

A large orange circle is positioned on the left side of the image, partially overlapping the text.

**backend  
development .**

# BACKEND DEV

01

Back-end разработка – это работа с серверной частью веб-приложений, включающая логику, базы данных, серверы и взаимодействие с клиентской частью (front-end)

02

Основные задачи включают:

- Написание серверного кода с использованием языков (например, Python, JavaScript, PHP).
- Управление базами данных (MySQL, PostgreSQL).
- Создание API для связи с фронтендом.
- Обеспечение безопасности и оптимизация работы серверов.

Back-end разработка отвечает за то, что происходит "за кулисами" веб-приложений, обеспечивая их функциональность.

# ОСНОВНЫЕ ХАРАКТЕРИСТИКИ

01

- Логика приложения: Обработка данных и выполнение бизнес-логики на сервере.
- Работа с базами данных: Хранение, изменение и управление данными через базы данных (SQL или NoSQL).

02

- API и интеграции: Обеспечение взаимодействия с фронтендом и внешними сервисами через API.
- Безопасность: Защита данных, управление доступом пользователей и предотвращение атак.

03

- Производительность: Оптимизация скорости обработки запросов и эффективности серверов.
- Масштабируемость: Возможность обработки большого числа пользователей и данных с увеличением нагрузки.

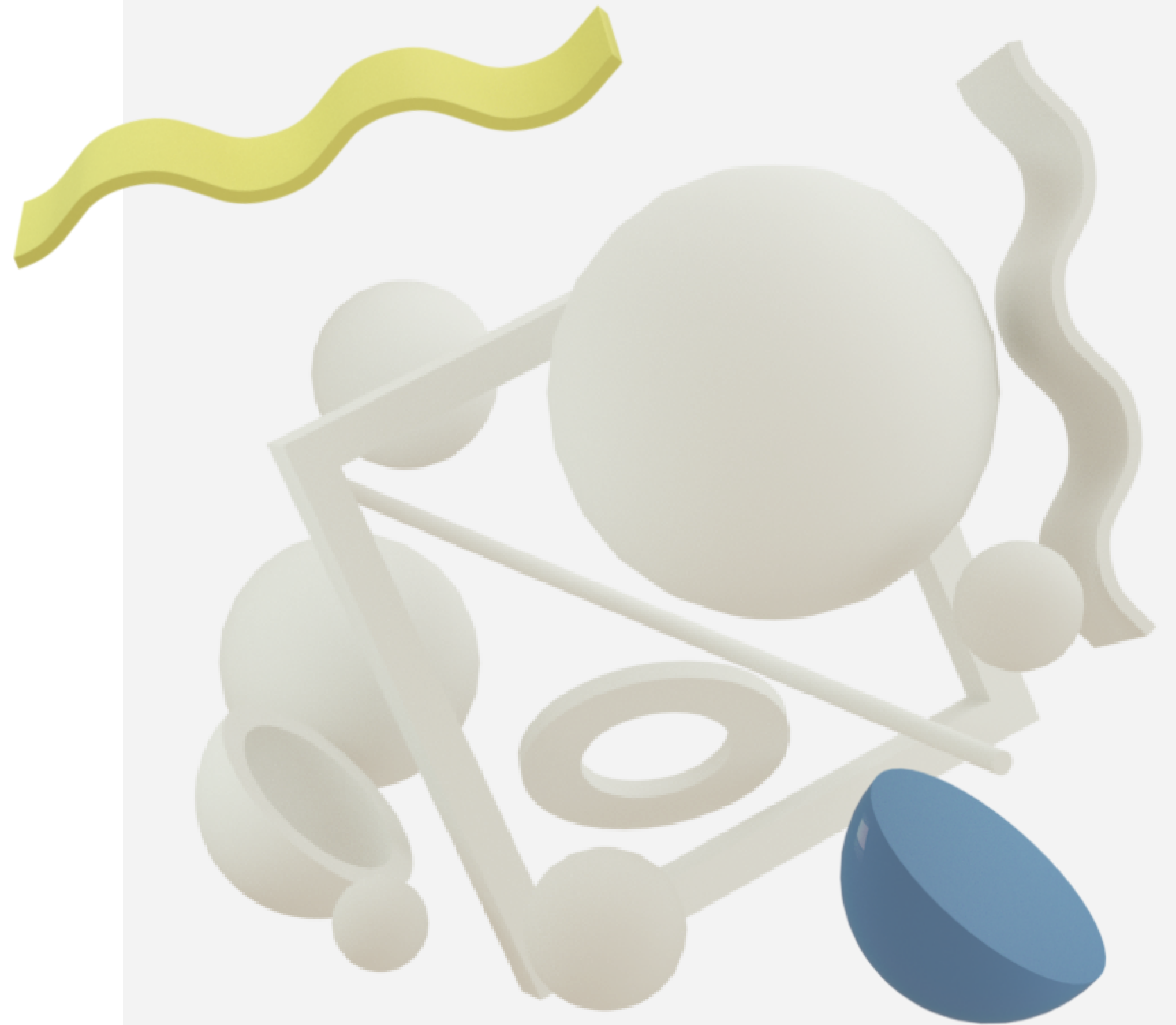
# ОСНОВНЫЕ ПРАВИЛА


Чистый и поддерживаемый код:

- Следуйте принципам KISS (Keep It Simple, Stupid) и DRY (Don't Repeat Yourself).
- Используйте комментирование кода и пишите код, который легко поддерживать.
- Безопасность:
- Защищайте данные с помощью шифрования (например, SSL/TLS).
- Используйте защиту от SQL-инъекций и атак типа XSS (Cross-Site Scripting).
- Реализуйте аутентификацию и авторизацию (OAuth, JWT и другие методы).
- Производительность:
- Оптимизируйте запросы к базе данных (используйте индексы, избегайте лишних запросов).
- Используйте кэширование для ускорения работы (Redis, Memcached).
- Следите за временем отклика сервера.
- Масштабируемость:
- Стройте архитектуру, которая легко масштабируется (используйте микросервисы, контейнеры, как Docker).
- Обеспечьте горизонтальное и вертикальное масштабирование в зависимости от роста нагрузки.


- Работа с базами данных:
- Проектируйте структуры данных, следуя нормализации, и применяйте индексы для ускорения работы.
- Резервируйте данные и реализуйте стратегии отката в случае ошибок.
- API-дизайн:
- Разрабатывайте RESTful или GraphQL API с ясными и стабильными интерфейсами.
- Документируйте API и обеспечьте его тестируемость.
- Обработка ошибок:
- Реализуйте централизованную обработку ошибок с четкими сообщениями для разработчиков.
- Логируйте критические события и ошибки для последующего анализа.
- Тестирование:
- Пишите юнит-тесты и интеграционные тесты, чтобы проверять корректность работы системы.
- Автоматизируйте тестирование, особенно при развертывании новых функций.

**frontend  
development**





Front-end разработка — это создание клиентской части веб-сайтов и приложений, то есть того, что видит и с чем взаимодействует пользователь. Включает работу с HTML, CSS и JavaScript для создания структуры, стилей и интерактивных элементов. Front-end разработчик отвечает за удобство интерфейса, адаптивность дизайна и связь с сервером для отображения данных.



# ОСНОВНЫЕ ПРАВИЛА

- Чистый и структурированный код:
- Следуйте принципам DRY (Don't Repeat Yourself) и KISS (Keep It Simple, Stupid).
- Используйте понятные имена переменных и классов, а также структурируйте HTML, CSS и JavaScript для лучшей читаемости.
- Адаптивный дизайн:
- Убедитесь, что сайт корректно отображается на различных устройствах и экранах (телефоны, планшеты, компьютеры).
- Используйте медиазапросы и гибкие макеты для адаптации интерфейса.

- Оптимизация производительности:
- Минимизируйте CSS и JavaScript.
- Оптимизируйте изображения и загружайте их только тогда, когда они необходимы (ленивая загрузка).
- Используйте кэширование для ускорения загрузки страниц.
- Совместимость между браузерами:
- Обеспечьте корректную работу сайта во всех современных браузерах.
- Тестируйте и решайте проблемы, связанные с различиями в рендеринге.
- 

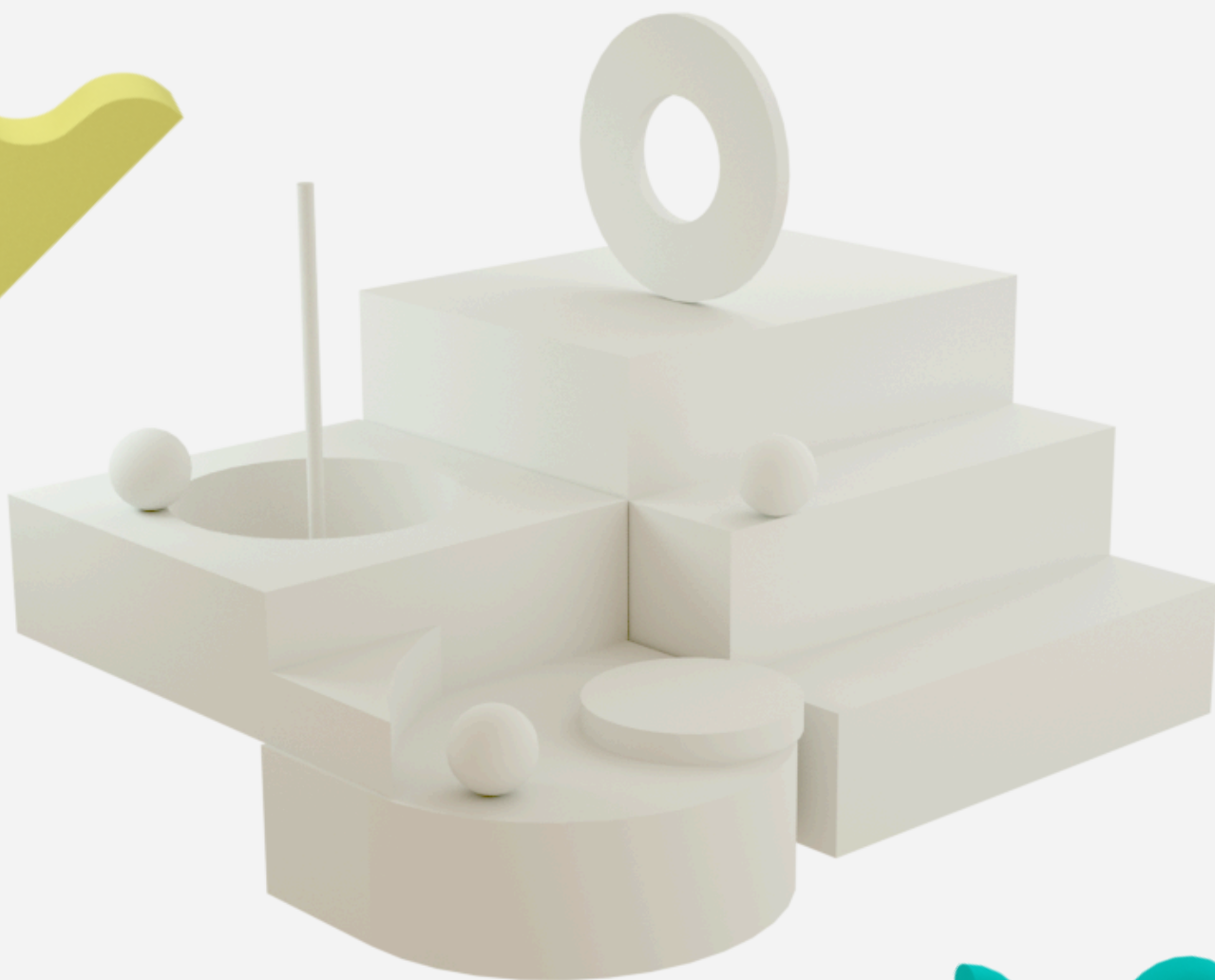
- Удобство пользователя (UX):
- Разрабатывайте интуитивные и понятные интерфейсы.
- Учитывайте удобство навигации, быстроту отклика элементов и четкость структуры контента.
- Доступность (Accessibility):
- Обеспечьте доступ к сайту для людей с ограниченными возможностями (например, используйте семантический HTML, поддержку экранных читалок).
- Добавляйте альтернативные тексты к изображениям и делайте интерфейсы управляемыми с клавиатуры.

- Интерактивность и отзывчивость:
- Используйте JavaScript для реализации динамических элементов (анимации, валидации форм, динамические обновления контента).
- Обеспечьте быстрое реагирование элементов интерфейса на действия пользователя.
- Документирование кода:
- Комментируйте сложные участки кода для облегчения работы в команде и будущих изменений.





# КАК ОН ИСПОЛЬЗУЕТСЯ



- Создание структуры страниц: С помощью HTML (языка разметки) разработчики создают основу страницы, определяя, какие элементы будут присутствовать (заголовки, абзацы, кнопки, изображения и т.д.).
  - Оформление и стилизация: Используя CSS, разработчики добавляют стили, чтобы придать веб-странице внешний вид — цвета, шрифты, размеры, расположение элементов и т.д.
  - Интерактивность: При помощи JavaScript добавляется динамическое поведение. Например, пользователи могут нажимать на кнопки, и контент страницы будет изменяться без перезагрузки.
  - Связь с сервером: Front-end взаимодействует с серверной частью (back-end) через API. Например, пользователь отправляет форму на сайте, и данные отправляются на сервер для обработки или хранения, а затем результат отображается обратно на странице.
1. Адаптивный дизайн: Front-end разработка позволяет сайтам корректно отображаться на различных устройствах (компьютеры, мобильные телефоны, планшеты) с помощью медиазапросов и гибких макетов, создавая адаптивные интерфейсы.
  2. Оптимизация: Front-end разработка используется для повышения производительности сайта: уменьшение времени загрузки страницы, оптимизация изображений, минимизация CSS и JavaScript.





# WEB DEVELOPMENT





Web-разработка — это процесс создания, разработки и поддержания веб-сайтов и веб-приложений. Он включает как front-end (клиентская часть, с которой взаимодействует пользователь), так и back-end (серверная часть, обрабатывающая логику и данные).

Основные аспекты web-разработки:

- Front-end разработка: Создание интерфейса с использованием HTML, CSS и JavaScript для обеспечения удобного взаимодействия с пользователями.
- Back-end разработка: Обработка данных и логики на сервере с помощью языков программирования (например, Python, PHP, Java) и управления базами данных.
- Дизайн: Создание визуального стиля сайта и его структуры.
- Оптимизация: Улучшение производительности и доступности веб-сайтов.

Web-разработка включает все этапы — от планирования и дизайна до разработки, тестирования и развертывания веб-приложений.





## ДЛЯ ЧЕГО ОНА НУЖНА

Предоставление  
информации

Онлайн-бизнес

Коммуникация

Автоматизация процессов

Удобство доступа

Обучение и развитие

Интерактивность





## КАК ИСПОЛЬЗУЕТСЯ?

- Web-разработка используется для создания и поддержания веб-сайтов и веб-приложений. Вот как она применяется в кратце:
- Создание сайтов:
- Статические (HTML, CSS) и динамические (серверные языки, базы данных) веб-страницы.
- Разработка веб-приложений:
- Использование JavaScript и фреймворков (React, Angular) для создания интерактивных интерфейсов.
- Оптимизация производительности:
- Улучшение загрузки ресурсов, применение кэширования и ленивой загрузки.
- Интеграция с API:
- Взаимодействие с внешними API для получения данных и создания собственных API.
- Тестирование и отладка:
- Проверка совместимости и выявление ошибок на различных устройствах и браузерах.
- Поддержка и обновление:
- Регулярное обновление контента и технологий для обеспечения безопасности и актуальности.
- Создание пользовательского опыта (UX):
- Оптимизация интерфейса на основе анализа поведения пользователей и обратной связи.
- 





# THANK YOU

Илияр и Санжар

