

ИТ. 3 семестр

Do not die

25 февраля 2026 г.

Содержание

1	Информационные технологии. Основные понятия (!!!)	4
2	Информационные технологии. Основные направления развития	5
3	Разработка ПО как одно из направлений информационных технологий (!!!)	6
4	Сопровождение ПО как одно из направлений информационных технологий	8
5	Технологии сетевого взаимодействия как одно из направлений информационных технологий	9
6	Интернет-технологии как одно из направлений информационных технологий	10
7	Веб-технологии как направления информационных технологий. Веб-разработка (!!!)	11
8	Технологии работы с медиаконтентом как направление информационных технологий	12
9	Базы данных, базы знаний как средства поддержки информационных технологий	13
10	Искусственный интеллект и машинное обучение как одно из направлений информационных технологий	14
11	Информационная безопасность как одно из направлений информационных технологий	15
12	Крупные комплексные решения как одно из направлений информационных технологий	16

13 Программирование. Основные понятия и процессы	17
14 Автоматизированные системы и основные виды обеспечения их функционирования	18
15 Основные виды обеспечения функционирования и их взаимосвязь (!!!)	19
16 Программное обеспечение автоматизированных систем как комплекс программ. Особенности создания комплексов программ	20
17 Основная задача профессионального программирования и понятие качества ПО (!!!)	21
18 Понятие качества ПО и его основные характеристики (!!!)	22
19 Функциональность как характеристика качественного ПО	23
20 Надежность как характеристика качественного ПО	24
21 Удобство как характеристика качественного ПО	25
22 Сопровождаемость как характеристика качественного ПО	26
23 Эффективность как характеристика качественного ПО	27
24 Принципы выбора характеристик качества ПО. Понятия «достаточно хорошего» программного продукта (!!!)	28
25 Тестирование и отладка ПО как взаимосвязанные процессы. Основные принципы тестирования	30
26 Тестирование и отладка ПО как взаимосвязанные процессы. Основные принципы отладки	32
27 Основные стратегии тестирования. Понятие о критериях тестирования (!!!)	33
28 Применение стратегии черного ящика при тестировании. Возможные критерии тестирования (!!!)	34
29 Применение стратегии белого ящика при тестировании. Возможные критерии тестирования (!!!)	36
30 Стратегия черного ящика и нагрузочное тестирование	38
31 Стратегия черного ящика. Модульное, интеграционное и регрессионное тестирование	39

1 Информационные технологии. Основные понятия (!!!)¹

Информационные технологии

Опр Информационные технологии — это методы и процессы:

1. поиска (поисковые системы),
2. сбора (получения; датчики температур),
3. хранения (база данных, знаний),
4. преобразования (обработки; алгоритм сортировки),
5. передачи (СМИ)

информации и способы осуществления таких методов и процессов.

Основные понятия

Информация

Опр Информация — это сведения (сообщения, данные), не зависящие от формы их представления:

- Сообщение — это конкретная форма передачи (предоставления) информации от источника к получателю (текст в чате, звук сирены, сигнал светофора).
- Данные — это сведение, приведенное к формализованному виду, пригодному для хранения и обработки (JSON-файл: формат представления данных в виде набора пар ключ-значение).

Доступ к информации

Опр Доступ — это возможность получения и использования информации:

1. легальный (доступ к собственному аккаунту),
2. нелегальный (доступ к чужому аккаунту в результате взлома).

Передача информации: распространение и предоставление

Опр Предоставление — передача информации определенному кругу лиц (письмо или имейл лицу).

Опр Распространение — передача информации неопределенному кругу лиц (пост в соцети, афиша).

¹Часто обсуждаемые вопросы с Барчевым, т.е. он может их задать, даже если этого вопроса нет в билете

2 Информационные технологии. Основные направления развития

Информационные технологии

См. вопрос 1

Основные направления

- ☐ Разработка ПО,
- ☐ Сопровождение ПО,
- ☐ Сетевые технологии,
- ☐ Интернет-технологии,
- ☐ Веб-технологии,
- ☐ Технологии работы с медиаконтентом,
- ☐ AI и ML,
- ☐ Информационная безопасность,
- ☐ Крупные комплексные решения.

3 Разработка ПО как одно из направлений информационных технологий (!!!)

Разработка ПО

Опр Разработка ПО — систематический процесс создания программного продукта — совокупности компьютерных программ для решения конкретных задач.

Виды ПО

Классификация по назначению:

- ☐ СПО (системное) — ПО, обеспечивающее функционирование вычислительной машины (ОС, драйверы, утилиты).
- ☐ ППО (прикладное), или приложение — ПО, предназначенное для выполнения конкретных задач пользователя (текстовые редакторы, браузеры, игры).

Классификация по способу распространения:

- ☐ Коробочное ПО — программный продукт, предназначенный для широкого круга пользователей (Paint, Microsoft Office).
- ☐ Заказное ПО — ПО, разработанное специально под требования заказчика.

Проектирование ПО

Опр Проектирование ПО — это процесс разработки архитектуры программной системы до начала ее реализации.

Этапы разработки ПО

1. Проектирование архитектуры.
2. Программная реализация архитектурных решений.
3. Вопрос обеспечения качества разработки (тестирование, отладка, анализ).
4. Сопровождение.
5. Организация и автоматизация процесса разработки (распределение ролей и постановка задачи — как, автотесты, автопроверка кода, автосборка и автодеплой).

Дополнительно²

Опр ОС (операционная система) — СПО, управляющее ресурсами компьютера (оперативной памятью, файловой системой) и обеспечивающее взаимодействие пользователя и прикладных программ с аппаратной частью.

Опр Драйвер — ПО, обеспечивающее взаимодействие ОС с конкретным аппаратным устройством (клавиатура, мышь, видеокарта, принтер).

Опр Утилита — вспомогательная программа для выполнения типовых служебных задач.

Опр Вычислительная система (в т.ч. ЭВМ — на электронных компонентах: транзисторы, микросхемы) — совокупность аппаратно-программных средств, предназначенных для автоматической обработки информации.

Опр Архитектура ПО — структура программной системы и принципы взаимодействия ее компонентов (монолит, клиент-сервер).

²Под этим заголовком содержатся термины, про которых нет отдельного вопроса в билете, однако их могут вполне спросить и без того, поэтому стоит аккуратно подбирать слова, чтобы у Барчева не было возможности спрашивать тебя про каждое слово. Термины беру не по прогнозу, а из опыта прошедшего зачета, т.е. это действительно спрашивали

4 Сопровождение ПО как одно из направлений информационных технологий

Сопровождение

Опр Сопровождение — совокупность работ, выполняемых после ввода программной системы (ПО и аппаратное обеспечение) в эксплуатацию с целью обеспечения ее стабильной работы и развития.

Виды сопровождения

- ☐ **Корректирующее** (исправление выявленных ошибок, привлечших за собой дефект, из-за которого у пользователя случился баг).
- ☐ **Адаптивное** (модификация системы в связи с изменением внешней среды: законодательство, обновленное оборудование).
- ☐ **Совершенствующее, или эволюционное** (добавление новых функций и улучшение существующих – обновление).
- ☐ **Профилактическое, или предупредительное** (работы по предотвращению потенциальных, но нетекущих проблем, по уменьшению технического долга: переписывание легаси-модулей, рефакторинг сложного кода, обновление устаревших или небезопасных библиотек).

Дополнительно

Опр Рефакторинг — изменение программного кода без изменения его поведения в профилактических целях (качество, производительность).

Опр Технический долг — накопленные архитектурные и кодовые решения, принятые в ущерб качеству ради ускорения разработки, которые впоследствии требуют дополнительных затрат на исправление.

Опр SLA (Service Level Agreement) — соглашение об уровне обслуживания программной системы, определяющее параметры его качества и срока поддержки.

Опр Легаси — эксплуатируемый программный код, созданный на устаревших технологиях или с архитектурными ограничениями, затрудняющими ее сопровождение.

5 Технологии сетевого взаимодействия как одно из направлений информационных технологий

Технологии сетевого взаимодействия

Опр Технологии сетевого взаимодействия (сетевые технологии) — это методы и средства организации обмена данными между устройствами по сети.

Направления деятельности

- Организация и администрирование сетей: создание, настройка и поддержка работы компьютерных сетей (настройка локальной сети в организациях).
- Разработка приложений распределенных, т.е. таких, что все его компоненты работают независимо друг от друга на разных компьютерах как единое целое, что достигается взаимодействием по сети (клиент, сервер и база данных на 3 разных компьютерах).

Дополнительно

Опр Компьютерная сеть — совокупность взаимосвязанных компьютеров и устройств, обеспечивающих обмен данными между ними (LAN и WLAN — локальные проводная и беспроводная: Ethernet и Wi-Fi).

Опр Интернет — глобальная система взаимосвязанных компьютерных сетей, использующих единый набор протоколов передачи данных.

Опр Протокол — набор правил обмена данными между устройствами в сети.

6 Интернет-технологии как одно из направлений информационных технологий

Интернет-технологии

Опр Интернет-технологии — это методы и средства организации передачи данных и функционирования информационных систем в сети Интернет.

Направления деятельности

- ☐ Веб-разработка.
- ☐ Поддержка информационных ресурсов: администрирование сайтов и серверов, обновление контента, обеспечение безопасности и доступности ресурсов.
- ☐ Облачные технологии — предоставление вычислительных ресурсов (на разворачивание – установку, настройку и запуск – сервера, поднятие – создание и запуск – базы данных) через сеть Интернет.
- ☐ Медиаконтент.
- ☐ Поисковые технологии и оптимизация поиска ресурсов — разработка алгоритмов поиска, индексации и ранжирования (сортировки по определенным критериям) информации.

Дополнительно

Опр Интернет — глобальная система взаимосвязанных компьютерных сетей, использующих единый набор протоколов передачи данных.

Опр Индексирование информации — процесс сканирования, анализа и добавления данных со страниц сайта в базу данных поисковых систем (Гугл, Яндекс).

Опр DNS (Domain Name System) — система доменных имен, обеспечивающая соответствие между доменным именем и IP-адресом сервера.

Опр Домен (Доменное имя) — это символьное имя, используемое для идентификации ресурса в сети Интернет.

Опр IP-адрес — это уникальный числовой идентификатор устройства в сети, использующий протокол IP, предназначенный для адресации и доставки данных.

7 Веб-технологии как направления информационных технологий. Веб-разработка (!!!)

Веб-технологии

Опр Веб-технологии — это технологии создания и функционирования гипертекстовых информационных ресурсов (предназначенные для хранения и предоставления информации, статический: новостной сайт, электронная библиотека) и веб-приложений (обеспечивающие интерактивное взаимодействие пользователя с сервером, динамический: почтовый сервис, онлайн-магазин) в среде WWW.

Веб-разработка

Опр Веб-разработка — это процесс создания веб-сайтов и веб-приложений. Разделяется на:

1. Клиентскую разработку (Front-end), отвечающую за внешнюю, клиентскую часть веб-сайтов и приложений: отображение информации и взаимодействие с ней. Используются HTML, CSS, JavaScript.
2. Серверную разработку (Back-end), отвечающую за обработку запросов клиента в рамках бизнес-логики, безопасности, хранения данных в базе данных и т.д. Используются PHP, Python, Java, Go.

Дополнительно

Опр WWW (World Wide Web) — это распределенная система гипертекстовых документов, ресурсов, доступных через сеть Интернет с использованием HTTP.

Опр HTTP (HyperText Transfer Protocol) — это протокол, использующийся для передачи данных в сети Интернет.

Опр Гипертекст — способ организации информации в виде взаимосвязанных документов или их фрагментов, между которыми возможен переход с помощью гиперссылок.

Опр Гиперссылка — элементов гипертекста, обеспечивающих переход к другому документу или его части.

Опр HTML (HyperText Markup Language) — язык разметки гипертекста, предназначенный для структурирования информации на веб-странице.

Опр Клиент — программа, инициирующая запрос к серверу для получения данных или услуг (в частности, браузер).

Опр Сервер — программа, принимающая и обрабатывающая запросы от клиентов для предоставления им ответных данных.

8 Технологии работы с медиаконтентом как направление информационных технологий

Технологии работы с медиаконтентом. Медиаконтент

Опр Технологии работы с медиаконтентом — технологии создания, обработки, хранения и передачи медиаконтента, анализа мультимедийной информации (объединяющей в себе разные виды медиаконтента).

Опр Медиаконтент — разнообразная цифровая информация, представленная в аудио- и визуальных форматах (видео, фото).

Направления деятельности

- ☐ Обработка изображений и видео (сжатие (компрессия), улучшение качества).
- ☐ Поддержка информационных ресурсов: администрирование сайтов и серверов, обновление контента, обеспечение безопасности и доступности ресурсов.
- ☐ Компьютерное зрение (CV): анализ визуального медиаконтента (в частности, распознавание).
- ☐ Обработка и синтез речи (распознавание речи, голосовые ассистенты, преобразование текста в речь).

9 Базы данных, базы знаний как средства поддержки информационных технологий

База данных

Опр База данных (БД) — это организованная совокупность структурированных данных, предназначенных для хранения, поиска и обработки информации.

Управление БД

Взаимодействие с БД осуществляется через **СУБД** (систему управления базой данных): создание таблиц хранения данных, управление доступом, защита данных, выполнение запросов через **SQL** (Structured Query Language – язык структурированных запросов) для управления, создания, модификации и получения данных в БД.

База знаний

Опр База знаний — это структурированная совокупность знаний о предметной области, включающая факты и правила вывода, т.е. хранилище знаний (правила и зависимости, за счет которых заключается по имеющимся данным новые, заключающие вердикт данные).

Базы знаний используются внутри **экспертных систем** (ЭС) — ППО, использующего базу знаний и механизм логического вывода для решения задач на уровне эксперта.

Прояснение различия

- Базы данных отвечают на вопрос: “Какие данные есть?”. Например, (Температура двигателя) = 105, (Обороты) = 900.
- Базы знаний хранят правила и зависимости, отвечают на вопрос: “Какие связи существуют между данными?”. Например, IF (температура двигателя) > 100, THEN <перегрев>.
- ЭС выдает конкретный логический вердикт по выводу базы знаний, отвечает на вопрос: “Какой вывод можно сделать из имеющихся данных?”. Например, “Обнаружен перегрев двигателя. Рекомендуется проверить систему охлаждения.”

10 Искусственный интеллект и машинное обучение как одно из направлений информационных технологий

Искусственный интеллект

Опр Искусственный интеллект (ИИ) — технологии создания программных систем, способные имитировать когнитивные функции человека: обучение, анализ (в частности, диагностика), распознавание образов (в частности, CV), принятие решений, обработка естественного языка (NLP: анализ и генерация человеческой речи или текстов).

Машинное обучение

Опр Машинное обучение (МО) — это раздел ИИ, основанный на создании алгоритмов, способных обучаться на данных (data) и улучшать результаты без явного программирования правил.

Типы МО

- Обучение с учителем: есть входные данные X , правильные ответы Y , модель учится отображению $X \rightarrow Y$. В частности, задача классификации: задано конечное множество объектов, для которых известно, к каким классам они относятся. Цель: автоматически отнести новые объекты к одному из заранее известных классов на основе анализа набора признаков. Например, соотнести спам или не спам.
- Обучение без учителя: есть только данные X , правильных ответов нет. В частности, задача кластеризации: заданы объекты, группирующиеся по сходству без заранее заданных классов. Цель: самостоятельно выявить скрытые группы (кластеры) без использования размеченных ответов. Например, есть данные о возрасте, количеству покупок, а алгоритм сам выделяет группы: “молодые”, “постоянные клиенты” и т.д.

11 Информационная безопасность как одно из направлений информационных технологий

Информационная безопасность

Опр Информационная безопасность — технологии защиты информации от угроз, утечек и атак, направленные на обеспечение конфиденциальности, целостности и доступности данных.

Основные аспекты

- ☐ Анализ угроз безопасности.
 - ☐ Анализ возможных векторов атак (определение путей, через которые может быть осуществлена атака на систему, например, через уязвимости в ПО).
 - ☐ Анализ способов противодействия (выявление методов защиты от атак, например, антивирусные системы).
 - ☐ Построение модели угроз (создание схемы потенциальных атак и защитных мер, чтобы заранее понимать, какие угрозы могут возникнуть).
- ☐ Тестирование на проникновение (пентест) — метод реальных атак на систему с целью выявления ее слабых мест (атака системы, анализ уязвимости, предложения по усилению защиты).
- ☐ Защита компьютерных систем:
 - ☐ Криптографические мероприятия: шифрование данных, создание безопасных каналов связи.
 - ☐ Организационно-технические мероприятия: политики безопасности, обучение персонала, контроль доступа.

Дополнительно

Опр Идентификация (“кто Вы?”) — процесс сообщения системе пользователем своего уникального идентификатора (логин, email, номер телефона).

Опр Аутентификация (“точно ли это Вы?”) — процесс проверки подлинности заявленной личности пользователя (пароль, пин-код, SMS-код, отпечаток пальца, Face ID).

Опр Авторизация (“что Вам разрешено?”) — процесс предоставления прав доступа пользователю после успешной аутентификации (права пользователя, модератора, админа).

12 Крупные комплексные решения как одно из направлений информационных технологий

Крупные комплексные решения

Опр Крупные комплексные решения — это системы, состоящие из множества взаимосвязанных компьютеров, работающих совместно для решения масштабных и сложных задач в различных сферах.

Примеры крупных комплексных решений

- Кластерные системы — для повышения производительности (НРС – высокопроизводительные вычисления из кластеров сотен или тысяч связанных серверов для решения математических задач) и отказоустойчивости (НА – кластер высокой доступности, гарантирующий минимальное время простоя за счет аппаратной избыточности: нагрузки вышедшего из строя узла переносятся на исправные узлы).
- Мультимедийные платформы — для обработки и передачи мультимедийного контента (платформы стриминга видео: YouTube, Netflix), образовательные платформы (Stepik, LMS).
- Поисковые машины — для поиска информации по запросам (Google, Yandex) с применением алгоритмов ранжирования и индексации.
- Системы бронирования — для резервирования услуг: билеты, отели (Booking.com), аренда автомобилей и т.д.
- Инженерные системы — для решения инженерных задач: моделирование, проектирование (CAD-системы), управление сложными технологическими процессами (SCADA).

13 Программирование. Основные понятия и процессы

Программирование

Опр Программирование — это теоретическая и практическая деятельность, направленная на создание и сопровождение программ.

Опр Программа — это запись алгоритма в форме, пригодной для его обработки вычислительной системой.

Опр Язык программирования — формальная знаковая система для записи алгоритмов в форме, пригодной для их обработки средствами вычислительной техники.

Алгоритм

Опр Алгоритм — это строго определенная последовательность шагов, обеспечивающая достижение решения поставленной задачи за определенное количество шагов.

Формы записи алгоритма

- Словесное описание — описание алгоритма словами.
- Блок-схема — графическое представление алгоритма в виде блоков и стрелок.
- Псевдокод — запись алгоритма в виде упрощенного кода, похожего на программный язык, но более понятного для какого бы ни было человека.

14 Автоматизированные системы и основные виды обеспечения их функционирования

Автоматизированные системы

Опр Автоматизированные системы (АС) — сложные человеко-машинные системы, осуществляющие сбор, хранение и обработку информации с использованием средств автоматизации и вычислительной техники. В отличие автоматических систем, автоматизированные требуют некоторого человеческого вмешательства для корректной работы.

Классификация АС

- Системы управления (АСУ) — для управления объектами и процессами. Например, системы управления производственными процессами, системы управления транспортом.
- Системы обработки данных (АСОД/АСОИ) — для обработки, хранения и анализа данных. Например, системы дистанционного обучения, планирования ресурсов предприятия (ERP-системы, как 1С).

Структура АСУ

- Объект управления — то, что мы управляем (завод, транспортное средство, робот).
- Контур управления — механизм, через который мы влияем на объект управления (контроллеры, сенсоры, исполнительные устройства).

Принцип работы АСУ

1. Сбор информации: система получает данные от объекта управления (с датчиков).
2. Обработка данных: система анализирует данные, используя заранее заданные правила.
3. Управляющие воздействия: на основе анализа система формирует управляющие команды (включить насос).
4. Обратная связь: новые параметры состояния объекта фиксируются и снова обрабатываются системой.

Виды обеспечения функционирования АС

См. вопрос 15

15 Основные виды обеспечения функционирования и их взаимосвязь (!!!)

Виды обеспечения функционирования АС

- Техническое — комплекс механических средств, предназначенный для обеспечения функционирования АС: вычислительные машины, серверы, датчики, исполнительные устройства.
- Математическое — комплекс методов, моделей, алгоритмов для обработки данных (для вычислений, прогнозирования).
- Программное — комплекс программ, реализующий обработку алгоритмов данных при решении задач АС.
- Информационное — совокупность данных, сопровождающих решения задач в процессе человеко-машинной обработки (данные о текущем состоянии процесса, результаты вычислений и прогнозов, историческая информация о предыдущих действиях).
- Организационное — совокупность предписаний, регламентирующих деятельность людей в рамках АС, т.е. взаимодействие между человеком и машиной, а также координацию всех участников системы (процесс работы оператора, алгоритм работы системы в экстренных ситуациях).

Взаимосвязь всех видов обеспечения

- Техническое предоставляет средства для работы системы, на которых выполняется обработка данных (математическое и программное обеспечения).
- Математическое и программное обеспечения работают с данными, обрабатывая их для решения задач.
- Информационное предоставляет данные для обработки и анализа, которые используются программным и математическим обеспечениями.
- Организационное регулирует процессы и действия людей, взаимодействующих с системой, обеспечивая правильную работу всех компонентов.

16 Программное обеспечение автоматизированных систем как комплекс программ. Особенности создания комплексов программ

Комплекс программ

Опр Комплекс программ (КП) — это программная система высокой степени сложности, состоящая из множества взаимосвязанных компонентов.

Характеристика КП

- ☐ Число компонентов: десятки, сотни и тысячи взаимодействующих компонентов.
- ☐ Число строк программного кода (LOC): оно может быть огромным.
- ☐ Число разработчиков: десятки специалистов.
- ☐ Трудозатраты: создание требует значительных усилий, часто измеряемых в человеко-годах (пояснение: 100 человеко-лет – либо 1 человек работает над проектом 100 лет, либо 100 человеком над ним работают год).
- ☐ Стоимость: программная часть КП зачастую значительно дороже ее аппаратной части.

Особенности проектирования КП

- ☐ Качество: важно обеспечить высокое качество всех компонентов, поскольку они не являются изолированными программами, а должны работать как единая система.
- ☐ Работа в коллективе: разработка КП требует умения эффективно работать в команде и организовывать процессы разработки.
- ☐ На этапе проектирования важно сформировать архитектуру комплекса такую, которая бы обеспечила корректное взаимодействие всех компонентов.
- ☐ Тестирование: функциональное и интеграционное.

17 Основная задача профессионального программирования и понятие качества ПО (!!!)

Основная задача

Основная задача профессионального программирования — создание качественного ПО.

Качество

Опр 1 Качество — весь объем признаков и характеристик продукции или услуги, который относится к их способностям удовлетворять установленным или предполагаемым потребностям (По ГОСТу “Информационная технология. Оценка программной продукции. Характеристики качества и руководства по их применению”).

Опр 2 Качество — способность ПО подтверждать свою спецификацию, т.е. соответствовать заранее определенным в техническом задании набору характеристик или свойств.

18 Понятие качества ПО и его основные характеристики (!!!)

Качество

См. вопрос 17

Основные характеристики

1. Функциональность,
2. Надежность,
3. Сопровождаемость,
4. Удобство,
5. Эффективность,
6. Способность к взаимодействию — возможность ПО взаимодействовать с другими продуктами для обмена данными или интеграции,
7. Согласованность — соответствие ПО установленным стандартам и нормативным документам,
8. Защищенность — способность ПО защищать данные от несанкционированного доступа и атак,
9. Восстанавливаемость — способность ПО сохранять свою работу, данные и функциональность после сбоев в приемлемые сроки,
10. Переносимость (мобильность, портируемость) — свойство ПО сохранять свою работоспособность при переносе на другие аппаратные или программные среды.

19 Функциональность как характеристика качественного ПО

Функциональность

Опр Функциональность (функциональные возможности) — способность ПО выполнять определенный набор действий, удовлетворяющий заданным требованиям.

Функциональность оценивает, что может делать ПО, насколько оно выполняет все заявленные функции и задачи, которые были предусмотрены в спецификации. Например, функциональность веб-приложения для онлайн-банкинга включает возможность проверить баланс, проведение переводов, просмотра истории транзакции.

20 Надежность как характеристика качественного ПО

Надежность

Опр Надежность — способность ПО безотказно выполнять свои функции в полном соответствии со спецификацией и адекватно реагировать на действия пользователя или воздействия извне в течение заданного периода времени с большей долей вероятности.

Надежность оценивает, насколько стабильно ПО выполняет свои функции и как оно справляется с внешними воздействиями. Например, надежность веб-приложения для онлайн-банкинга будет определяться тем, насколько она будет работать без сбоев при высоких нагрузках (много пользователей одновременно), неожиданных действиях (многочисленный некорректный ввод пароля), внешних воздействиях (плохое интернет-соединение).

21 Удобство как характеристика качественного ПО

Удобство

Опр Удобство (практичность) — способность ПО минимизировать усилия пользователя при подготовке исходных данных, использовании программы и оценке результатов, а также вызывать у пользователя положительные эмоции.

Это характеристика UX – пользовательский опыт. Простой и интуитивно понятный интерфейс – высокое удобство. Так, у веб-приложения онлайн-банкинга удобство зависит от того, насколько легко пользователю проверять баланс, делать переводы, искать и просматривать историю операций.

22 Сопровождаемость как характеристика качественно-го ПО

Сопровождаемость

Опр Сопровождаемость (расширяемость) — способность ПО легко модифицироваться для исправления ошибок и адаптации к новым требованиям с минимальными усилиями.

Сопровождаемость веб-приложения для онлайн-банкинга включает возможность быстрого исправления багов, добавления новых функций (новый тип перевода), адаптации к изменяющимся регламентам (изменения в законодательстве).

Сопровождаемость vs Сопровождение

Сопровождаемость — способность ПО изменяться, тогда как сопровождение — сам процесс управления изменениями и поддержания ПО в стабильном состоянии.

23 Эффективность как характеристика качественного ПО

Эффективность

Опр Эффективность — соотношение между качеством функционирования ПО и объемом использования ресурсов (память, время работы и т.д.) при определенных условиях.

Эффективность оценивает, насколько хорошо ПО использует ресурсы (память, процессорное время) при выполнении своих задач. Так, эффективность веб-приложения для онлайн-банкинга будет зависеть от того, насколько быстро загружаются страницы, как эффективно работает поиск по данным.

24 Принципы выбора характеристик качества ПО. Понятия «достаточно хорошего» программного продукта (!!!)

Принцип выбора характеристик качества ПО

- Степень соответствия сроков выпуска ранее запланированным: качество можно оценить по тому, насколько сроки выпуска соответствуют ранее установленным планам, задержки могут свидетельствовать по проблемах в процессе разработки.
- Корректность документации: документация должна точно и полно описывать функциональность ПО, а ошибки в ней могут привести к неправильному использованию, а значит и к снижению качества системы.
- Удобство и скорость обучения пользователей работе с ПО: ПО должно быть интуитивно понятным, а процесс обучения пользователя — быстрым и безболезненным.

Принцип выбора характеристик для «достаточно хорошего» ПО. Понятие «достаточно хорошего» ПО

- Баланс между сроками и качеством: нужно найти оптимальное соотношение между временем разработки и качеством продукта. Это значит, что в некоторых случаях можно жертвовать незначительным улучшением, чтобы успеть в сроки, и наоборот.
- Стоимость и качество взаимосвязаны. Иногда слишком высокое качество может не оправдать затрат, если продукт не требует таких ресурсов и наоборот. Нужно хотя бы стремиться к MVP (Minimum Viable Product) — минимально жизнеспособному продукту, стадии разработки, на которой продукт способен удовлетворять набору требований и выполнять минимальный набор задач заказчика.

Опр «Достаточно хорошего» ПО — ПО, удовлетворяющее базовым требованиям и выполняющее свои функции с удовлетворительным качеством, при

этом ресурсы на его разработку и сопровождение не превышают разумные пределы.

Требования к «достаточно хорошему» ПО, или принципы must-, should-, could- have для приоритизации

- Must-have — обязательные требования, являющиеся критически важными для системы, без которых ПО не будет работать или будет работать с ошибками, реализующиеся в первую очередь. Например, авторизация пользователя в приложении онлайн-банкинга.
- Should-have — желательные требования, не являющиеся критическими для функционирования ПО, но очень желательные, повышающие UX. Если они не реализованы, это не приведет к сбоям системы, но добавят ценность продукту. Например, мобильная версия сайта, темная тема в приложении онлайн-банкинга.
- Could-have — дополнительные требования, совсем не критические, то, что можно добавить, если есть дополнительные ресурсы и время, делают систему более интересной для пользователя. Например, возможность сменить аватар, получать уведомления о расходах в приложении онлайн-банкинга.

25 Тестирование и отладка ПО как взаимосвязанные процессы. Основные принципы тестирования

Тестирование и отладка

Опр Тестирование — процесс выполнения программы с целью обнаружения ошибок, т.е. неадекватного поведения программы с точки зрения пользователя.

Опр Отладка (дебаггинг) — процесс локализации (определения участков кода, ответственных за сбой) и устранения ошибок, которые были обнаружены в ходе тестирования (здесь кроется взаимосвязь).

Тестирование и отладка — это итеративный процесс: после устранения ошибок тестирование выполняется снова для подтверждения исправлений и поиска новых проблем.

Основные принципы тестирования

- Всегда предполагается наличие ошибок (поэтому надо быть внимательным и тщательно проверять программу).
- Тестирование включает в себя не только входные данные, но и ожидаемые результаты работы программы для этих данных.
- Тестовые данные должны быть реалистичными и понятными, т.е. отражающими реальную ситуацию использования программы.
- Тестирование начинается на этапе разработки, т.е. программист должен задумываться о тестах еще при проектировании системы.
- Фиксация результатов тестирования, ведение протокола тестирования.
- Тестируются как корректные, так и некорректные данные, чтобы убедиться, что программа работает корректно в обеих ситуациях.
- Важно знать, что программа должна делать и чего не должна.
- Тестирование должно воспроизводиться и после внесения изменений, чтобы проверить, не возникли ли новые проблемы (регресс).

- Ошибки склонны группироваться, т.е. часто происходят не в одиночку, а значит, если в одной части программы найдена ошибка, то есть высокая вероятность, что в других частях программы могут быть аналогичные ошибки.
- Тестировать должны другие люди, а не автор программы: разработчик может быть слеп к своим собственным ошибкам.

Дополнительно

Тестирование является динамическим способом обеспечения качества. К статическим относятся “прокрутка” (ручное выполнение алгоритма), математическое доказательство правильности программы, экспертный анализ кода, использование статического анализатора (инструмента для автоматического анализа исходного кода без выполнения программы).

Тестирование может быть не только ручным, но и автоматизированным.

26 Тестирование и отладка ПО как взаимосвязанные процессы. Основные принципы отладки

Тестирование и отладка

См. вопрос 25

Основные принципы отладки

- Избегать экспериментов и необдуманных исправлений, т.е. нужно четко понимать цель изменения и возможные последствия. Особенно это важно при сопровождении ПО. Каждый шаг должен быть обоснованным и продуманным.
- Не вносить множество изменений одновременно, т.к. это затруднит анализ результатов и оценку причин возникновения ошибок. Изменения должны быть поэтапными.
- Нужно стремиться к тому, чтобы ошибка была полностью понята и объяснена, т.е. необходимо разобраться в корне проблемы, а не устранять ее симптомы.
- Добавлять новые тесты для проверки изменений в процессе отладки. Это помогает удостовериться, что изменения не вызвали новых проблем. Тестирование должно быть непрерывным и постоянно обновляемым.

Дополнительно

Опр Ретроанализ — восстановление последовательности действий в программе, приводящей к возникновению ошибки.

27 Основные стратегии тестирования. Понятие о критериях тестирования (!!!)

Основные стратегии тестирования

- ☐ Черный ящик.
- ☐ Белый ящик.
- ☐ Серый ящик – стратегия, сочетающая в себе элементы тестирования белого и черного ящиков (ограниченное знание о внутренней структуре системы, сохранение фокуса на внешнем поведении системы). Тестировщик использует частичное знание о системе для более целенаправленного тестирования.

Критерии тестирования

Опр Критерии тестирования — условия и требования, определяющие, когда тестирование считается завершенным, а система (или ее компонент) прошла тестирование с требуемым качеством. Критерии будут основываться зависимости от того, насколько тестировщик понимает внутреннее устройство программы, т.е. от выбора стратегии тестирования.

28 Применение стратегии черного ящика при тестировании. Возможные критерии тестирования (!!!)

Применение стратегии черного ящика при тестировании

Опр “Черный ящик” — стратегия тестирования, при котором не учитывается внутренняя структура системы, где тестировщик работает только с входами и выходами системы.

Плюсы

Выявляется поведение системы при различных входных данных.

Минусы

Отсутствие знания внутренней структуры (невозможно покрыть все возможные пути выполнения программы, что может привести к пропуску некоторых ошибок), сложность тестирования в сложных системах (тяжело предсказать все возможные комбинации входных данных, что может привести к недостаточному тестированию).

Критерии тестирования черного ящика

- Критерий тестирования функций: есть хотя бы раз на каждую функцию (чтобы удостовериться, что он выполняет свою задачу).
- Критерий тестирования классов входных данных: корректные (позитивное тестирование), некорректные данные (негативное тестирование), граничные значения. Если программа функционирует корректно при вводе некоторого произвольного набора данных некоторого класса, то она должна функционировать корректно при вводе любых данных этого класса.
- Критерий тестирования классов выходных данных: проверка правильности выходных данных в зависимости от различных входных, проверка соответствия результатам.

- Критерий тестирования области допустимых значений, т.е. с данными, лежащими в допустимой области значений для входа/вывода (проверка корректной отработки минимума и максимума, нормальных, граничных и исключительных значений).
- Критерий тестирования длины набора входных данных: пустой набор, единичный, недопустимо короткий или длинный, минимальный или максимальный допустимый, нормальной длины.
- Критерий упорядоченности входных данных: неупорядоченные, упорядоченные прямо или обратно данные.

29 Применение стратегии белого ящика при тестировании. Возможные критерии тестирования (!!!)

Применение стратегии белого ящика при тестировании

Опр “Белый ящик” — стратегия тестирования, при котором тестирующий имеет доступ к исходному коду системы.

Плюсы

Идет тестирование не только функциональности, но и структуры программы, т.е. проводится анализ кода.

Минусы

Высокая трудоемкость (особенно в больших и сложных системах), необходимость доступа к коду, высокая квалификация специалистов (нужны глубокие знания в программировании и аналитических способностей), низкая надежность проверки циклов (особенно сложны вложенные циклы).

Критерии тестирования белого ящика

- Критерий покрытия операторов (проверка того, что каждый оператор (if, switch) был выполнен хотя бы раз во время тестирования).
- Критерий покрытия ветвей (проверка того, что каждая возможная ветвь (например, if-else if-else) была протестирована).
- Критерий покрытия условий (проверка того, что для каждого логического условия были протестированы как истинные, так и ложные значения). Например, для условия if ($x > 0$ and $y < 5$) тест должен проверить истинность, так и ложность этого условия при $x \leq 0$ или $y \geq 5$.
- Критерий покрытия решений и условий (проверка выполнения всех возможных решений и комбинаций условий в программе, особенно касается сложных условий с несколькими логическими операторами). Например, if ($x > 0$ and $y < 5$) or $z == 10$. Здесь нужно проверить не только все

условия по отдельности, но и их комбинации (оба условия истинны, оба ложны, одно истинно, а другое ложно).

- Критерий комбинаторного покрытия условий (проверка всех возможных комбинаций значений всех логических условий программы. Например, для `if (x > 0 and y < 5) or z == 10` нужно рассмотреть все комбинации `x`, `y`, `z`, например `(x>0, y<5, z<10)`, `(x>0, y<5, z<=10)`, `(x>0, y<5, z==10)` и т.д.).

Дополнительно

Опр Тестовое покрытие — степень охвата всей системы тестам (по отношению к коду и пользовательским сценариям). Измеряется в процентах и зависит от того, сколько операторов, ветвей, условий и решений в коде было протестировано.

30 Стратегия черного ящика и нагрузочное тестирование

Стратегия черного ящика

См. вопрос 28

Нагрузочное тестирование

Опр Нагрузочное тестирование — процесс тестирования системы для проверки ее производительности под различными уровнями нагрузки с целью убедиться, что система может работать эффективно и без сбоев, когда количество пользователей или объем данных увеличивается.

Виды нагрузочного тестирования

- Тестирование производительности — как быстро система выполняет свои задачи при различных условиях нагрузки с целью измерить время отклика и скорость обработки данных. Например, сколько времени занимает загрузка страницы веб-приложения при разных объемах данных.
- Стресс-тестирование — проверка того, как система работает при экстремальных нагрузках, критических условиях. Выявляется, как система выходит из строя при перегрузке или какие сбои происходят.
- Объемное тестирование — проверка, как система справляется с определенным объемом данных или количеством пользователей с целью удостовериться, что система может обрабатывать заранее прогнозируемую нагрузку. Например, проверка того, что если X пользователей работают с системой одновременно, то все стабильно.
- Тестирование надежности (стабильности) — проверка того, как система работает в течение длительного времени под нормальной или повышенной нагрузкой. Важно для приложений, работающих 24/7.

31 Стратегия черного ящика. Модульное, интеграционное и регрессионное тестирование

Стратегия черного ящика

См. вопрос 28

Модульное тестирование

Опр Модульное (юнит) тестирование — тестирование отдельных компонентов или модулей программы. Например, проверка работы функции.

Интеграционное тестирование

Опр Интеграционное тестирование — тестирование системы или ее частей после их объединения, т.е. проверка того, как модули взаимодействуют вместе. Например, соединение базы данных с сервером, организация передачи данных между ними.

Регрессионное тестирование

Опр Регрессионное тестирование — тестирование системы после внесения изменений, чтобы удостовериться, что новые изменения не вызвали проблем в уже работающем функционале.

Дополнительно

Опр Смок-тестирование — поверхностное тестирование, выполняемое для проверки, что базовая функциональность системы работает, как ожидается. Цель: убедиться, что система не имеет критических ошибок, которые могут помешать дальнейшему тестированию. Например, проверка того, что приложение запускается и основные функции работают: логин, главная страница — работают, следовательно, можно проводить более глубокое тестирование.

32 Понятие стиля программирования (общая характеристика)

Регрессионное тестирование

Опр Стиль программирования — совокупность приемов кодирования и документирования программ, включающие в себя рекомендации и принципы, которые помогают создавать читаемый, поддерживаемый и надежный код.

Стандарты и соглашения

Для создания хорошего стиля программирования существуют международные стандарты и соглашения, которые помогают унифицировать кодирование и делают его более читаемым:

- Java Code Conventions (1991) — официальный стиль программирования для Java.
- Google Style Guides — стиль для Java и C++, принятый компанией Google.
- Object Pascal Style Guide (Embarcadero) — соглашения для программирования на языке Object Pascal.
- PEP8 (Python Enhancement Proposal) — стиль программирования для Python.

Книги по стилю программирования

- “Совершенный код” (Макконелл) — книга о лучших практиках программирования, включает методы написания качественного кода.
- “Чистая архитектура” (Мартин Р.С.) — рекомендации по созданию архитектуры ПО.
- “Рефакторинг” (Фаулер, Бек, Брант) — книга о рефакторинге существующего кода, улучшения его структуры без изменения функционала.