Введение

Веб-разработка представляет собой важное направление в современной IT-индустрии, охватывающее создание веб-сайтов, приложений и серверных систем. В ходе практики были изучены главные языки программирования, технологии и принципы, применяемые в этой области. Данный отчет содержит анализ необходимых языков программирования, а также ключевые аспекты и подходы веб-разработки.

Анализ языков программирования для веб-разработки

Фронтенд-разработка (клиентская часть)

Фронтенд отвечает за визуальную часть веб-приложения, с которой работает пользователь. К основным языкам и технологиям относятся:

HTML (HyperText Markup Language) – это базовый язык разметки, нужный для определения структуры веб-страницы. Современные стандарты включают HTML5, который поддерживает мультимедиа и семантические теги.

CSS (Cascading Style Sheets) – это язык стилей, который определяет внешний вид страницы. Cовременные подходы включают CSS3, Sass/SCSS (препроцессоры) и Tailwind CSS (утилитарные классы).

JavaScript (JS) – основной язык программирования для обеспечения интерактивности на стороне клиента. Часто применяются фреймворки и библиотеки, такие как React.js (компонентный подход, виртуальный DOM), Vue.js (реактивность, простота интеграции) и Angular (полнофункциональный фреймворк от Google). Cовременные стандарты: ES6+ (стрелочные функции, промисы, модули).

Бэкенд-разработка (серверная часть)

Бэкенд обеспечивает логику приложения, работу с базами данных и API. Главные языки:

JavaScript (Node.js) дает возможность использовать JS на сервере. Популярные фреймворки: Express.js, NestJS, Fastify.

Python отличается простым синтаксисом и большим количеством библиотек. Используются фреймворки Django (полнофункциональный), Flask (микрофреймворк), FastAPI (для API).

PHP – часто применяемый язык для веба, используемый в WordPress, Laravel, Symfony.

Java – часто применяется для больших enterprise-решений (Spring Boot).

C# (.NET) – применяется для разработки на платформе Microsoft (ASP.NET Core).

Ruby – используется с фреймворком Ruby on Rails для быстрой разработки.

Go (Golang) отличается высокой производительностью и часто применяется в микросервисах.

Базы данных

SQL (реляционные): PostgreSQL, MySQL, SQLite.

NoSQL (нереляционные): MongoDB, Redis (кеширование).

Языки запросов: SQL, GraphQL (для API).

Принципы программирования в веб-разработке

Основные принципы:

DRY (Don’t Repeat Yourself) – помогает избежать дублирования кода.

SOLID (принципы ООП):

Single Responsibility (одна ответственность класса).

Open-Closed (открытость для расширения, закрытость для изменений).

Liskov Substitution (наследники должны заменять родителей).

Interface Segregation (узкоспециализированные интерфейсы).

Dependency Inversion (зависимость от абстракций).

RESTful API – архитектурный стиль для веб-сервисов (GET, POST, PUT, DELETE).

Популярные подходы при создании веб-приложений:

БЭМ (Блок, Элемент, Модификатор) – компонентный подход к веб-разработке, основанный на разделении интерфейса на независимые блоки.

Cоздан специалистами Яндекса и получил широкое распространение. В отличие от других CSS-концепций, БЭМ также работает с JavaScript, что упрощает код и рефакторинг.

БЭМ позволяет многократно использовать одни и те же фрагменты кода, в том числе на одном DOM-узле и назначение CSS-селекторов становится более понятным.

Основные понятия БЭМ:

Блок – это логически и функционально независимый компонент страницы, содержащий шаблоны, CSS-стили, скрипты JavaScript, документацию в формате XML или Markdown.

Элемент – это составная часть блока, которая не может существовать и использоваться вне его.

Модификатор – это сущность, которая определяет внешний вид, состояние и поведение элемента или блока.

MVC (Model-View-Controller) – разделение логики, интерфейса и данных (используется в Django, Ruby on Rails).

MVVM (Model-View-ViewModel) – реактивное программирование (Vue.js, Angular).

Singleton – гарантирует единственный экземпляр класса (например, подключение к БД).

Dependency Injection (DI) – внедрение зависимостей (Spring, NestJS).

Middleware – промежуточная обработка запросов (Express.js, Django).

Microservices – разделение приложения на независимые сервисы (Docker, Kubernetes).