МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ ФГБОУ ВО «ОРЛОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ И.С.ТУРГЕНЕВА»

Кафедра «Информационные системы и технологии»

**Отчет**

По лабораторной работе №2

На тему: «Обоснование выбора информационных технологий для реализации программного продукта»

По дисциплине: «Требования к программному обеспечению и спецификации»

Работу выполнила: Куксенко Д. С., Стрижова Е.В.

Институт приборостроения, автоматизации и информационных технологий

Направление/специальность: 09.03.04 «Программная инженерия»

Группа: 22ПГ

Преподаватель Олькина Е.В.

Отчет защищен с оценкой «\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_»

Орёл, 2024

**Основные компоненты системы:**

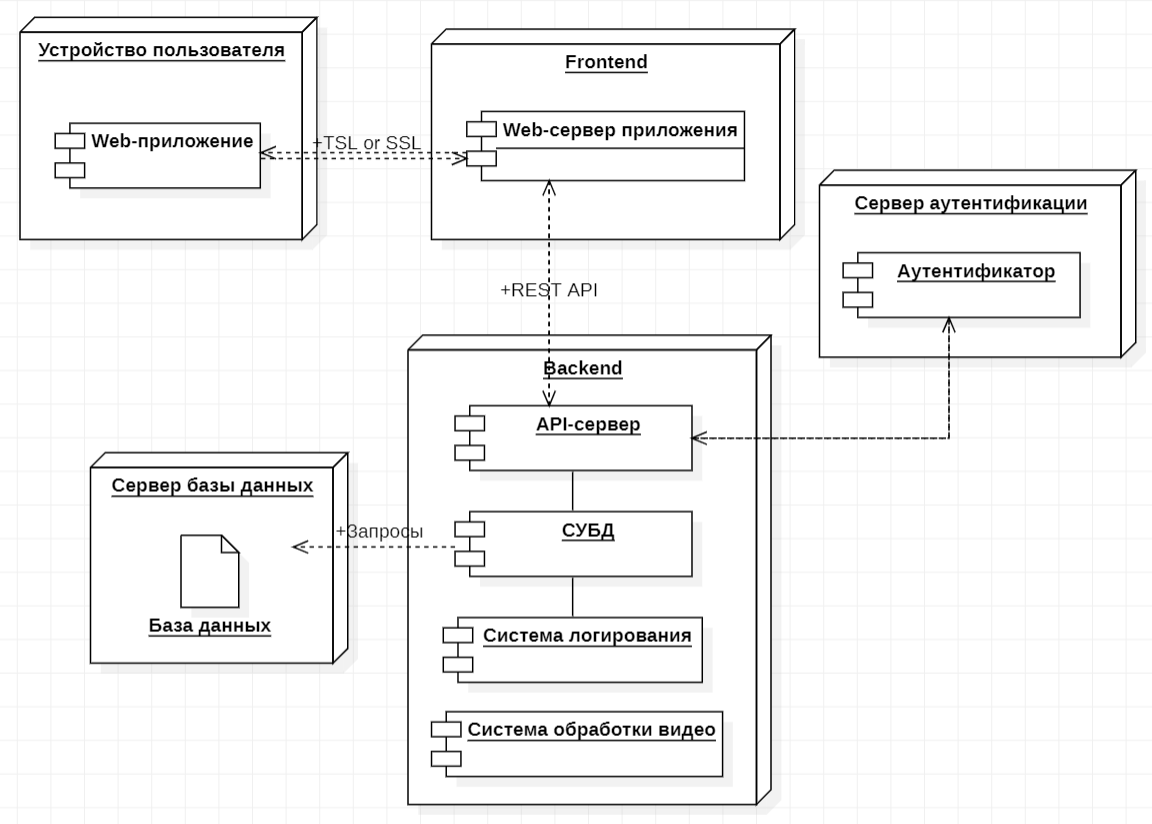
1. Клиентское веб-приложение с механизмом геолокации. Веб-приложение будет обеспечивать интуитивно понятный интерфейс для пользователей, позволяющий им просматривать, фильтровать и искать видео. Механизм геолокации будет встроен для определения местоположения пользователя и для дальнейшего применения необходимых ограничений.
2. Сервер веб-приложения. Сервер будет обрабатывать запросы от клиента, управлять взаимодействием с базой данных и API, а также обеспечивать безопасность приложения: функционал сервера должен содержать программную логику, ограничивающую доступ к контенту в зависимости от страны и возраста, а также настройку для приема запросов на администрирование только с заранее определенных IP-адресов.
3. Механизм аутентификации. Обязательная система аутентификации в системе позволит обеспечивать безопасность, разрешая доступ к административным функциям только авторизованным пользователям с определенным IP-адресом, и предотвратить попытки несанкционированного доступа к данным. Кроме того, система аутентификации поможет предоставить пользователям персонализированный опыт использования приложения, например, формируя рекомендации видео на основе их истории просмотров и взаимодействия с контентом.
4. API-сервер: мост между клиентским веб-приложением и сервером, взаимодействует с различными сервисами для выполнения запросов: с сервером аутентификации, хранилищем данных и сервером базы данных.
5. База данных: хранение всей необходимой информации о видео, пользователях и их взаимодействии.
6. Система безопасности: обеспечит защиту данных и действий в приложении путем шифрования паролей пользователей и их личных данных, логирования всех совершаемых ими действий, включая попытки входа, запросы на администрирование и доступ к видео, что поможет выявлять возможные нарушения безопасности.

**Программные средства:**

1. Язык программирования и фреймворки:
   * JavaScript: позволяет использовать один язык на стороне клиента и сервера, что упрощает разработку и снижает количество контекстных переключений, кроме того позволяет использовать готовые библиотеки для работы с геолокационными данными, что снижает стоимость разработки. Кроме того, JavaScript предоставляет возможность асинхронной обработки запросов, что особенно актуально в контексте видеохостинга.
2. Веб-сервер
   * Node.js + Express: Позволяет создать масштабируемый и асинхронный серверный код, что делает его идеальным для выполнения многопользовательских запросов, которые будут обрабатывать API для взаимодействия с фронтендом и базой данных.
3. Клиентское веб приложение:
   * React или Vue.js: Эти фреймворки создают динамичные пользовательские интерфейсы, которые могут легко взаимодействовать с API и обновлять данные в реальном времени, обеспечивая положительный пользовательский опыт.
   * GeoIP библиотеки (например, geoip-lite для Node.js): Позволяют определить страну пользователя на основе его IP-адреса, что критично для ограничения доступа к контенту в зависимости от страны пользователя.
4. СУБД:
   * PostgreSQL или MySQL. Обе СУБД позволяют создать реляционная базу данных, подходящую для хранения информации о видео и пользователях, а также выполнения сложной логики выборки.
5. Аутентификация и авторизация:
   * JWT (JSON Web Tokens) для безопасного управления сессиями пользователей. Это позволит предоставлять доступ к контенту только авторизованным пользователям.
6. API-сервер:
   * Express.js: Фреймворк для Node.js, который упрощает создание API, предоставляет удобные средства маршрутизации и обработки запросов.
7. Система безопасности
   * TLS/SSL: Эти протоколы шифрования используются для обеспечения безопасности данных, передаваемых между клиентом и сервером. Использование HTTPS вместо HTTP предотвращает прослушивание и подмену данных в процессе передачи данных для аутентификации или трансляции видео.
   * Использование библиотек для логирования, таких как Winston или Morgan для Node.js позволит отслеживать события и ошибки в веб-приложении.
8. Хранение видеофайлов
   * Облачные технологии: Amazon S3 или Google Cloud Storage. Они представляют собой надежное и масштабируемое решение благодаря своей высокой доступности.

**Обоснование выбора технологий:**

* Гибкость: Использование облачных решений позволяет быстро масштабировать и адаптировать систему под изменяющиеся требования.
* Безопасность: Выбор технологий, поддерживающих шифрование и безопасную передачу данных, помогает соответствовать требованиям законодательства, обеспечивая защиту пользовательских данных.
* Совместимость: Выбор языков и фреймворков, которые хорошо работают друг с другом и имеют большое сообщество для получения поддержки.
* Удобство использования: Интуитивно понятные интерфейсы и механизмы аутентификации упрощают взаимодействие пользователей с системой.



Устройство пользователя представляет собой конечную точку, через которую пользователь взаимодействуют с системой с помощью веб-приложения. Сервер веб-приложения обрабатывает запросы пользователя и возвращает ему ответы по защищенному соединению. Запросы от пользователя передаются в API-сервер, взаимодействующий с различными системами и серверами, к примеру, с сервером аутентификации. Этот сервер отвечает за проверку идентификационных данных пользователя для получения доступа к ресурсам хранилища видеоданных или доступа к административным функциям только авторизованным пользователям с определенным IP-адресом. Сервер базы данных за хранение всех данных, связанных с пользователями и видео. Система логирования заносит в БД информацию обо всех совершаемых пользователем действиях, включая попытки входа, запросы на администрирование и доступ к видео, что помогает выявить возможные нарушения безопасности. Система обработки видео обеспечивает нужные функции обработки и конвертации видео.