Министерство науки и высшего образования Российской Федерации

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ

УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ,

МЕХАНИКИ И ОПТИКИ»

**Проект**

**Nonamechan**

Выполнил Щербаков Роман Игоревич

студент группы P3222

Преподаватель Ефимчик Евгений Александрович

Санкт-Петербург

2019

Содержание

[Введение 3](#_heading=h.30j0zll)

[Описание информационный системы 4](#_heading=h.1fob9te)

[Краткое описание системы 4](#_heading=h.3znysh7)

[Используемые технологии 4](#_heading=h.2et92p0)

[Анализ технических требований 4](#_heading=h.tyjcwt)

[Обзор аналогов 5](#_heading=h.3dy6vkm)

[Pinterest 5](#_heading=h.1t3h5sf)

[Tumblr 5](#_heading=h.4d34og8)

[Анализ требований, прецеденты использования 6](#_heading=h.2s8eyo1)

[Действующие лица: 6](#_heading=h.17dp8vu)

[Прецеденты использования: 6](#_heading=h.3rdcrjn)

[Устройство базы данных 11](#_heading=h.26in1rg)

[Доступ к платформе 16](#_heading=h.lnxbz9)

[Взаимодействие пользователя с системой 17](#_heading=h.35nkun2)

[Заключение 22](#_heading=h.44sinio)

[Список использованных источников 23](#_heading=h.2jxsxqh)

# Введение

Цель данного проекта – приобретение навыков самостоятельного проектирования информационных систем, оптимальное использование их в создании web-сервиса и поддержании бесперебойной работы при обслуживании неограниченного числа пользователей.

# Описание информационный системы

## Краткое описание системы

Проект nonamechan – это платформа для размещения изображений с возможностью их комментирования, сортировки по ключевым словам и добавлением в избранное. Помимо загрузки изображений пользователи могут подписываться на авторов, если им нравится загружаемый ими контент или авторы создают нечто уникальное. Проект не запрашивает cookie-файлы пользователей для обеспечения приватности во время пребывания на платформе nonamechan.

## Используемые технологии

Проект создан с помощью свободно распространяемых библиотек для языка программирования Python версии 3.7 Flask, Flask-SQLAlchemy, Bcrypt; дизайн страниц оформлен с помощью HTML и CSS, для размещения элементов на страницах используется шаблонный движок Jinja2.

## Анализ технических требований

Веб-сервис должен быть доступен всем пользователям сети интернет, отвечать на действия посетителей без существенной задержки(не более двух секунд) и обеспечивать сохранность данных всех пользователей даже при сбоях в работе системы. Несанкционированный доступ(например, вставка javascript кода в формы) предотвращен с помощью использования CSRF(Cross Site Request Forgery) защиты.

# Обзор аналогов

## Pinterest

Популярная сеть с картинками-ссылками на первоисточник. Можно добавлять свои картинки, прикреплять их в свой профиль, искать по тегам, но их не видно в самом посте. Реализована система подбора контента по недавно просмотренному.

## Tumblr

Тот же функционал что и у пинтереста, только теги отображаются под постом и картинки существуют на самом сайте вместо ссылок на другие источники. Отличается интересным дизайном и магией скриншотов комментариев которые заполняют весь интернет своими глубокими мыслями.

Критерии оценивания(таблица 1):

* Красивая страница пользователя
* Удобство поиска контента
* Шпионаж(рекомендуемый контент, контекстная реклама и т.д.)

Таблица 1 - Критерии оценивания аналогов

|  |  |  |
| --- | --- | --- |
|  | Pinterest | Tumblr |
| Красивая страница пользователя | Нет | Да |
| Удобство поиска контента | Да | Да |
| Шпионаж | Да | да |

# Анализ требований, прецеденты использования

## Действующие лица:

* Гость – неавторизованный пользователь, может смотреть картинки и профили людей, может зарегистрироваться и стать участником
* Участник – авторизованный пользователь, может оставлять комментарии, загружать картинки, редактировать и удалять свои картинки, добавлять в избранное, подписываться на других участников.
* Заблокированный участник – не может оставлять комментарии и загружать картинки
* Модератор – привилегированный участник, может редактировать чужие посты
* Создатель – в дополнение ко всему вышесказанному может назначать модераторов и банить пользователей

## Прецеденты использования:

На рисунке 1 представлена диаграмма прецедентов использования

A close up of a map

Description automatically generated

*Рисунок 1 - Прецеденты использования*

Действующие лица и их возможности. Прецеденты подробно описаны в таблицах 2-11.

1. Гость
   1. Регистрация
   2. Вход в аккаунт (include a)
   3. Просмотр картинок, профилей, фильтр по тегам
2. Участник
   1. Изменение информации своего профиля
   2. Восстановление пароля при входе в аккаунт
   3. Загрузка картинок и последующее редактирование
   4. Комментирование картинок (extend e)
   5. Просмотр картинок, профилей, фильтр по тегам (extend d)
   6. Добавление картинок в избранное, отобразится в отдельной колонке в профиле (include e)
3. Заблокированный участник
   1. Изменение информации своего профиля
   2. Восстановление пароля
   3. Просмотр картинок, профилей, фильтр по тегам
   4. Добавление картинок в избранное (include e)
4. Модератор
   1. Изменение информации своего профиля
   2. Восстановления пароля при входу в аккаунт
   3. Загрузка картинок и последующее редактирование
   4. Комментирование картинок (extend g)
   5. Редактирование и удаление чужих картинок(extend g)
   6. Добавление картинок в избранное (include g)
   7. Просмотр картинок, профилей, фильтр по тегам
5. Создатель
   1. Изменение информации своего профиля
   2. Восстановления пароля при входу в аккаунт
   3. Загрузка картинок и последующее редактирование
   4. Комментирование картинок (extend g)
   5. Редактирование и удаление чужих картинок(include g)
   6. Добавление картинок в избранное (include g)
   7. Просмотр картинок, профилей, фильтр по тегам
   8. Повышение участников до модераторов и наоборот (include+extend g)
   9. Изменение статуса участников заблокированный/обычный (include+extend g)

Таблица 2 – Прецедент использования регистрация

|  |  |
| --- | --- |
| Прецедент использования | Регистрация |
| Краткое описание | Позволяет создать учетную запись |
| Действующие лица | Гость |
| Предусловия | Гость хочет создать аккаунт |
| Основной поток | Гость заполняет форму регистрации состоящую из почты, имени аккаунта и пароля, нажимает на кнопку регистрации и перенаправляется на страницу входа в аккаунт |
| Альтернативные потоки | Неверно введенная почта, неверно повторенный пароль не позволяют создать аккаунт, пользователь остается на странице регистрации и получает соответствующие сообщения об ошибке.  Если почта/никнейм уже зарегистрированы пользователю предлагается перейти на страницу входа |
| Постусловия | После основного потока в базе данных создается новая учетная запись с почтой, никнеймом, хешированным паролем и незаблокированным статусом |

Таблица 3 – Прецедент использования вход(log in)

|  |  |
| --- | --- |
| Прецедент использования | Вход в аккаунт |
| Краткое описание | Позволяет войти в заранее созданную учетную запись |
| Действующие лица | Гость |
| Предусловия | Гость хочет зайти в аккаунт |
| Основной поток | Гость вводит почту и пароль. После нажатия кнопки войти гость отправляется на главную страницу со свежими картинками. |
| Альтернативные потоки | Неверно введенный пароль/почта выводит сообщение об ошибке. Если почты нет в базе появляется предложение регистрации |
| Постусловия | Гость авторизуется в сессии и получает привилегия своей учетной записи |

Таблица 4 – Прецедент использования просмотр картинки

|  |  |
| --- | --- |
| Прецедент использования | Просмотр картинок, профилей, фильтр по тегам |
| Краткое описание | Позволяет просмотреть подробную информацию о пользователе или картинке |
| Действующие лица | Все пользователи |
| Предусловия | Пользователь нажимает на ссылку/вводит ее вручную, вводит желаемые теги в окно в шапке сайта |
| Основной поток | Пользователь переходит на страницу с картинкой/профилем и всей сопутствующей информацией. При фильтре выводится интерфейс домашней страницы с указанием фильтра. Появляются картинки только с указанными тегами. |
| Альтернативные потоки | Профиля/картинки не существует по ссылке – выводится ошибка 404. Пост по ссылке существует, но картинки нет в базе. Тогда вместо картинки выводится стандартный знак отсутствия мультимедиа. Нет картинок по запросу – выводится пустая домашняя страница. |
| Постусловия | Пользователь продолжает лазить по сайту |

Таблица 5 – Прецедент использования изменение профиля

|  |  |
| --- | --- |
| Прецедент использования | Изменение информации аккаунта |
| Краткое описание | Позволяет изменить аватарку, никнейм, пароль |
| Действующие лица | Все зарегистрированные пользователи |
| Предусловия | Пользователь переходит на страницу своего аккаунта |
| Основной поток | Пользователь вводит новые значения в предложенные поля или нажимает на кнопку «изменить аватар», загружает картинку |
| Альтернативные потоки | Картинка слишком большая. Выводится сообщение о превышении максимального размера файла |
| Постусловия | В базе данных обновляются все измененыые поля |

Таблица 6 - Прецедент использования загрузка картинки

|  |  |
| --- | --- |
| Прецедент использования | Загрузка картинок |
| Краткое описание | Пользователь загружает картинку с тегами и она появляется на сайте |
| Действующие лица | Незаблокированные пользователи |
| Предусловия | Пользователь нажимает на кнопку «загрузить» в шапке сайта |
| Основной поток | Пользователь загружает картинку и вводит теги. Потом попадает на страницу с постом |
| Альтернативные потоки | Картинка слишком большая. Выводится сообщение о превышении максимального размера файла. Теги введены не по правилам – выводится ошибка. |
| Постусловия | В базе данных появляется новая запись о посте, формируются связь многие-ко-многим между постами и тегами |

Таблица 7 - Прецедент использования оставление комментария

|  |  |
| --- | --- |
| Прецедент использования | Комментирование |
| Краткое описание | Пользователь оставляет комментарий под картинкой |
| Действующие лица | Незаблокированные пользователи |
| Предусловия | Пользователь хочет поделиться впечатлением |
| Основной поток | Пользователь пишет комментарий в окне и нажимает кнопку «submit» |
| Альтернативные потоки | Текстовое поле комментария пустое – выводится сообщение об ошибке. Пользователь заблокирован – возможности оставить комментарий нет вообще. Пользователь не авторизован – появляется надпись «You need to log in to leave comments» |
| Постусловия | В базу данных в таблицу комментариев добавляется текст комментария, в таблицу поста в список связей один-ко-многим добавляется новый комментарий. |

Таблица 8 - Прецедент использования редактирование поста(только для модератора)

|  |  |
| --- | --- |
| Прецедент использования | Редактирование(своих/чужих) картинок |
| Краткое описание | Можно изменить набор тегов и картинку поста |
| Действующие лица | Зарегистрированные пользователи, модератор/создатель(для чужих) |
| Предусловия | Пользователь нажимает ссылку Edit под картинкой |
| Основной поток | Пользователь меняет желаемые поля, нетронутые поля остаются неизменными, нажимает Submit, переносится на страницу с обновленным постом |
| Альтернативные потоки | Такие же как при загрузке |
| Постусловия | В базе данных обновляются все обновленные поля, если изменилась картинка, старая удаляется |
| Прецедент использования | Удаление(своих/чужих) картинок |
| Краткое описание | Удаление картинки с сайта |
| Действующие лица | Зарегистрированные пользователи, модератор/создатель(для чужих) |
| Предусловия | Пользователь нажимает ссылку Delete под картинкой |
| Основной поток | Пользователь попадает на страницу с подтверждением удаления, подтверждает удаление и возвращается на домашнюю страницу. |
| Альтернативные потоки | Пользователь отказывается удалять пост на странице подтверждения и возвращается на страницу с картинкой. |
| Постусловия | Из базы данных удаляется пост и все связи с тегами |

Таблица 9 – Прецедент использования добавление картинки в избранное

|  |  |
| --- | --- |
| Прецедент использования | Добавление картинок в избранное |
| Краткое описание | Пользователь сохраняет картинку к себе в профиль и может со страницу аккаунта посмотреть все сохраненные. Также может скрыть эту секцию от других пользователей |
| Действующие лица | Зарегистрированные пользователи |
| Предусловия | Пользователь хочет сохранить понравившуюся картинку себе в профиль чтобы не потерять |
| Основной поток | Пользователь нажимает на Save to favorites под картинкой, потом может увидеть ее на своей странице |
| Альтернативные потоки | Неавторизованные пользователи не видят кнопки. Если картинка уже добавлена в избранное, вместо кнопки Save to favorites появляется Remove from favorites |
| Постусловия | В таблицу пользователя добавляется новый элемент в связь один-ко-многим профиль-ид\_постов |

Таблица 10 - Прецедент использования изменение роли пользователя(для админа)

|  |  |
| --- | --- |
| Прецедент использования | Изменение роли пользователя |
| Краткое описание | Создатель дает права модератора участнику или забирает их |
| Действующие лица | Создатель, участник |
| Предусловия | Создатель хочет дать привилегии участнику |
| Основной поток | На странице пофиля кандидата появляется кнопка Promote\Disapprove в зависимости от текущего статуса участника. При нажатии роль меняется +1\-1 |
| Альтернативные потоки | Нет |
| Постусловия | В таблице пользователей в записи соответствующего кандидата меняется значение поля role |

Таблица 11 - Прецедент использования бан/разбан пользователя(для админа)

|  |  |
| --- | --- |
| Прецедент использования | Изменение статуса пользователя |
| Краткое описание | Создатель блокирует или возвращает права участнику |
| Действующие лица | Создатель, участник |
| Предусловия | Создатель хочет забанить\простить пользователя |
| Основной поток | На странице пофиля кандидата появляется кнопка Ban\Unban в зависимости от текущего статуса участника. При нажатии роль меняется |
| Альтернативные потоки | Нет |
| Постусловия | В таблице пользователей в записи соответствующего кандидата меняется значение поля status и пользователь попадает под ограничения в зависимости от статуса |

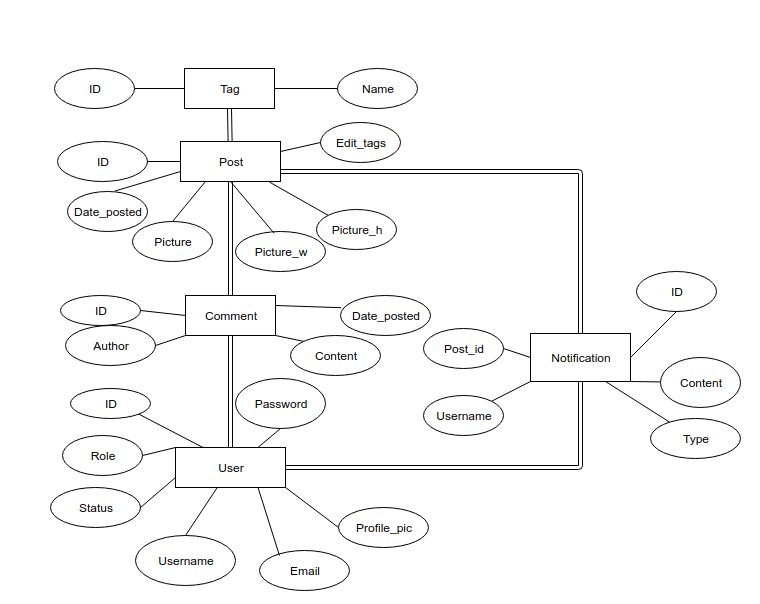
# Устройство базы данных

В базе данных взаимодействуют 4 сущности: пользователь, пост, комментарий, оповещение. Они соединены посредством отношений(relationship), позволяющими устанавливать связь между записями в разных таблицах. В базе используются как многие-ко-многим, так и один-ко многим отношения. Для построения связи многие-ко-многим сбыли созданы специальные ассоциативные таблицы, хранящие записи id-id, устанавливающие соответствие между идентификаторами записей разных таблиц. Схема взаимодействия приведена на рисунке 3, ER диаграмма на рисунке 2, табличное описание в таблице 12.

Основная сложность в оперировании такими связями состоит в удалении связанных элементов. Часто применяется каскадное удаление, которое может внезапно нарушить целостность данных. Если удалить один из связанных элементов, его отношение либо перейдет на другой – идентификатор останется в ассоциативной таблице, но на его месте окажется другая запись, либо связь останется пустой, и при обращении выдаст элемент None, с которым ни одна операция SQLAlchemy не работает. Поэтому такое действие запрещено. При удалении элемента нужно сначала удалить все связи, а потом и сам элемент из таблицы, поэтому операция удаления самая трудозатратная.

Таблица 12 – Сущности базы данных и их свойства

|  |  |  |
| --- | --- | --- |
| **Сущность** | **Свойство** | **Комментарии** |
| User(пользователь) | ID | Идентификатор пользователя. Ключевое свойство |
| Role | Роль пользователя определяет привилегии. Обычный\модератор\создатель |
| Status | Определеяет ограничения пользователя. Активен\заблокирован |
| Username | Имя пользователя на сайте |
| Email | Почта привязанная к аккаунту. Нужна для восстановления пароля |
| Profile\_pic | Аватар пользователя. Отображается на странице профиля. Хранит имя файла картинки. |
| Password | В базе данных хранится хешированный пароль |
| Posts | Хранит отношение один-ко-многим с постами |
| Fav\_list | Отношение многие-ко-многим к постам через atable\_fav. Используетяс для отображения избранных картинок в профиле |
| Comment\_list | Отношение один-ко-многим комментариям. Присутствует только у забаненных пользователей |
| Post(пост) | ID | Идентификатор поста. Ключевое свойство |
| Date\_posted | GMT время загрузки поста |
| Picture | Имя файла картинки поста |
| Picture\_w | Ширина картинки |
| Picture\_h | Высота картинки поста |
| User\_id | Идентификатор автора поста |
| Edit\_tags | Текстовое поле с именами привязанных тегов |
| Tag\_list | Отношение многие-ко-многим с тегами через atable\_tag |
| Comment\_list | Отношение один-ко-многим с комментариями поста |
| Likers\_list | Отношение многие-ко-многим с пользователями через atable\_fav |
| Tag(тег) | Id | Идентификатор тега. Ключевое свойство |
| Name | Название тега |
| Post\_list | Отношение многие-ко-многим для доступа к постам имеющим данный тег |
| Atable\_tag(ассоциативная таблица) | Post\_id | Идентификатор поста в отношении |
| Tag\_id | Идентификатор тега в отношении |
| Tag | Отношение один-ко-многим по которому из таблицы поста обращаются к atable и получают все tag\_id |
| Post | Отношение один-ко-многим по которому из таблицы тега обращаются к atable и получают все post\_id |
| Comment(коммент) | ID | Идентификатор коммента. Ключевое свойство |
| Author | Никнейм автора комментария |
| Date\_posted | GMT время оставления коммента |
| Content | Текстовое поле с содержанием коммента |
| Post\_id | Идентификатор поста под которым оставлен коммент |
| User\_id | Идентификатор пользователя под которым оставлен коммент. Указывается только для забаненных, у остальных пользователей нет комментов |
| Under\_post | Отношение многие-к-одному комментариев к посту |
| Under\_user | Так же как и under\_post |
| Notification(уведомление) | Id | Идентификатор уведомления. Ключевое свойство |
| Post\_id | Ид поста к которому относится уведомление. Используется для создания ссылки |
| Username | Имя забаненного пользователя. Используется просто для удобного отображения |
| Type | Тип уведомления. Бан или новый пост. Когда новый пост то заполнены обя поля ид. С баном только поле username |
| Content | Текст уведомления. Вариантов два, как и типов. |
| Atable\_fav | Id | Ключевое свойство |
| User\_id | Ид пользователя добавившего пост в избранное |
| Fav\_id | Ид картинки добавленной user\_id в избранное |
| Atable\_subs | Cmaker\_id | Ид контентмейкера на которого подписваются люди |
| Sub\_id | Ид пользователя подписанного на cmaker\_id |
| Atable\_notif | Notif\_id | Ид уведомления из таблицы notifications |
| Recip\_id | Ид пользователя для которого предназначено уведомление notif\_id |



*Рисунок 2. ER-диаграмма базы данных*

A close up of a map

Description automatically generated

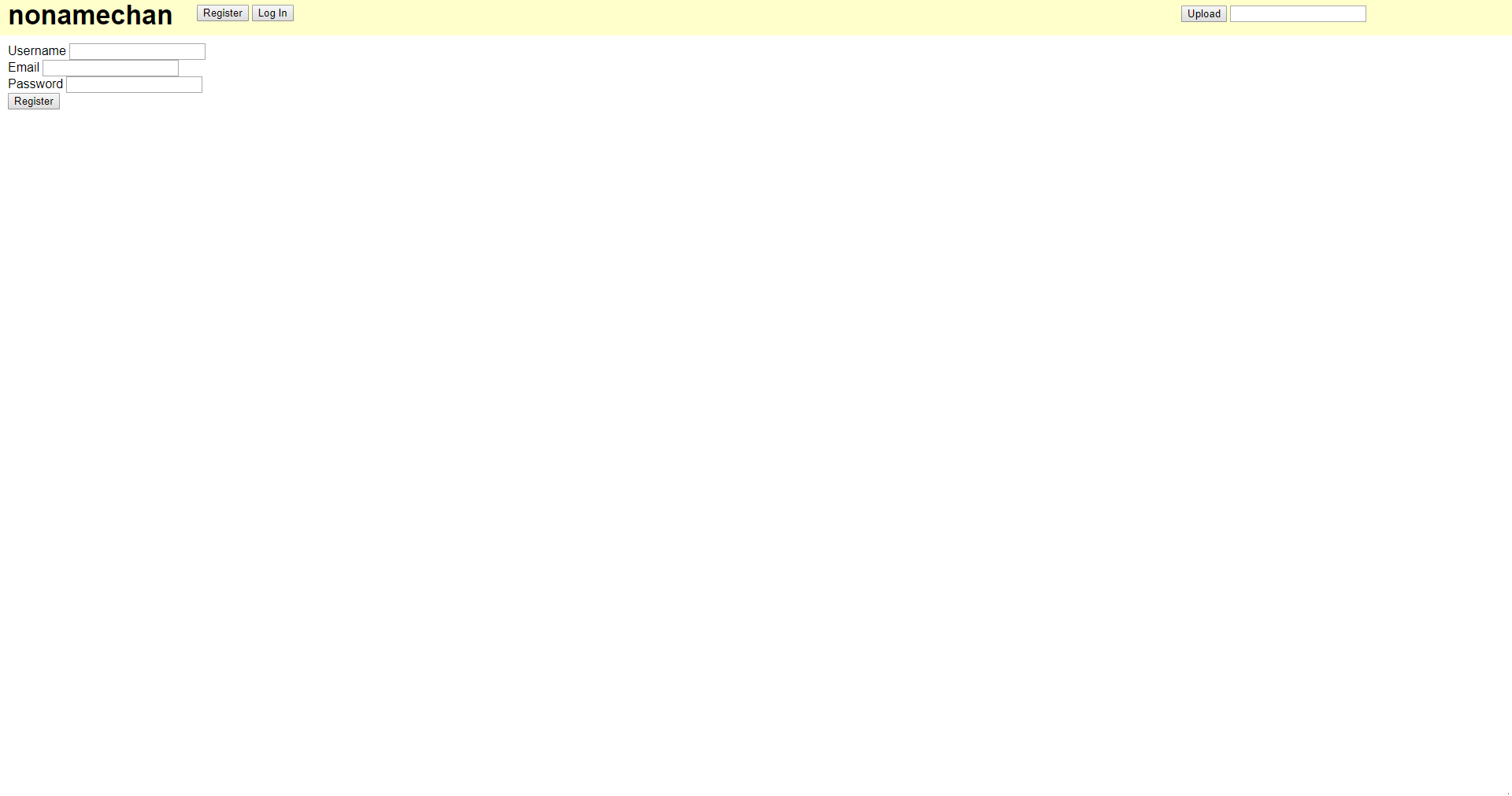
*Рисунок 3. Схема базы данных*

# Доступ к платформе

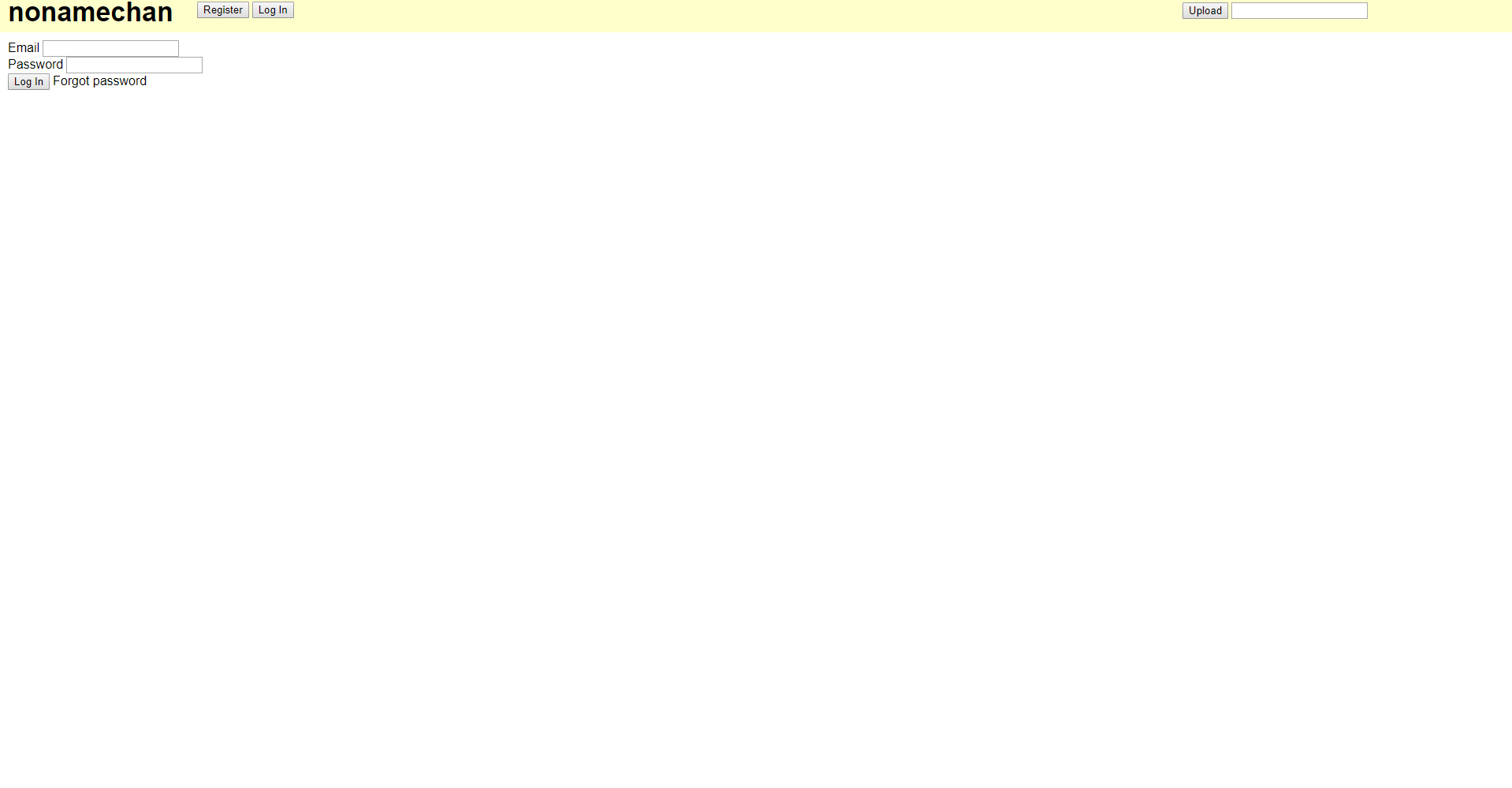
Весь проект стал доступен широкой аудитории благодаря сервису linode. Сервис предоставляет виртуальные машины с разной платой в зависимости от нагрузки. Я создал nanode ubuntu 18.10, в котором для выгрузки используются nginx и gunicorn. Nginx отвечает за загрузку статических файлов(например css), а gunicorn активирует код на python. Чтобы проект работал без моего участия используется supervisor. Этот пакет позволяет запустить процесс и закрыть терминал, так что мне не нужно постоянно быть в профиле виртуальной машины.

# Взаимодействие пользователя с системой

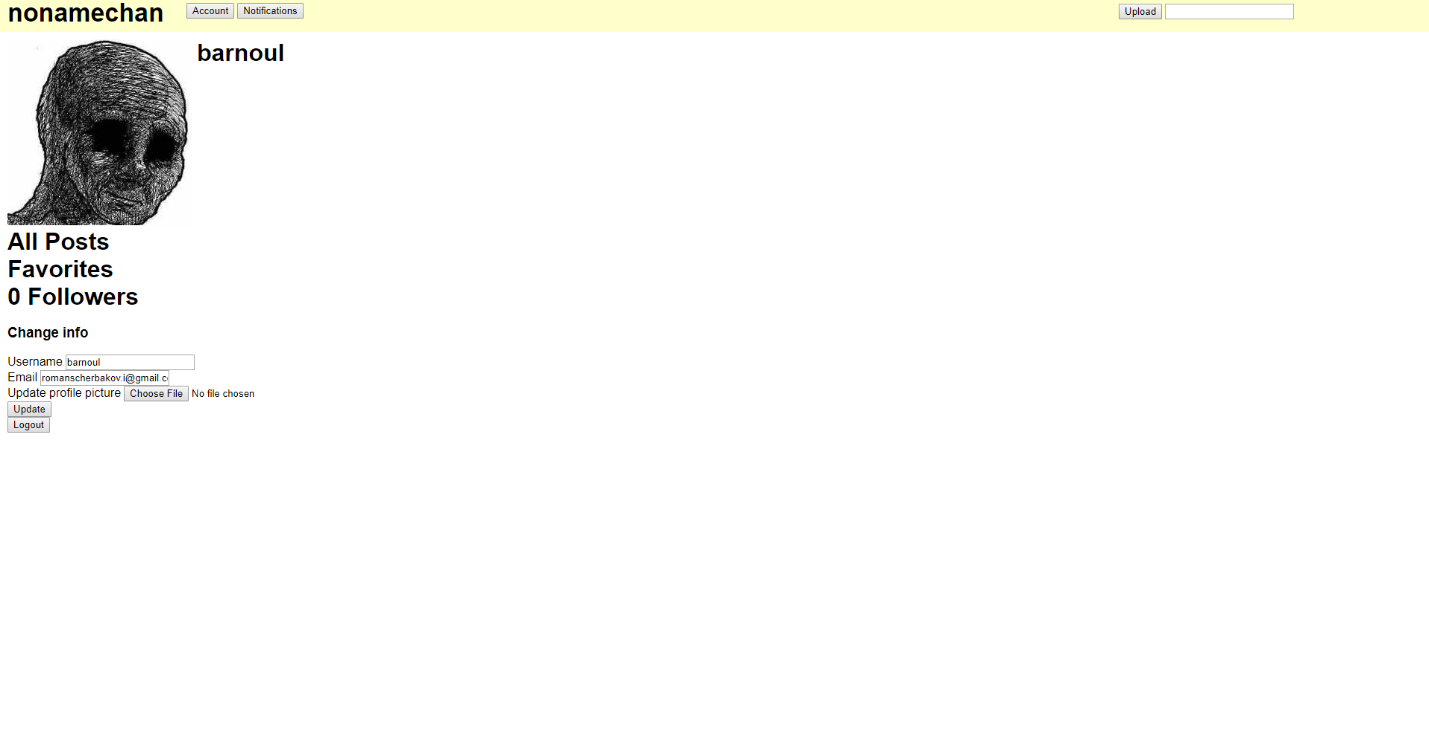
Первый этап – это регистрация(рисунок 4). Во время регистрации создается запись в таблице пользователей, в которой хранятся почта, имя, идентификатор, хешированный пароль. Пароль обработан алгоритмом одностороннего хеширования, поэтому никто из получивших доступ к базе данных не сможет его узнать. Пароль проверяется при входе в аккаунт(рисунок 5) посредством шифрования введенных данных и сравнения их с последовательностью символов в базе данных. На рисунке 6 изображен стандартный профиль пользователя.



*Рисунок 4. Регистрация*

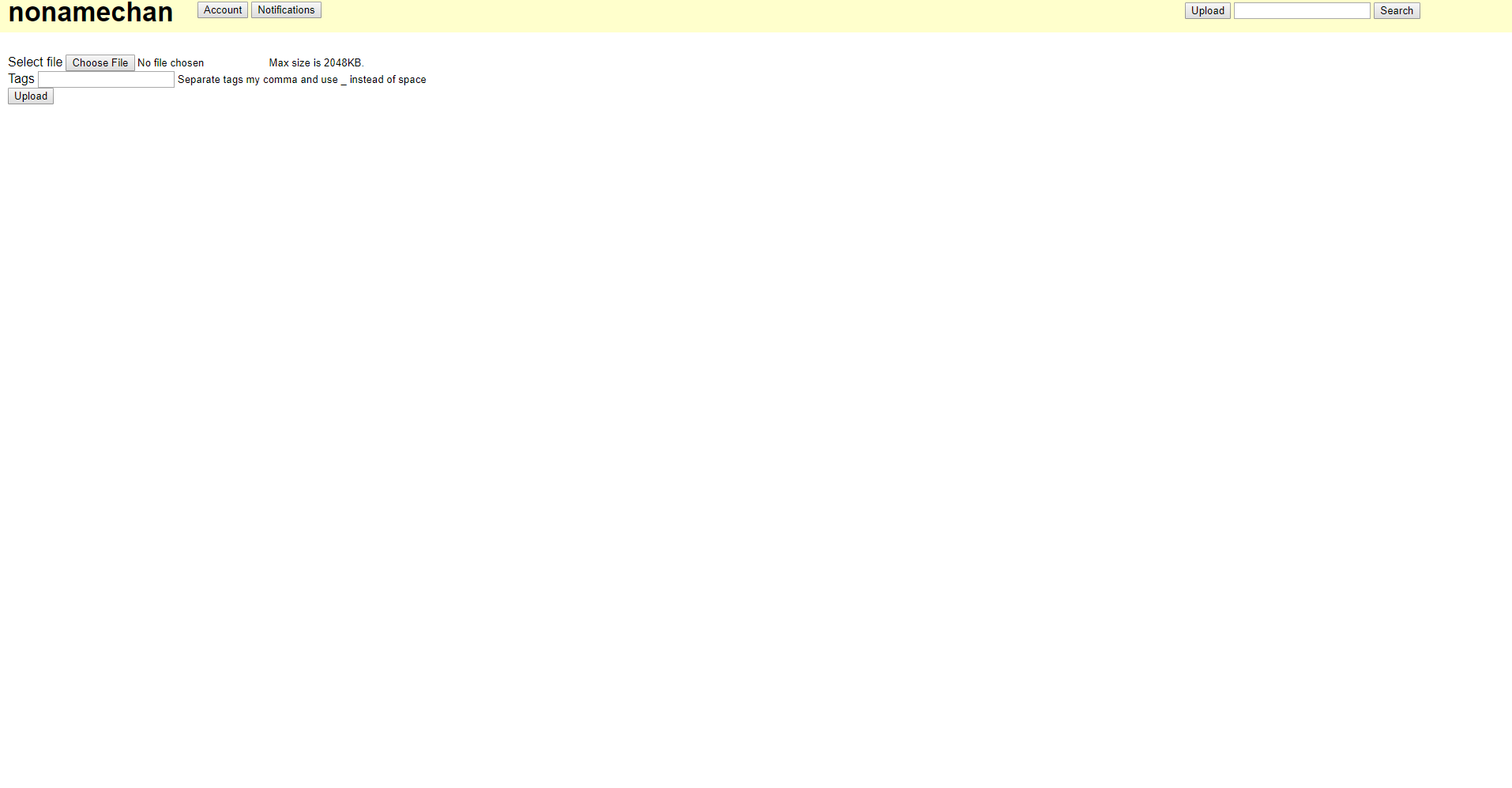


*Рисунок 5. Вход*



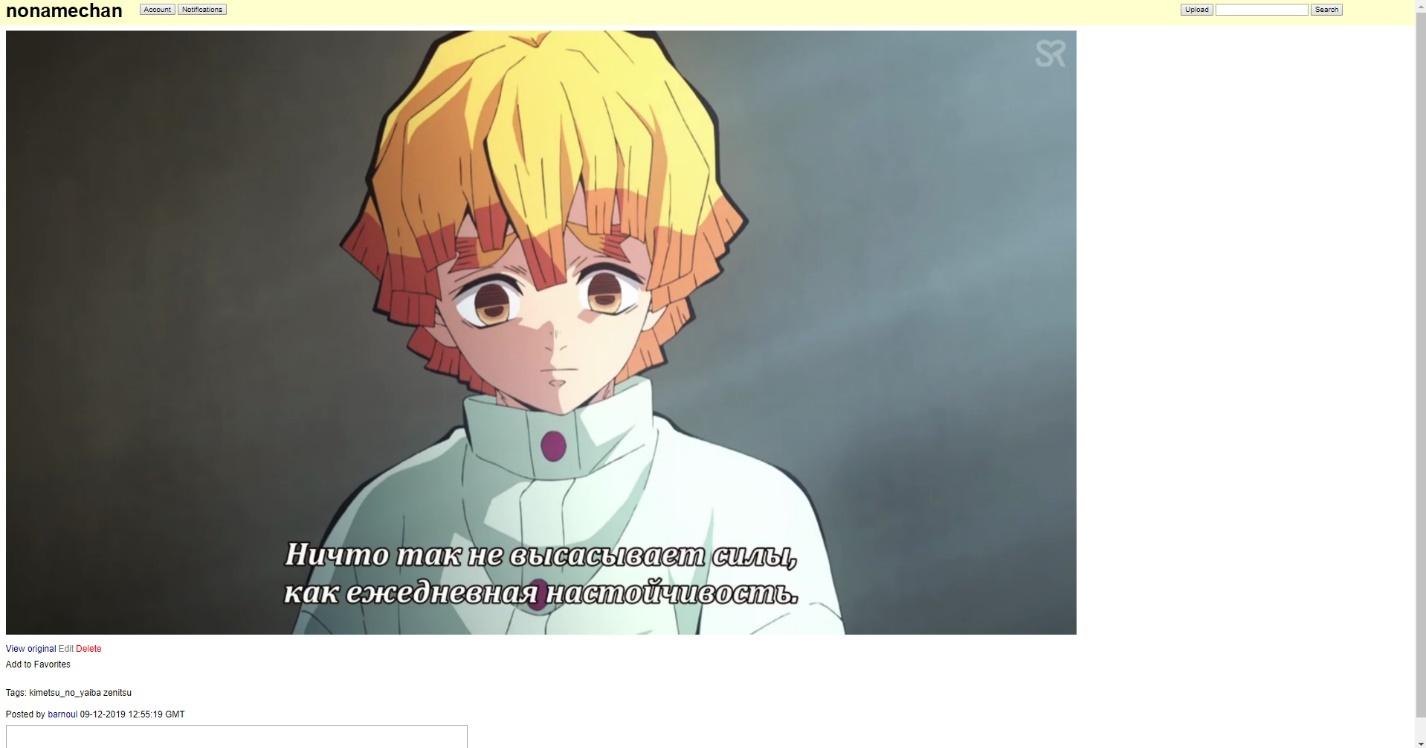
*Рисунок 6. Свой профиль пользователя*

Далее – загрузка картинки(рисунок 7), что могут сделать только зарегистрированные пользователи. Максимальный размер картинки ограничен 4МБ. Для загрузки используется FileForm из библиотеки WTForms. Пользователь обязан указать теги(ключевые слова) под картинкой, иначе загрузки не произойдет. При успешном выполнении всех требований картинка попадает на обработку, получает специальное имя файла – случайная последовательность символов длины 16, и создается запись в таблице постов со всей информацией о картинке. Автор картинки либо модератор может изменить теги под картинкой, и только автор или создатель системы могут удалить картинку.



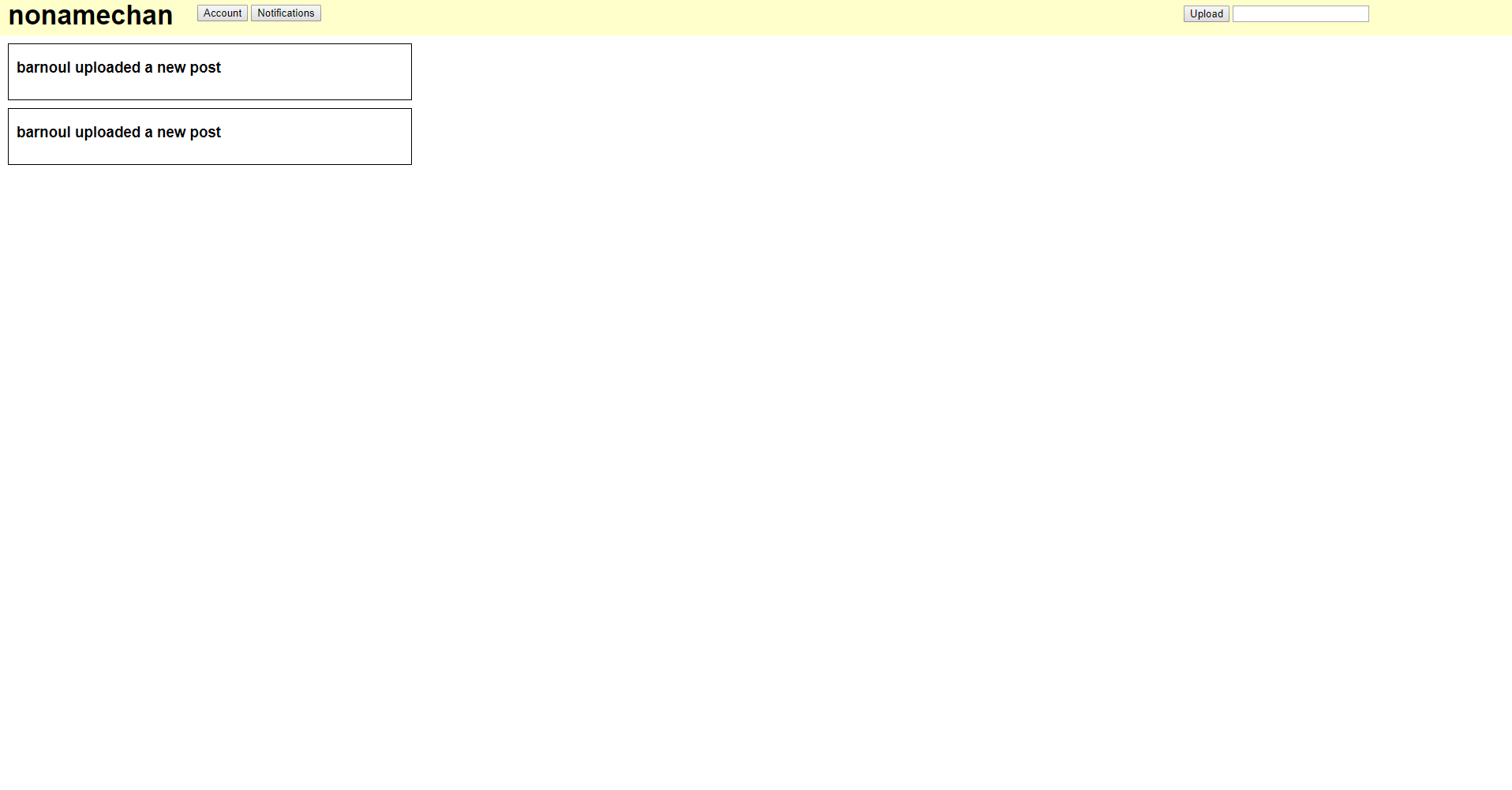
*Рисунок 7. Загрузка картинки*

Под любой картинкой зарегистрированный пользователь может оставить комментарий(рисунок 8). Здесь пользователям нужно быть осторожными, потому что если содержимое не понравится многим участникам либо лично администратору, то этого участника настигнет блокировка возможности загружать картинки и оставлять комментарии, а так же в его профиле откроется такой же раздел комментариев, где оставшиеся пользователи могут высказать свое недовольство. После снятия блокировки эти комментарии не исчезают, их нельзя удалить.

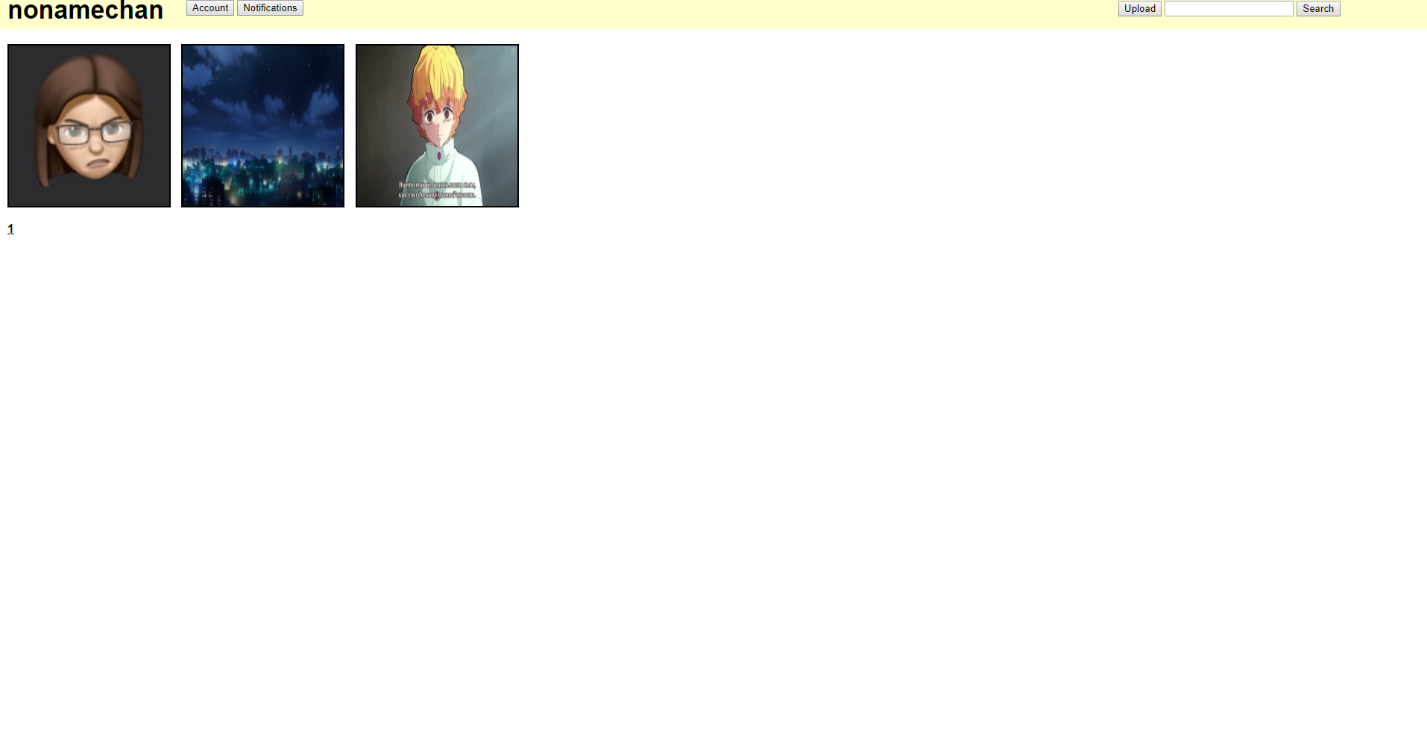


*Рисунок 8. Просмотр картинки*

Каждый пользователь может добавить понравившуюся картинку в избранное, она будет доступна по специальной ссылке в профиле. Избранное – простая связь многие-ко-многим пользователей и картинок. Так же работает и подписка на автора. При загрузке картинки автором и наличии подписчиков создается отдельная запись в таблице оповещений и связывается во всеми получателями(рисунок 9). Когда все подписчики просмотрят оповещение, оно удалится из таблицы. Домашняя страница на рисунке 10.

**

*Рисунок 9. Оповещения. Автор и post кликабельны*

*Рисунок 10. Домашняя страница*

Поиск по тегам(рисунок 11) казался сложным до того, как был реализован. Это тоже простая связь многие-ко-многим между сущностями пост и тег. Когда пользователь вводит ключевые слова в строку поиска либо переходит по ссылке на тег под постом в базу отправляется запрос с фильтром по тегу, если тегов несколько – операция повторяется рекурсивно, каждый раз отсеивая неподходящие выводы. После фильтрации все посты выводятся в порядке убывания по времени с версткой как на главной странице. Диаграма переходов на рисунке 12.

**

*Рисунок 11. Поиск по тегам*

A close up of a map

Description automatically generated

*Рисунок 12 - Переходы между интерфейсами*

# Заключение

При работе над проектом были изучены средства проектирования информационных систем и web-сервисов, обеспечения шифрования и защиты данных от несанкционированного доступа. Код лежит на <https://github.com/rischev/nonamechan>

# Список использованных источников

1. Schafer C. M. Python Flask Tutorial: Full-Featured Web App [Электронный ресурс] – URL: [https://www.youtube.com/playlist?list=PL-osiE80TeTs4UjLw5MM6OjgkjFeUxCYH](https://www.youtube.com/playlist?list=PL-osiE80TeTs4UjLw5MM6OjgkjFeUxCYH%20) (дата обращения 24.11.2019).
2. Pallets. Flask-SQLAlchemy Documentation [Электронный ресурс] – URL: <https://flask-sqlalchemy.palletsprojects.com/en/2.x/> (дата обращения 19.10.2019)
3. Ефимчик Е.А., Лямин А. В., Скшидлевский А.А. Основы разработки информационных систем – 2013