Техническая записка

InfoDesk является аналогом Википедии. Сайт выдает информацию по запросу пользователя.

На главной странице расположены строка для запросов и кнопка поиска. Под ними находится таблица с топ-10 статьями. При запросе слова идет поиск в базе данных. Если информация о запросе присутствует, то она берется из базы данных. В противном случае, пользователь сам может добавить информацию (при условии, если он авторизован), либо обратиться к Википедии, откуда и будет браться информация по запросу. Также на главной странице присутствуют кнопки личного кабинета, «выйти» и «добавить» - если пользователь авторизован, кнопки «зарегистрироваться» и «войти» - если авторизация не пройдена.

При регистрации от пользователя требуется указать следующую информацию: имя, фамилия, почта (она является логином), создать пароль и повторить его. Если все данные указаны корректно (почта является действительной, все остальные строки не являются пустыми), то создается аккаунт и пользователю присваивается уникальный токен (для API), который в дальнейшем можно изменить. Во время авторизации нужно указать почту(логин) и пароль, при желании можно поставить галочку «запомнить меня». Если пользователь введет верные данные, то он будет перенаправлен на главную страницу. При нажатии кнопки «выйти» происходит выход из личного кабинета. Регистрация дает пользователю возможность оценивать информацию, добавлять новую и оставлять комментарии.

В личном кабинете указывается информация о пользователе: его имя и фамилия, почта, дата создания профиля, статус пользователя, токен и его работы, если они есть (дата создания, запрос и количество лайков). Существует три уровня пользователей – базовый, модератор и администратор. Базовый уровень выдается при регистрации, модератор – если работы пользователя набрали 5 лайков. Модератор, в отличии от базового пользователя, имеет возможность редактировать. Администратор имеет доступ к жалобам (специальная кнопка жалобы в личном кабинете) и может скрывать работы. Жалобы видны всем администраторам. К жалобам админу можно добавлять комментарии и отмечать жалобы прочитанными. Уровень администратора не выдается.

Информацию можно оценивать – ставить лайки и оставлять жалобу. Лайки можно и удалять, если снова нажать на него. Жалоба отправляется на рассмотрение администратору.

На страницу добавления информации можно выйти двумя путями – кнопка будет доступна на главной странице (при условии регистрации) и на странице поиска информации, если её нет в базе данных. Чтобы расширить данные, пользователю нужно заполнить форму (добавить информацию или загрузить файл). Если добавить информацию в форме, файл не будет учитываться. Форма состоит из следующих частей: строка для введения основного слова, строка для введения дополнительных слов для поиска (формат: слово; слово) и введение самой информации. При добавление файла требуется указать главное слово, выбрать файл и отправить его.

Требования:

При написании нашего сайта мы использовали orm-модели и базу данных. Вся информация о пользователях, комментариях, жалобах и лайках хранится в ней. Информация, которую загружают наши пользователи, хранится в отдельных html-файлах, ссылки на которые хранятся также в бд.

Наш сайт написан с использованием html-шаблонов, стилей bootstrap и дополнительных стилей собственной разработки.

Разработчики могут использовать API нашего сайта для удобства получения информации.

Информация о версиях используемых библиотек хранится в файле requirements.txt. Главный файл - main.py

## Классы:

* User - отвечает за информацию о самом пользователе, его личный кабинет и его возможности в зависимости от статуса
* Information - отвечает за добавление новой и показ старой информации
* Comment - отвечает за показ и добавление комментариев
* Complaints - отвечает за передачу жалобы пользователей на информацию администратору
* Likes - отвечает за оценивание работ
* Word - отвечает за поиск информации
* API-token - отвечает за выдачу информации (и жалоб для админов) с помощью REST-запросов

## API

Во всех запросах в headers нужно указать токен пользователя!!!

### Базовый уровень:

* Запрос основной информации по слову: адрес “/api/main\_information/word”, где word - слово, по которому ищется информация. В случае наличия нескольких информаций возвращается та, которая имеет наибольшее количество лайков (GET-запрос)
* Запрос всей информации по слову: адрес “/api/all\_information/word”, где word - слово, по которому ищется информация. Возвращается вся информация, которая есть по этому слову (GET-запрос)
* Добавить информацию: адрес “/api/information”. Формат тела запроса:

{

"text": "text",

"word": "new\_word",

"words": **null**

}

“text” - текст информации

“word” - Основное слово для поиска

“words” - Дополнительные слова для поиска в формате “слово; слово” или null

(POST-запрос)

### Администратор:

* Блокировка информации по адресу: адрес “/api/information/address”, где address - адрес страницы, которую блокируют. (DELETE -запрос)
* Запрос всех жалоб: адрес “/api/complaints”. (GET-запрос)
* Запрос жалоб по номеру: адрес “/api/complaints/number”, где number - номер жалобы, которую нужно получить (GET-запрос)
* Поставить статус жалобе (прочитано/не прочитано): адрес “/api/complaints/number”, где number - номер жалобы, которую вы хотите отметить. Формат тела запроса:

{

"is\_reading": **true**

}

(PUT-запрос)