

*Ершов Тимофей Андреевич, студент, Хакасский государственный университет им Н. Ф. Катанова, Россия, г. Абакан*

*Голубничий Артем Александрович, научный руководитель, старший преподаватель кафедры ПОВТиАС, Хакасский государственный университет им. Н. Ф. Катанова, Россия, г. Абакан*

## ОБЗОР СОВРЕМЕННЫХ ТЕХНОЛОГИЙ РАЗРАБОТКИ ДЛЯ WEB-ПРИЛОЖЕНИЙ

**Аннотация:** В данной статье представлена информация о инновационных подходах и инструментах, которые позволяют разработчикам создавать мощные и эффективные веб-приложения. Все эти технологии предоставляют разработчикам широкий спектр инструментов для создания современных и инновационных веб-приложений с учетом потребностей и требований пользователей.

**Ключевые слова:** Web-технологии, back-end, front-end, React, Django, Laravel, СУБД, PostgreSQL.

**Annotation:** The article presents information about innovative approaches and tools that enable developers to create powerful and efficient web applications. All these technologies provide developers with a wide range of tools for creating modern and innovative web applications that meet the needs and requirements of users.

**Keywords:** Web-technology, back-end, front-end, React, Django, Laravel, DBMS, PostgreSQL.

В мире визуальное содержание всегда являлось неотъемлемой частью общества и жизни каждого отдельного человека, но в разные времена оно имело совершенно разные способы распространения и передачи. Раньше для

популяризации и передачи визуального содержания служили книги, картины, живопись и даже наскальные рисунки. Разумеется, они были ограничены местом, в котором находятся. Сейчас же благодаря развитым технологиям визуальная информация может распространяться мгновенно и быть доступной из всех точек земного шара.

Платформы для распространения изображений носят названия фотохостинги. Они помогают не только хранить, распространять фотографии, но и популяризировать контент, зарабатывать творчеством, вести бизнес, помогать художникам и фотографам в развитии, повышать культурное развитие общества в целом, делиться идеями.

Раньше для создания библиотеки нужны были доски и инструменты, так и сегодня для разработки фотохостинга необходимы свои инструменты и материалы.

Первое, что стоит учесть, приложение будет работать через сеть, то есть будет доступно многим пользователям одновременно, оно будет хранить большие объемы информации, поскольку изображения требуют много памяти, даже современные алгоритмы сжатия без потерь с оптимизацией искусственного интеллекта, не решают проблему полностью и это необходимо учесть при выборе технологий для разработки. К примеру, уже в 2020 году Google заявил, что их пользователи еженедельно загружают 28 миллиардов фотографий в день, всего их сервис хранит более чем 4 триллиона фотографий. Поэтому даже для начинающего фотохостинга скорость, производительность и возможность расширения сервиса просто необходимы [1].

Разработку можно разделить на две большие части `back-end` и `front-end`. Первая часть касается серверной разработки, а именно обработки и хранения всей информации сайта, изображений, данных пользователей, безопасности и шифрования. Вторая часть отвечает за взаимодействие с пользователем, подачей контента и функционала, отображения интерфейса и создание анимаций.

Исходя из поставленных критериев для `front-end` или клиентской части разработки можно выбрать такие фреймворки как `React` или `Angular`. Данные

фреймворки помогают разрабатывать быстрые и динамичные современные интерфейсы, они созданы для клиентской части, у них очень схожий подход к разработке, и это идеальный вариант для фотохостинга, поскольку подобные приложения очень требовательны к UI. Их главная особенность – это динамическое обновление отдельных частей веб-страницы, или же по-другому: AJAX-запросы, что делает работу с получением и отправлением данных со страницы и на страницу значительно легче. К примеру, если взять такие фреймворки, как Laravel, Django или Spring, то по умолчанию при использовании blade-шаблонов для обновления некоторого содержания веб-страницы, необходимо полностью обновить все содержимое страницы, что вызывает неприятный опыт работы, а также увеличивает нагрузку на сервер в целом. Описанные выше React либо же Angular решают данную проблему по умолчанию [2].

Фреймворк React использует JSX, расширение для языка JavaScript, представляя все HTML-элементы интерфейса в роли компонентов, используя подход «разделяй и властвуй», упрощая разработку проекта, особенно удобно для командной разработки. Подобный подход используется во всех современных фреймворках для разработки динамического web-интерфейса.

Для back-end составляющей подойдут такие фреймворки, как Spring, Laravel, Django. При выборе фреймворка для back-end разработки важно учитывать скорость развития проекта, важность роли безопасности и хранения персональных данных, а также постепенного улучшения проекта.

Если проект только начинает свое развитие, не имеет клиентской базы, но должен быть развернут в минимальные сроки, то лучше сделать выбор в пользу Laravel или Django, данные фреймворки реализованы с помощью языков PHP и Python соответственно, позволяют не только быстро создать необходимый функционал, благодаря уже реализованным библиотекам, таким как ORM для быстрого развертывания таблиц баз данных при помощи классов, но и легко внедрить и обучить новых сотрудников, благодаря обширной документации и достаточно простым синтаксисом языков.

Фреймворк Spring написан на Java, по своей сути является наиболее сложным по сравнению с другими, поскольку имеет высокий порог вхождения, особенно если проект уже давно развивается, за счет своего предварительно сконфигурированного подхода, который задает определенный способ разработки, Spring прячет детали разработки от глаз разработчиков, что зачастую приводит к непониманию работы кода и соответственно ошибкам. Многие крупные сервисы используют именно Spring для серверной части, в пример можно привести Netflix, BlaBlaCar, Lanit-tercom. Но он имеет массу плюсов, огромное количество библиотек и удобного помощника для сборки проекта Maven или Gradle и считается одним из лучших сервисов в плане безопасности, имея отдельный модуль для авторизации и контроля доступа Spring security [3].

Для хранения всех изображений и данных пользователей и взаимодействием с back-end сервером также потребуется отдельный сервер для базы данных или СУБД, поскольку сложные сервисы должны иметь полный функциональный независимый сервер и возможность создания backup системы. Некоторые варианты зависят от бюджета, если разработка идет на коммерческой основе, то наилучшие варианты: MySQL, PostgreSQL и MariaDB.

PostgreSQL является бесплатным решением с открытым исходным кодом, один из главных плюсов данной СУБД – это отсутствие ограничений по объему базы данных, потому что при развитии проекта фотографий и прочей информации может стать критически много, и некоторые СУБД накладывают ограничение на объем базы или количеству записей. Также стоит отметить соответствие стандарту ACID, то есть максимальная безопасность и предсказуемость и возможность расширения и модификации [4].

Исходя из обзора всех вышеперечисленных технологий, для разработки фотохостинга лучше всего выбрать для клиентской части React, для серверной Django. Поскольку сервис только будет начинать свое развитие, то нет смысла выбирать более тяжелые фреймворки, Django позволит быстро и очень дешево развернуть сервис, для того чтобы опробовать его в действии, React предоставит динамичный и удобный интерфейс. PostgreSQL позволит создать надежное и

расширяемое хранилище для всех данных сайта.

### **Библиографический список:**

1. Google. Updating Google Photos' storage policy to build for the future. [Электронный ресурс] // URL: <https://blog.google/products/photos/storage-changes/> (Дата обращения: 17.08.2023).
2. w3schools. AJAX Introduction. [Электронный ресурсы] // URL: [https://www.w3schools.com/js/js\\_ajax\\_intro.asp](https://www.w3schools.com/js/js_ajax_intro.asp) (Дата обращения: 17.08.2023).
3. Emeritus. A Comprehensive Guide to the Top 7 Spring Boots Projects Ideas in 2023. [Электронный ресурс] // URL: <https://emeritus.org/blog/technology-spring-boot-projects-ideas/> (Дата обращения: 18.08.2023).
4. Skillfactory Media. PostgreSQL. [Электронный ресурс] // URL: <https://blog.skillfactory.ru/glossary/postgresql/> (Дата обращения: 18.08.2023).