

Отчет о выполнении тестового задания на вакансию стажер-разработчик Python.

Задание

Реализовать сервис для хранения файлов пользователя, организовать структуру директорий. UI не обязателен, достаточно хорошей документации на API. Сервис должен предоставлять следующую функциональность:

- Создание нового пользователя
- Авторизация пользователя
- Удаление пользователя
- Обновление данных авторизации
- Сохранение файла
- Удаление файла
- Обновление файла
- Получение файла
- Создание директории
- Удаление директории
- Смена имени директории
- Получение списка файлов в директории

Язык программирования: python.

Код приложения необходимо разместить на любом публичном веб-сервисе для хранения проектов.

Оглавление

Уровень выполнения	2
Используемые программные средства	3
Описание приложения	4
От автора	6

Уровень выполнения

Создание нового пользователя

Реализовано. Осуществляется переходом по ссылке «Registration», вводом данных и подтверждением. Сразу создается папка пользователя, где будут храниться его файлы.

Авторизация пользователя

Реализовано. Осуществляется переходом по ссылке «Sign In» или при переходе на страницы, доступ к которым требует подтверждения личности.

Удаление пользователя

Реализовано. Доступно после авторизации посредством нажатия «Delete account». Папка пользователя удаляется вместе с записью о нем из базы данных.

Обновление данных авторизации

Реализовано. Доступно после авторизации посредством нажатия «Edit personal data». Если пользователь изменяет логин, одноименная папка также переименовывается.

Сохранение файла

Реализовано.

Удаление файла

Не реализовано.

Обновление файла

Не реализовано.

Получение файла

Не реализовано.

Создание директории

Не реализовано.

Удаление директории

Не реализовано.

Смена имени директории

Не реализовано.

Получение списка файлов в директории

Реализовано. Список файлов и список директорий отображается при нажатии «Files and folders list».

Отсутствие данных функций работы с директориями связано с тем, что автор пока не нашел инструментов для интерактивной работы с иерархией папок. Однако у автора имеется представление о логике реализации необходимого функционала после преодоления этого препятствия.

Поскольку функция загрузки файлов реализована, и остальные функции для работы с файлами связаны с данной, в их успешной реализации при устранении предыдущих слабых мест автор также не сомневается.

Тестирование было проведено в качестве апробирования программы с применением различных браузеров, а также ручное тестирование процессов авторизации, регистрации и обновления данных.

Используемые программные средства

Язык программирования: python (2.7).

Средство разработки: Flask (0.12).

Используемые модули и зависимости:

Jinja2

Werkzeug

WTForms

MongoEngine (0.13.0)

PyMongo (2.8)

Flask-Login

Bootstrap

База данных: MongoDB.

Написано и тестировалось в ОС Windows 8.1.

Проверялось с использованием браузеров Google Chrome, Mozilla Firefox, Internet Explorer (работает и отображается одинаково во всех трех).

Описание приложения

Приложение представляет собой веб-сайт, стартовая страница которого страница входа в систему выглядит следующим образом.

Сайт подразумевает авторизацию. Если у вас пока не имеется аккаунта, его можно завести, выбрав на верхней панели ссылку на регистрацию «Registration».

После входа или регистрации пользователю становится доступна основная страница: его хранилище файлов.

Также после авторизации появляются новые возможности по управлению аккаунтом: его удаление и обновление введенных при регистрации данных.

Все поля проходят интерактивную валидацию.

При регистрации: поля login и email должны быть уникальны.

Также на всех этапах:

- логин не больше 48 символов;
- пароль от 6 до 35 символов;
- почтовый адрес от 6 до 70.

Из недоделанной части (папки и файлы созданы в системе вручную, однако их отображение – программная логика):

- левая часть представляет собой древовидное отображение иерархии папок, созданных пользователем в своем аккаунте;
- правая часть – файлы, внутри каждой папки.

Кнопка «Выбрать файлы» предоставляет пользователю доступ к его файловой системе для выбора файлов, которые будут загружены на сайт, кнопка «Upload» выполняет отправку этих файлов на сервер.

На диске E:\ сервера имеется папка fss (file storage system), именно в ней в последствии создаются папки пользователей при регистрации, в которых размещаются файлы каждого авторизованного юзера.

Приложение запускается с помощью вызова на выполнение в командной строке файла run.py из корневой папки программы.

В основе лежит база данных, включающая в себя один документ – информация о пользователе и один вложенный документ – информация о файлах пользователя. Структура базы данных представлена ниже.

```
{
  "_id" : ObjectId<"59b57ff093b63f0db464d6d3">,
  "login" : "diana",
  "pswd" : "123456",
  "email" : "diana@mail.ru",
  "role" : 0,
  "confirmed_at" : ISODate<"2017-09-10T21:09:52.624Z">,
  "dirsp" : [ 1,
  "flsp" : [
    1,
    "rKPNd0CeQiw.jpg"
  ],
  "fls" : {
    "flnm" : "rKPNd0CeQiw.jpg",
    "flsz" : 33822,
    "flpth" : "E:\\fss\\diana\\rKPNd0CeQiw.jpg",
    "create_at" : ISODate<"2017-09-10T21:13:46.108Z">
  }
}
```

В документе User хранится информация о пользователе:

- логин
- пароль (да, не самый защищенный вариант, но в момент создания безопасность была второстепенной проблемой)
- email адрес
- роль (гость – 0, администратор – 1)
- дата регистрации
- список папок пользователя
- список файлов пользователя
- fls (files) - вложенный документ с информацией о файлах каждого пользователя:
 - имя файла
 - размер файла
 - путь к файлу на сервере
 - дата создания

В программе имеются некоторые отладочные моменты (например, flash сообщения в ответ почти на все действия пользователя), которые автор решил оставить, поскольку продукт не доработан.

Для разворачивания экспериментального стенда:

- 1) развернуть виртуальную машину
- 2) установить сервер Apache
- 3) произвести соответствующие настройки (установка ПО, кастомизация)
- 4) запустить приложение

В репозитории, где выложен проект, есть папка «Пример работы», в ней находятся скриншоты с примерами сайта.

От автора

Хотелось бы обратить внимание на то, что все знания и навыки, необходимые для выполнения данного задания приобретались с момента его получения, все средства и инструменты постигались с нуля. Не оправдание, просто чтобы вы знали.

Кроме того, спасибо за это задание и моё желание попасть в вашу команду, благодаря им я освоила много полезного.

Хорошего дня ☺