Утверждаю Лист утверждений

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Галкин В.А.

"\_\_\_"\_\_\_\_\_\_\_\_\_2021г.

Техническое задание

" Сервер заказов с хранением видео "

по курсу "Сетевые технологии в АСОИУ"

Вариант №5

Исполнители:

Мартынова П.В. ИУ5-61Б

Агличеев М.С. ИУ5-61Б

Москва 2021 г.

1. Наименование:

Сервер заказов с хранением видео.

1. Основание для разработки:

Основанием для разработки является учебный план

МГТУ им. Баумана кафедры ИУ5 на 6 семестр.

1. Исполнители:

Исполнителями являются студенты МГТУ им. Н.Э. Баумана группы ИУ 5-61Б: Мартынова П.В.(frontend) Агличеев М.С.(backend)

1. Цель разработки:

Автоматизированная система предназначена для осуществления возможности оформления заказа, просмотра видео, хранящихся на сервере.

1. Содержание работы:
   1. Авторизация пользователей

Авторизация пользователей осуществляется путем ввода в графическом интерфейсе логина и пароля и нажатия кнопки «Войти». После выполнения данных операций в графическом интерфейсе, выполняется асинхронный запрос на Backend. В случае успешной авторизации Backend формирует токен авторизованного пользователя (также называемый пользовательской сессией), который сохраняет в кэш и выдает на Frontend в качестве cookie.

* 1. Предусмотреть разделение пользователей на администраторов и обычных пользователей

В таблице пользователей содержится поле-флаг, которое характеризует: является ли данный пользователь администратором или нет.

Авторизация администраторов осуществляется путем ввода в графическом интерфейсе логина и пароля и нажатия кнопки «Войти». После выполнения данных операций в графическом интерфейсе, выполняется асинхронный запрос на Backend. В случае успешной авторизации Backend формирует токен авторизованного пользователя, который сохраняет в кэш и выдает на Frontend.

Стоит отметить, что токен администратора отличается от токена обычного пользователя. Это означает, что с токеном администратора можно выполнять только функции, разрешенные администратору, а с токеном обычного пользователя – только функции, разрешенные обычному пользователю.

* 1. Просмотр каталога товаров

Frontend формирует асинхронный запрос на получение данных из API, данные выводятся на экран.

* 1. Предусмотреть интерфейс администраторов для добавления элементов в API

Интерфейс представляет из себя CRUD-обёртку: любой объект в базе данных можно создать, убрать (путём выставления соответствующего флага), или изменить (фактическое удаление не предусмотрено). Внешне это делается посредством экранных форм.

При изменении данных автоматически происходят проверки на внешние связи объектов в БД, уникальность и прочее.

* 1. Возможность добавления товара в избранное и корзину

При нажатии на иконку избранного на карточке товара данный товар можно увидеть во вкладке избранного, аналогично с корзиной. Из вкладок избранного и корзины товар можно удалить также нажатием на соответствующую иконку.

Frontend реализует данную функцию с помощью хука UseState. При нажатии на соответствующую иконку состояние переменной меняется с False на True, состояние передается в виде пропса в компонент карточки товара.

* 1. Возможность оформления заказа

Пользователь может добавить товар в корзину на странице товара и удалить на странице корзины. После добавления всех необходимых товаров, по клику по кнопке в корзине можно перейти к оформлению заказа. С помощью экранной формы пользователь заполняет данные о себе и формирует заказ. Данные о новом заказе отправляются в базу данных, администратору приходит уведомление о новом заказе.

* 1. Страница для просмотра видео, хранящихся на сервере

Видео хранится в S3(протокол удаленного доступа к файловому хранилищу)-хранилище, ссылка на него – в базе данных на основном сервере. Когда пользователь (администратор) загружает видео, основной сервер подключается к s3-хранилищу и загружает туда полученный файл, сохраняя итоговую ссылку в БД и выдавая её по запросу от Frontend.

1. Требования к составу технических средств:

Для реализации frontend части используется язык TypeScript с использованием библиотеки React. Backend реализуется на Go, в качестве СУБД используется PostgreSQL.

1. Этапы разработки:
   1. Выбор темы – до конца 1 недели
   2. Разработка ТЗ с форматами обмена – до конца 9 недели
   3. Разработка остальной документации – до конца 12 недели
   4. Защита проекта – до конца 15 недели
2. Техническая документация, предъявляемая по окончанию работы:
   1. ТЗ
   2. РПЗ
   3. ПМИ
   4. Руководство системного администратора
   5. Руководство пользователя. Инструкция для пользователя со скриншотами и кратким описанием основных функций системы.

9. Порядок приемки работы:

Приемка работы осуществляется в соответствии с "Программой и методикой испытаний."

Работа защищается перед комиссией преподавателей кафедры.

10. Дополнительные условия:

Данное Техническое Задание может дополняться и изменяться в установленном порядке.