

1. Понятие информационной системы

Как говорить:

> Информационная система — это совокупность программного и аппаратного обеспечения, данных, пользователей и бизнес-процессов, предназначенная для сбора, хранения, обработки и передачи информации с целью поддержки деятельности организации.

2. Цели проектирования ИС

> Основная цель проектирования ИС — автоматизация и оптимизация бизнес-процессов, повышение эффективности работы и качества управленческих решений.

3. Жизненный цикл ИС

> Жизненный цикл ИС включает этапы: планирование, анализ требований, проектирование, разработку, тестирование, внедрение, эксплуатацию и вывод системы из эксплуатации.

4. Стадии жизненного цикла

> Основные стадии — анализ, проектирование, реализация, тестирование, внедрение и сопровождение. Каждая стадия имеет свои результаты и документацию.

5. Каскадная модель (Waterfall)

> Waterfall — это линейная модель разработки, где каждый этап начинается только после завершения предыдущего. Подходит для проектов с чётко зафиксированными требованиями.

6. Итеративная и инкрементальная модели

> Итеративная модель предполагает повторяющиеся циклы улучшения системы, а инкрементальная — поэтапную реализацию функционала частями.

7. Agile и Scrum

> Agile — это гибкий подход к разработке, ориентированный на быстрые итерации и изменения требований. Scrum — одна из реализаций Agile со спринтами, ролями и регулярными встречами.

8. Сравнение Waterfall и Agile

> Waterfall подходит для стабильных требований и формальных проектов, Agile — для динамичных проектов, где требования часто меняются.

9. Роль системного аналитика

> Системный аналитик занимается сбором и анализом требований, моделированием системы и взаимодействием между заказчиком и разработчиками.

10. Сбор требований

> Требования собираются с помощью интервью, опросов, наблюдений и анализа документов, после чего они уточняются и приоритизируются.

11. Виды требований

> Различают функциональные требования — что система должна делать, и нефункциональные — как она должна это делать, например по скорости или безопасности.

12. Методы выявления требований

> Используются интервью, фокус-группы, прототипирование, мозговой штурм и анализ существующих систем.

13. Документирование требований

> Требования оформляются в виде SRS, пользовательских историй или спецификаций с указанием приоритетов и ограничений.

14. Модель предметной области

> Модель предметной области описывает основные сущности, их атрибуты и связи в рамках бизнес-процессов организации.

15. Архитектура ИС

> Архитектура ИС определяет структуру системы, компоненты и способы их взаимодействия.

16. Виды архитектур

> Основные виды архитектур — клиент-серверная, многоуровневая и сервис-ориентированная архитектура.

17. Модульное проектирование

> Модульное проектирование основано на принципах слабой связанности и высокой сплочённости модулей.

18. UML

> UML — это унифицированный язык моделирования, используемый для визуального описания структуры и поведения системы.

19. Виды UML-диаграмм

> UML-диаграммы делятся на структурные и поведенческие, например диаграммы классов и диаграммы вариантов использования.

20. Диаграмма вариантов использования (Use Case)

> Use Case диаграмма показывает взаимодействие акторов с системой и основные сценарии использования.

21. Диаграмма классов

> Диаграмма классов отражает структуру системы: классы, их атрибуты, методы и отношения между ними.

22. Диаграмма последовательностей

> Диаграмма последовательностей показывает обмен сообщениями между объектами во времени.

23. Проектирование базы данных

> Проектирование БД включает определение таблиц, ключей, связей и индексов для эффективного хранения данных.

24. Модели данных

> Различают концептуальную, логическую и физическую модели данных.

25. ER-диаграмма

> ER-диаграмма используется для отображения сущностей, их атрибутов и связей между ними.

26. Нормализация

> Нормализация — это процесс приведения базы данных к нормальным формам для устранения избыточности и аномалий.

27. Проектирование пользовательского интерфейса

> UI-проектирование направлено на удобство, понятность и эффективность взаимодействия пользователя с системой.

28. Эргономика интерфейса

> Эргономика предполагает логичную навигацию, минимализм и соблюдение стандартов доступности.

29. Прототипирование

> Прототипирование позволяет визуально представить будущую систему и проверить требования на раннем этапе.

30. CASE-средства

> CASE-средства — это инструменты для автоматизации проектирования и документирования ИС.

31. Техническое проектирование

> Техническое проектирование включает детальное описание архитектуры, интерфейсов и алгоритмов системы.

32. Техническое задание

> Техническое задание — основной документ, который фиксирует требования, сроки и ограничения проекта и утверждается заказчиком.

33. Проектная документация

> Проектная документация включает спецификации, описания API, инструкции пользователей и администраторов.

34. Обеспечение качества

> Качество ИС обеспечивается с помощью стандартов, тестирования, код-ревью и контроля метрик.

35. Тестирование

> Тестирование — это процесс проверки соответствия системы требованиям.

36. Виды тестирования

> Существуют модульное, интеграционное, системное, нагрузочное и приёмочное тестирование.

37. Безопасность ИС

> Безопасность ИС включает аутентификацию, авторизацию, шифрование и защиту от несанкционированного доступа.

38. Защита данных

> Защита данных обеспечивается с помощью разграничения прав доступа, шифрования и журналирования.

39. Надёжность ИС

> Надёжность достигается за счёт резервного копирования, отказоустойчивости и репликации.

40. Производительность

> Производительность системы повышается за счёт кэширования и масштабирования.

41. Интеграция систем

> Интеграция позволяет обмениваться данными между системами через очереди сообщений или сервисы.

42. API и SOA

> API обеспечивает взаимодействие между компонентами, а SOA основана на использовании независимых сервисов.

43. Сопровождение ИС

> Сопровождение включает исправление ошибок, обновления и мониторинг системы.

44. Рейнжиниринг

> Рейнжиниринг — это переработка существующей системы для повышения её эффективности.

45. Управление изменениями

> Управление изменениями позволяет контролировать внесение правок без нарушения стабильности системы.

46. Риски проекта

> Управление рисками включает их выявление, анализ и минимизацию последствий.

47. Оценка эффективности

> Эффективность ИС оценивается с помощью показателей ROI, NPV и срока окупаемости.

48. Оценка затрат

> Для оценки затрат применяются модели СОСМО и метод функциональных точек.

49. Стандарты

> В проектировании ИС используются международные и национальные стандарты, такие как ISO и ГОСТ.

50. Современные тенденции

> Современные тенденции включают облачные технологии, DevOps, микросервисы и искусственный интеллект.

ИС — Информационная система

Как говорить:

> ИС — это информационная система, то есть совокупность программ, оборудования, данных, пользователей и процессов, которая предназначена для обработки информации и поддержки деятельности организации.

ПО — Программное обеспечение

> ПО — это совокупность программ, которые обеспечивают работу информационной системы и выполнение её функций.

ЖЦ ИС — Жизненный цикл информационной системы

> ЖЦ ИС — это последовательность этапов, через которые проходит информационная система от идеи и анализа требований до эксплуатации и вывода из использования.

SDLC — *System Development Life Cycle*

> SDLC — жизненный цикл разработки системы, включающий анализ, проектирование, разработку, тестирование, внедрение и сопровождение.

(Можно сказать: *по сути то же самое, что жизненный цикл ИС*.)

Waterfall — каскадная модель

> Waterfall — это модель разработки, при которой все этапы выполняются строго последовательно без возврата к предыдущим.

Agile — гибкая методология разработки

> Agile — это подход к разработке ИС, основанный на коротких итерациях, постоянной обратной связи и готовности к изменениям требований.

Scrum — фреймворк Agile

> Scrum — это методология в рамках Agile, которая использует спринты, роли и регулярные встречи для управления разработкой.

UML — *Unified Modeling Language*

> UML — унифицированный язык моделирования, который используется для визуального описания структуры и поведения информационной системы.

Use Case — вариант использования

> Use Case — это описание сценария взаимодействия пользователя или внешней системы с информационной системой.

ER — *Entity–Relationship*

> ER — модель «сущность–связь», применяемая для концептуального проектирования базы данных.

ERD — *Entity–Relationship Diagram*

> ERD — диаграмма «сущность–связь», которая графически отображает структуру данных предметной области.

БД — База данных

> База данных — это организованная совокупность данных, предназначенная для хранения и обработки информации.

СУБД — Система управления базами данных

> СУБД — программное средство, которое обеспечивает создание, хранение, изменение и доступ к данным в базе данных.

SQL — *Structured Query Language*

> SQL — структурированный язык запросов, используемый для работы с реляционными базами данных.

API — *Application Programming Interface*

> API — это интерфейс программного взаимодействия, который позволяет разным системам или модулям обмениваться данными и функциями.

REST — *Representational State Transfer*

> REST — архитектурный стиль построения API, основанный на использовании HTTP-методов и принципе работы с ресурсами.

SOA — *Service-Oriented Architecture*

> SOA — сервис-ориентированная архитектура, при которой система состоит из независимых сервисов, взаимодействующих между собой.

UI — *User Interface*

> UI — пользовательский интерфейс, то есть средства взаимодействия пользователя с информационной системой.

UX — *User Experience*

> UX — пользовательский опыт, который отражает удобство и эффективность работы пользователя с системой.

CASE — *Computer-Aided Software Engineering*

> CASE — это инструменты автоматизации процессов проектирования, моделирования и документирования информационных систем.

SRS — *Software Requirements Specification*

> SRS — спецификация требований к программному обеспечению, в которой формально описываются функциональные и нефункциональные требования.

ТЗ — Техническое задание

> ТЗ — официальный документ, который определяет требования к системе, сроки выполнения и ответственность сторон.

ISO — *International Organization for Standardization*

> ISO — международная организация по стандартизации, разрабатывающая стандарты в области качества и информационных технологий.

IEEE — *Institute of Electrical and Electronics Engineers*

> IEEE — международная организация, разрабатывающая стандарты в области программной инженерии.

ГОСТ — Государственный стандарт

> ГОСТ — национальный стандарт, регламентирующий требования к разработке и документации ИС.

RBAC — *Role-Based Access Control*

> RBAC — модель управления доступом, при которой права назначаются ролям, а не конкретным пользователям.

AES — *Advanced Encryption Standard*

> AES — симметричный алгоритм шифрования, используемый для защиты данных.

OAuth — *Open Authorization*

> OAuth — протокол авторизации, позволяющий безопасно предоставлять доступ к ресурсам без передачи пароля.

ROI — *Return on Investment*

> ROI — показатель эффективности инвестиций, показывающий окупаемость внедрения информационной системы.

NPV — *Net Present Value*

> NPV — чистая приведённая стоимость проекта, используемая для оценки экономической эффективности.

COCOMO — *Constructive Cost Model*

> COCOMO — модель оценки стоимости и трудоёмкости разработки программного обеспечения.

DevOps — *Development + Operations*

> DevOps — подход, объединяющий разработку и эксплуатацию для ускорения выпуска и повышения качества ИС.

Cloud — облачные технологии

> Cloud — модель предоставления вычислительных ресурсов через интернет.