

# Основные компоненты архитектуры информационной системы и их взаимодействие



# Что такое архитектура информационной системы?

Архитектура информационной системы (ИС) — если кратко, то это схема, которая показывает, из каких частей состоит система и как они работают вместе.

## Принципиальная организация

Набор компонентов, их взаимосвязи, связи с внешней средой и правила управления.

## Достижение целей

Структура, обеспечивающая достижение как бизнес-целей, так и технических требований.

## Последствия плохой архитектуры

Приводит к сбоям, высоким операционным затратам и ухудшению пользовательского опыта.



# Ключевые компоненты информационной системы



## Пользовательский интерфейс (Frontend)

отвечает за то, чтобы человеку было удобно пользоваться системой — показывает информацию, принимает команды (клики, ввод текста и т. п.) и делает всё понятным и удобным.



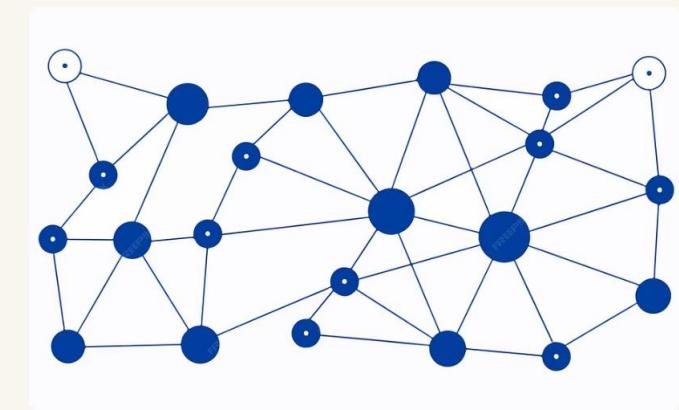
## Серверная часть (Backend)

Это невидимая для пользователя часть системы, которая выполняет всю «рабочую магию» — обрабатывает данные и делает так, чтобы система «работала по правилам».



## Хранилище данных (Базы данных)

Это организованное «место хранения» информации, где всё на своих местах, ничего не теряется и можно быстро найти нужное.



## Внешние системы и сервисы

Это «розетка/переходник» для вашей системы — вставляете плагин (платёжку, доставку, уведомления) — и сразу получаете готовую функцию.

# Архитектурные стили: монолит и микросервисы

## Монолитная архитектура

Единое, неделимое приложение, в котором все функциональные модули (UI, бизнес-логика, данные) тесно связаны.

- **Преимущества:** Простота начальной разработки, развертывания и тестирования.
- **Недостатки:** Трудно масштабировать отдельные части, сложная поддержка при росте системы, медленные циклы выпуска обновлений.
- **Примеры:** Shopify, Github, Stack Overflow и т.д



## Микросервисная архитектура

Набор независимых, небольших сервисов, каждый из которых отвечает за отдельную бизнес-функцию и взаимодействует с другими через API.

- **Преимущества:** Высокая гибкость, независимое масштабирование сервисов, быстрые и частые обновления, устойчивость к сбоям.
- **Недостатки:** Сложность интеграции, мониторинга и управления инфраструктурой, распределённые транзакции.
- **Примеры:** Amazon, Wildberries, Авito и много других

The image shows the logo for Wildberries, featuring the brand name in a large, bold, purple sans-serif font.

# Принципы взаимодействия компонентов и интеграции

## Слабая связанность

Компоненты системы должны работать почти независимо друг от друга.

## Специализация обязанностей

Каждый кусочек системы должен делать только свою конкретную работу — и ничего лишнего.

## Стандартизованные протоколы

Это общие «правила общения» для компьютерных систем — чтобы они могли понимать друг друга, как люди говорят на одном языке.

## Безопасность и надёжность

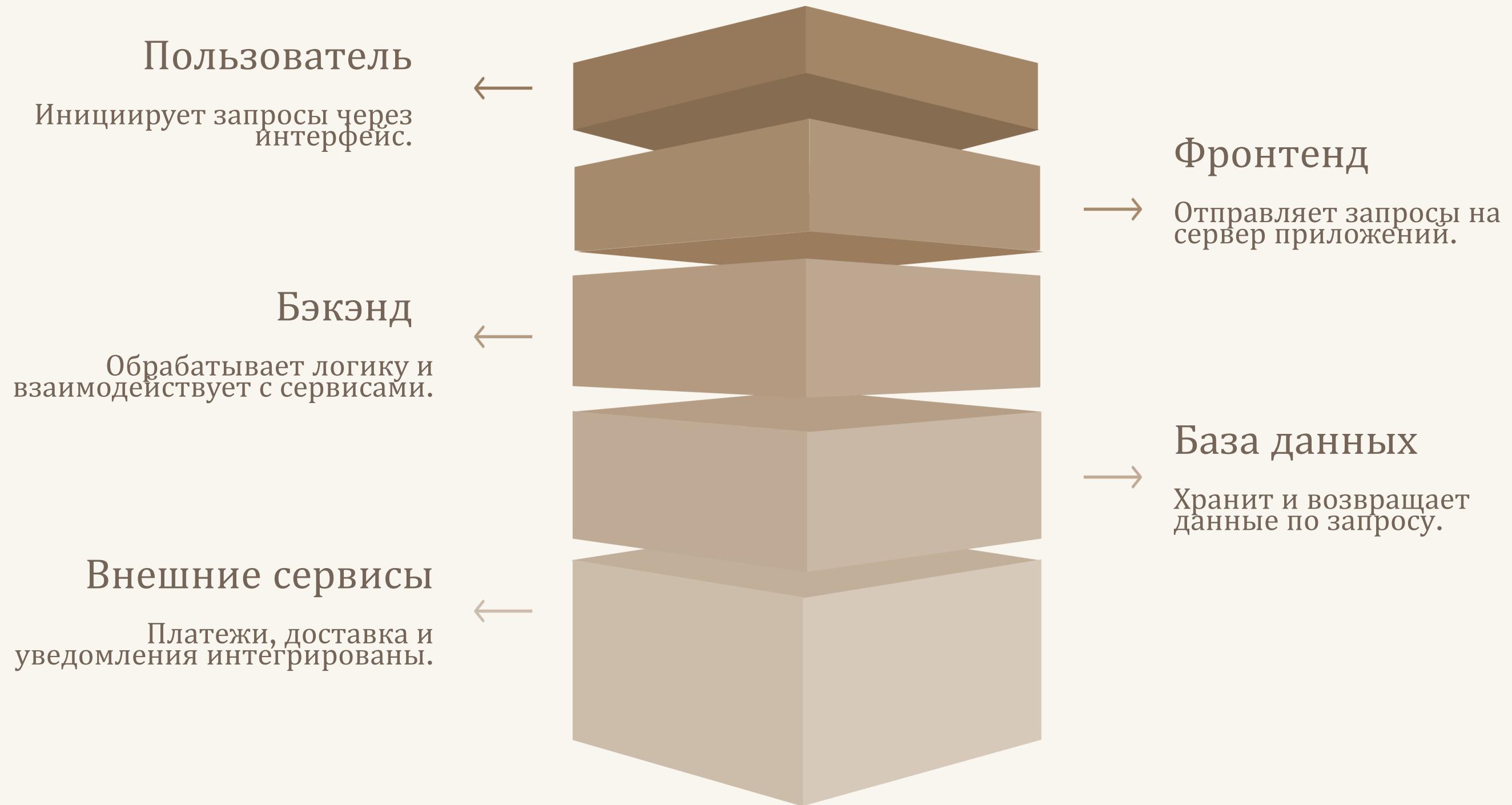
Это комплекс мер, чтобы система была безопасной и работала без перебоев

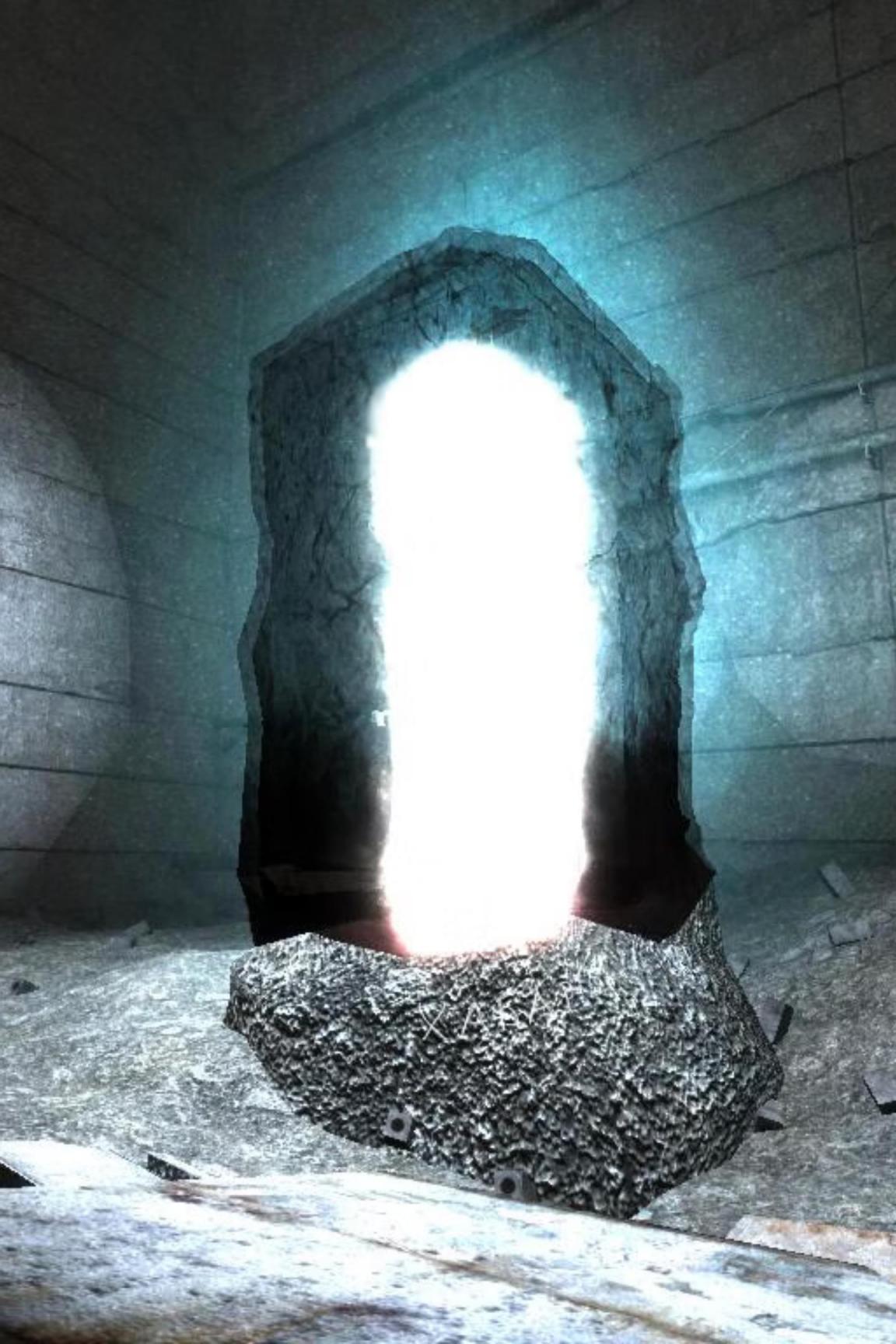


# Уровни архитектурного проектирования ИС



# Визуализация архитектуры: диаграмма взаимодействия компонентов





## Заключение

Архитектура — это фундамент, который позволяет создавать эффективные, конкурентоспособные и устойчивые информационные системы, минимизируя риски и затраты на их поддержку и развитие.