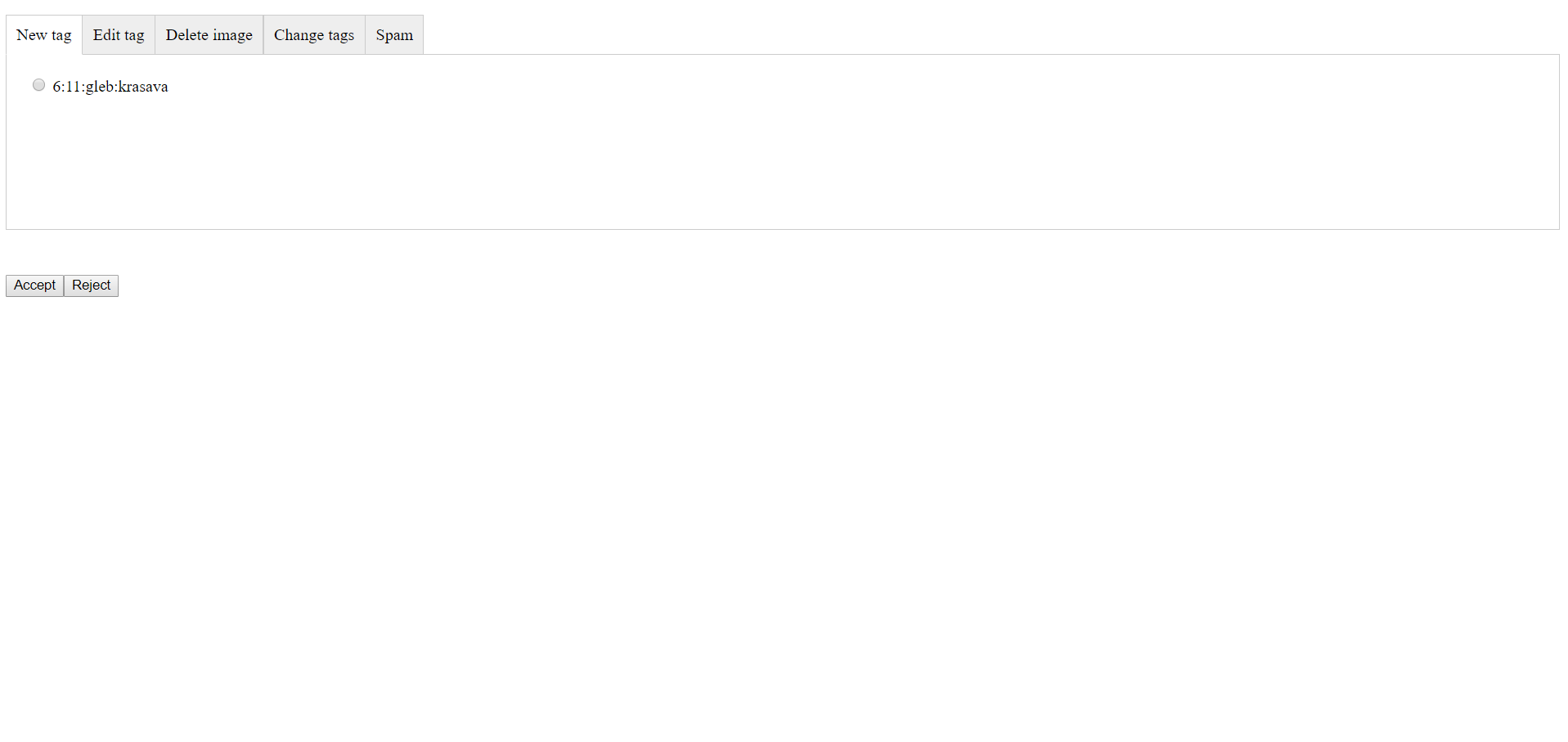
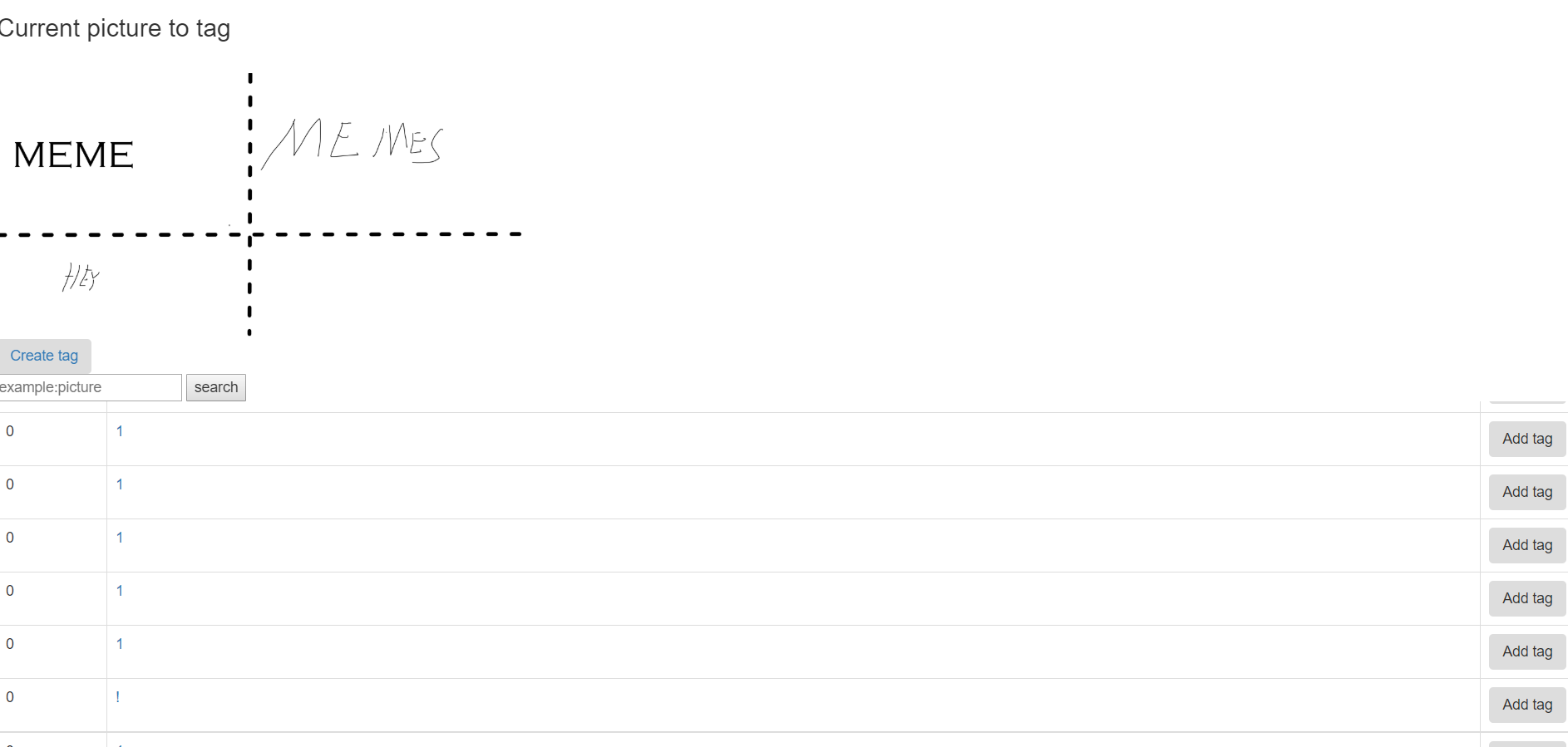
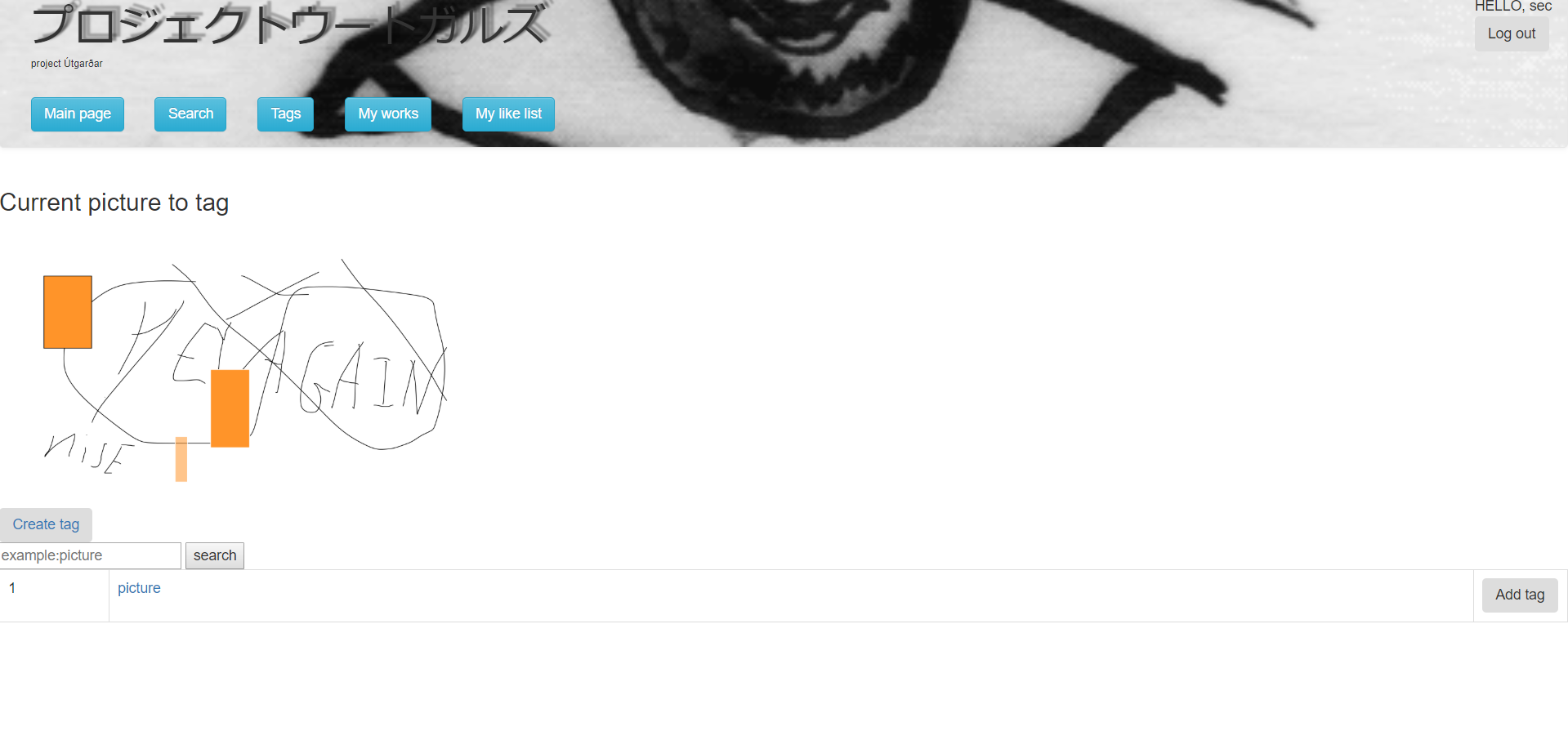


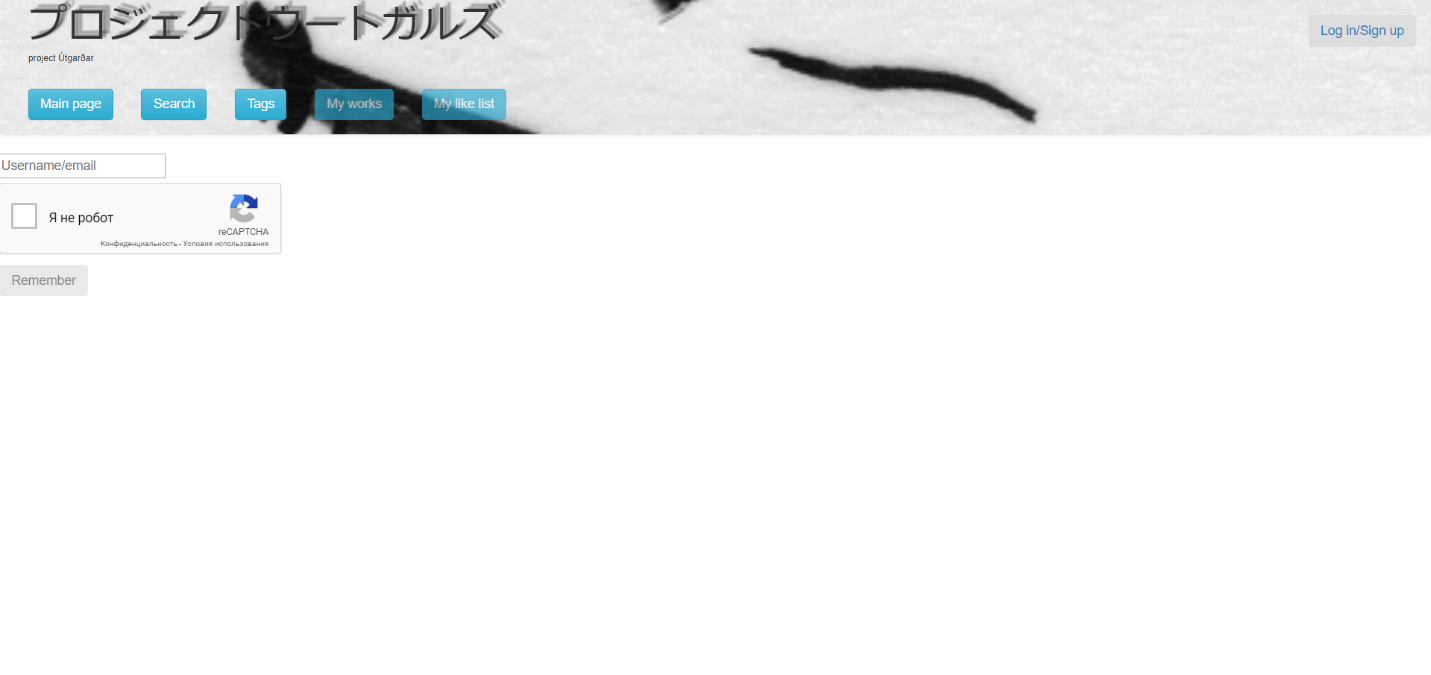
****

**---  
**



**(в поиске по тэгам и изображениям при прокрутке страницы ниже поисковой страницы последняя “прикрепляется” к верхушке страницы. При добавлении тэгов к изображениям соответствующее изображение также всё время держится на странице)**



****



**Содержание**

**Сроки выполнения………………………………….………………………………………………………………………………**

**Задание на разработку……………………………………………………………………………………………………………**

**Техническое задание на разработку………………………………………………………………………………………**

**Методы выполнения технического задания…………………………………….…………………………………**

**Описание структур данных………………………………………….…………………………………………………………**

**Проектированный интерфейс…………………..……………………………………………………………………………**

**Приложение……………………………………………………………………………………………………………………………**

**Содержание…………………………………………………….………………………………………………………………………**