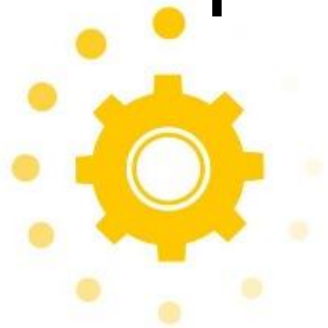
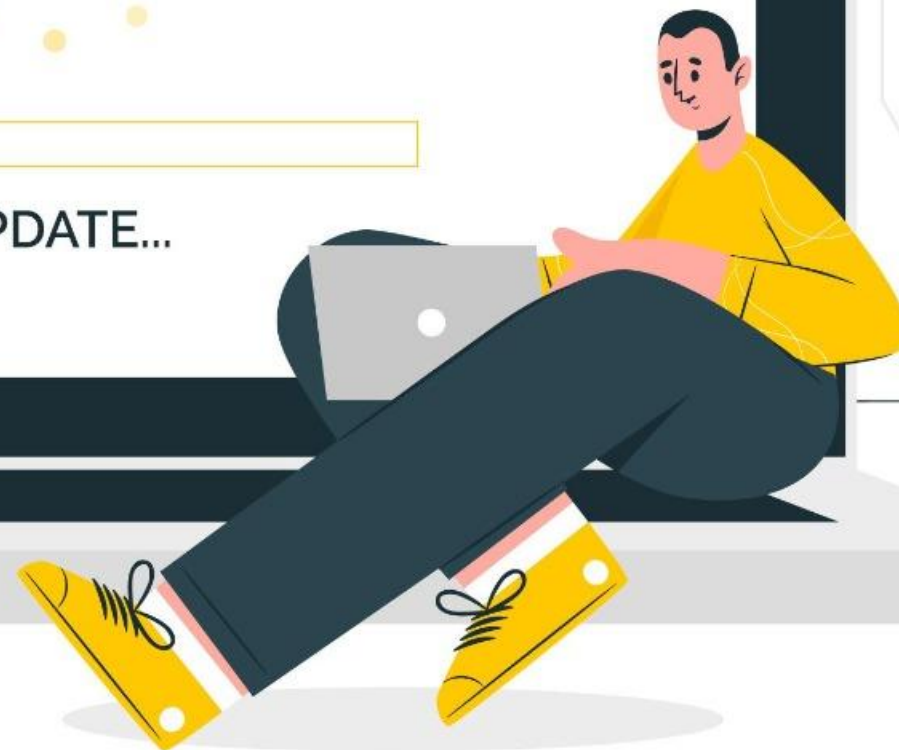


Дипломная работа Облачное хранилище



UPDATE...

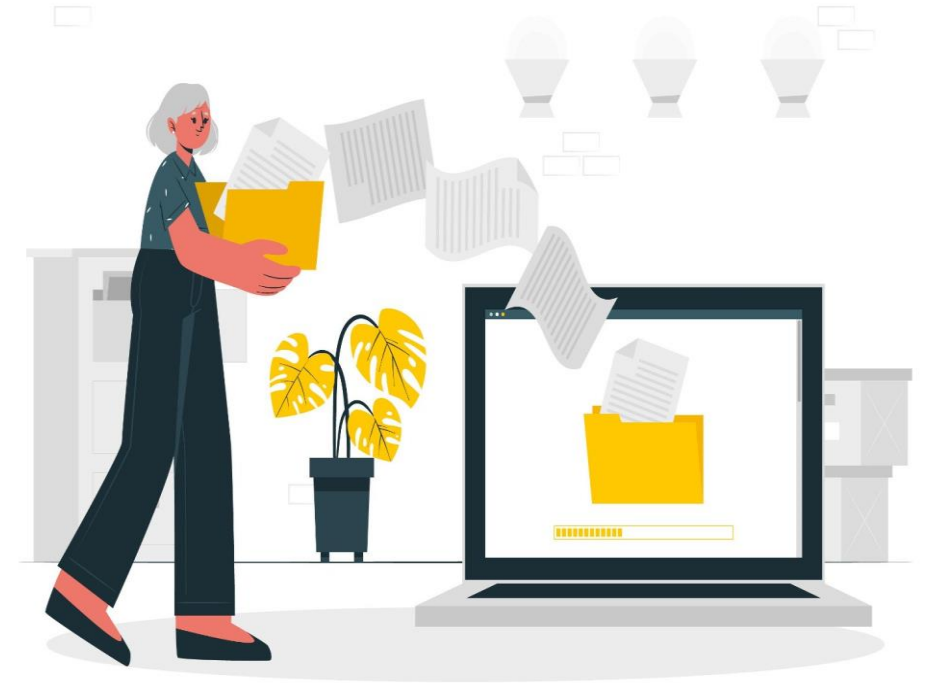


Цели проекта

Создать веб-приложение, которое будет работать как облачное хранилище. Приложение позволит пользователям отображать, загружать, отправлять, скачивать и переименовывать файлы.

Комплексное веб-приложение, будет в себя включать:

- Бэкенд на языке Python с использованием фреймворка Django и СУБД PostgreSQL;
- Фронтенд на языках JavaScript, HTML, CSS с использованием библиотек React, Zustand, React Router.



Бэкенд

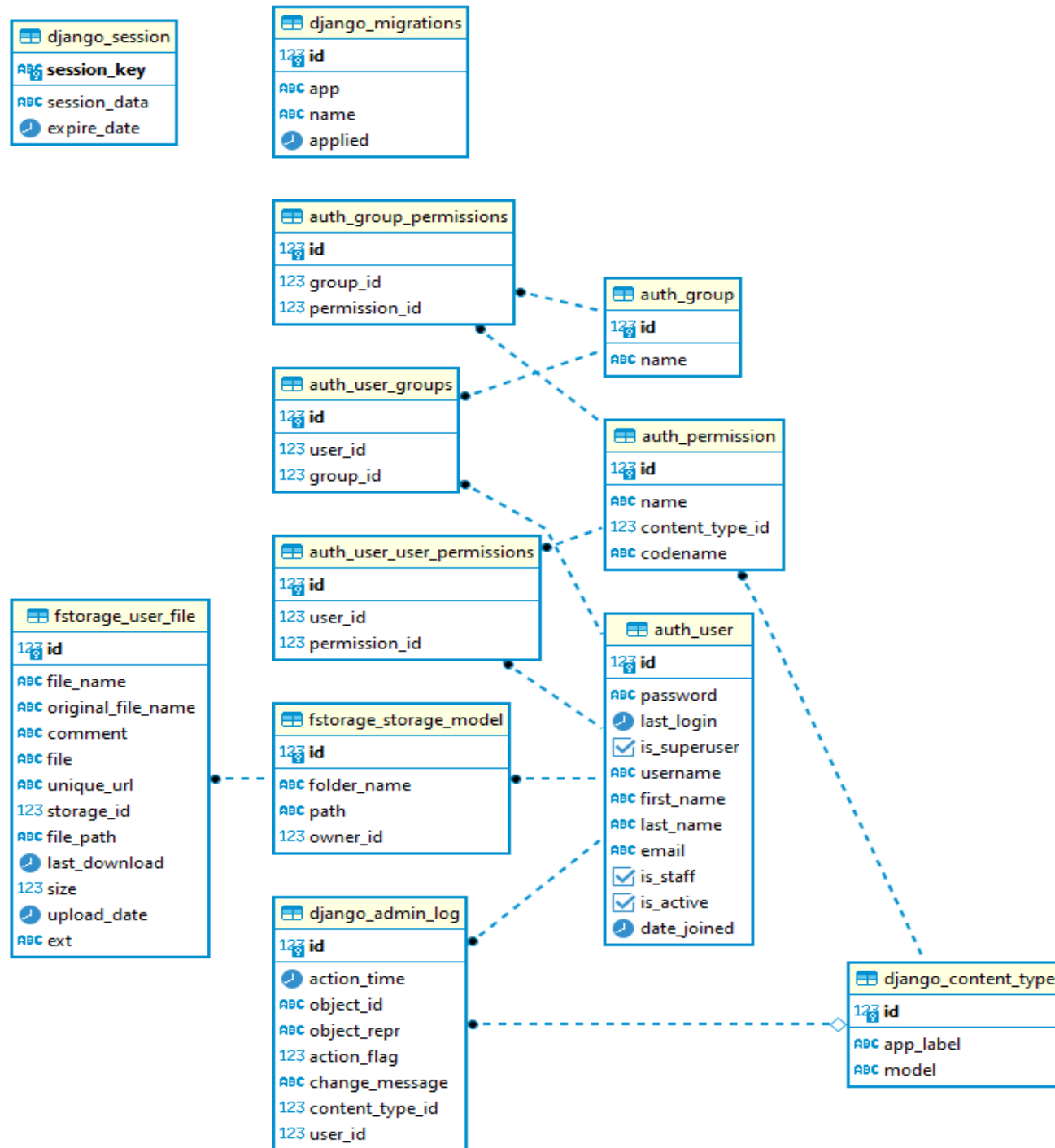
- Бэкенд проекта реализован на Python с использованием фреймворка Django и СУБД Postgres для хранения информации.
- Загрузка статических ресурсов (HTML, CSS, JS-файлы фронтенда), а также API-вызовы обрабатываются единым сервером
- Все API-вызовы соответствуют семантическим правилам для REST API, для обмена данными между фронтендом и бэкендом используется формат JSON.
- Сервер содержит реализацию бэкенда для двух основных блоков приложения: административный интерфейс и работа с файловым хранилищем.
- Данные о пользователях хранятся в таблице БД в полях, имеющих соответствующие им типы: логин, полное имя, email, пароль, признак администратора, путь к хранилищу пользователя относительно общего пути к хранилищу файлов.
- Все вызовы, кроме регистрации пользователя, защищены проверкой аутентификации пользователя в системе.
- Функция удаления пользователей доступна только пользователю, имеющему признак администратора.
- Состояние аутентификации пользователя отслеживается через сохранение информации о сессии.
- Все API-вызовы проверяют права доступа пользователя и возвращают соответствующие ошибки через HTTP-статус и сообщение в формате JSON.
- Все события сервера логируются путём вывода на консоль сообщений «debug», «info», «warning», «error» с указанием даты и времени, содержат информацию, достаточную для анализа работоспособности и отладки сервера.



База данных

База данных состоит из нескольких таблиц, в которых хранятся различные типы данных, связанных с пользователями и загруженными ими файлами. Некоторые из этих таблиц включают:

- Таблица «Пользователи»: в этой таблице хранится основная информация о каждом клиенте, такая как имя, адрес электронной почты, хэш пароля и т. д.
- Таблица файлов: в этой таблице хранятся данные о каждом файле, загруженном клиентом, включая имя файла, размер файла, отметку времени загрузки и т. д.
- Таблица «Хранилище»: в этой таблице записываются данные о существующих файловых хранилищах, их владельцах и месторасположении в файловой системе.
- Эти таблицы работают вместе, чтобы обеспечить полное представление обо всех данных пользователей в приложении.
- База данных оптимизирована для быстрого запроса и извлечения релевантных данных, обеспечивая бесперебойную работу даже при высокой нагрузке.



Фронтенд

Фронтенд реализован на на языках JavaScript, HTML, CSS с использованием библиотек React, Zustand, React Router.

Реализована концепция SPA (single page application) — весь переменный контент на странице (списки пользователей и файлов и т.п.) формируется кодом на JavaScript с использованием библиотеки React.

Для получения данных задействованы асинхронные API-вызовы к серверу приложения.

FREE
file host

Бесплатный сервис для хранения файлов

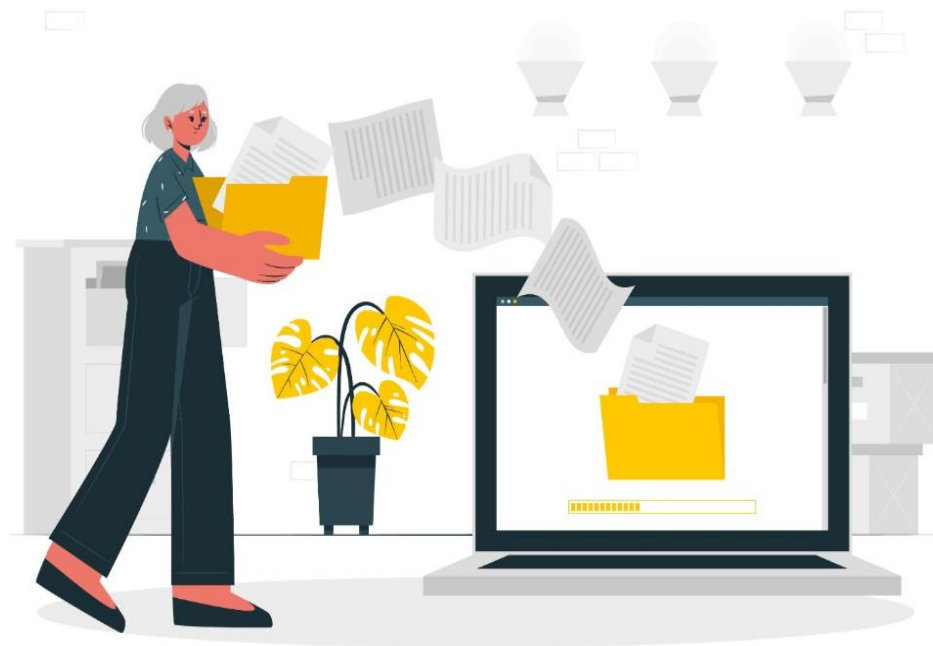
Чтобы пользоваться хранилищем
нужна регистрация

Зарегистрироваться

или

Войти

[Вход](#) | [Регистрация](#)



Формы

Формы имеют функциональный дизайн.

В формах присутствуют выпадающие подсказки.

Форма регистрации проверяется основные ограничения на значения указанных полей:

- логин — только латинские буквы и цифры, первый символ — заглавная буква, длина от 4 до 20 символов;
- email должен соответствовать формату адресов электронной почты — для проверки можно использовать регулярные выражения;
- пароль — не менее 6 символов: как минимум одна заглавная буква, одна цифра и один специальный символ.

FREE
file host

Вход

Логин

Необходимо заполнить

Пароль

Необходимо заполнить

Войти



[Вход](#) | [Регистрация](#)

FREE
file host

Регистрация

Придумайте логин

Логин должен начинаться с большой буквы

Имя

Фамилия

Email

Необходимо заполнить

Пароль

Необходимо заполнить

Повторите пароль

Зарегистрироваться



[Вход](#) | [Регистрация](#)

Пользовательский интерфейс

- Пользовательский интерфейс интуитивно понятен, обладает всеми необходимыми функциями, всплывающими подсказками.
- Файлы можно загружать при помощи Drag and Drop и добавлять комментарии на стадии загрузки.
- У загруженных файлом можно редактировать названия, комментарии.
- Файлы можно удалять, скачивать, передавать ссылки для скачивания
- Ссылки обезличены и не содержат названия или пути к файлу

•

FREE
file host

Хранилище
Пользователи

id	original_file_name	comment	size	external url	download	delete	last download	upload date
231	1.jpg	<input type="text"/> Save	4.41 КБ	Copy Url	Download	Dell		2023-08-09
232	2.jpg	<input type="text"/> Save	4.44 КБ	Copy Url	Download	Dell		2023-08-09
233	3.jpg	<input type="text"/> Save	7.34 КБ	Copy Url	URL copyad	Dell		2023-08-09

5.jpg
comment

Drag and drop files

Загрузить

Сохранить

Выйти

Интерфейс администратора

- Интерфейс администратора аналогичен пользовательскому интерфейсу.
- Интерфейс отображает список пользователей с выводом признака «администратор» и информации, указанной пользователем в форме регистрации (кроме пароля). В списке присутствует возможность удаления пользователей и изменения значения признака «администратор».
- В списке пользователей также отображается информация о файловых хранилищах пользователей: количество и размер файлов, ссылка для перехода к интерфейсу управления этими файлами.

FREE
file host

Хранилище	id	UserName	email	admin	storage	FullName	Dell User
Пользователи	107	Test_use27	example@example.ru	Admin	107 files-3 size - 16.19 KB	Asd User	Dell
	105	Test_use99	email@email.com	Admin	105 files-0 size - 0	UserName UserLatName	Dell
	87	root		Admin	"None"		Dell

Выйти