

Тест из 25 вопросов

Стадии проектирования автоматизированных информационных систем

Салимли Айзек

1. На какой стадии впервые фиксируется *полная* совокупность функциональных и нефункциональных требований в форме, пригодной для приёмки?
 - a) Содержательное описание предметной области
 - b) Формализация (математическая модель)
 - c) **Разработка и согласование ТЗ на систему и подсистемы**
 - d) Пилотная эксплуатация
2. Что является корректным критерием завершения стадии *содержательного описания*?
 - a) Наличие исходного кода ключевых модулей
 - b) **Согласованный глоссарий и модели бизнес-процессов, подтверждённые заказчиком**
 - c) Проведение нагрузочного тестирования
 - d) Утверждённый план сопровождения
3. Какой результат *формализации* обеспечивает проверяемость непротиворечивости требований?
 - a) Диаграммы пользовательских интерфейсов (UI mockups)
 - b) **Формальная модель (например, онтология/сети Петри/AADL) с правилами вывода/симуляцией**
 - c) Проект плана внедрения с бюджетом
 - d) Паспорт ИБ-мероприятий
4. Что является основным назначением пилотного проекта перед промышленным внедрением?
 - a) Сокращение документации
 - b) Замена всех интеграций на временные
 - c) **Проверка гипотез на реальных данных и уточнение рисков/метрик качества**
 - d) Переписывание архитектуры с нуля
5. Как обеспечить устойчивость к изменениям предметной области при переходах между стадиями?

- a) Жёсткое кодирование правил в интерфейсе
 - b) Отказ от версионирования
 - c) **Трассируемость «требование→модель→архитектура→реализация→тест»**
 - d) Хранение требований только в почте
6. Что из ниже перечисленного *непосредственно* снижает стоимость правок схем данных на поздних стадиях?
- a) Фиксация требований только в ТЗ
 - b) Ранний выбор СУБД
 - c) **Метауровень: метаданные/металогика/мета-интерфейсы и генерация артефактов**
 - d) Увеличение численности команды
7. Какой артефакт связывает содержательное описание с формализованной моделью?
- a) План закупок
 - b) **Карты сценариев/варианты использования с формальными пред-/постусловиями**
 - c) Политика резервного копирования
 - d) Руководство пользователя
8. Что является корректным критерием готовности ТЗ к разработке?
- a) Указаны только архитектурные диаграммы
 - b) **Определены границы системы, интерфейсы с внешними ИС, нефункциональные требования и критерии приёмки**
 - c) Утверждён только бюджет
 - d) Есть прототип UI без описания API
9. На какой стадии целесообразно провести моделирование очередей/нагрузок для оценки производительности?
- a) Только после внедрения
 - b) **На стадии формализации/проектирования с использованием исполняемых моделей**
 - c) На стадии сопровождения
 - d) На этапе закупки серверов, без модели
10. Какая практика минимизирует разрыв между пилотом и промышленным внедрением?
- a) Отдельные, несовместимые ветки кода
 - b) **Единый конвейер CI/CD и миграции БД, применимые и на пилоте, и в проде**
 - c) Запрет автоматического тестирования

- d) Хардкод тестовых заглушек в основной ветке
11. Что корректно характеризует переход от классического подхода к метамодельному?
- a) Отказ от формальной модели
 - b) **Сдвиг фокуса с «ручной» разработки артефактов к описанию метамодели и генерации**
 - c) Увеличение числа нестандартизованных интерфейсов
 - d) Игнорирование требований к качеству
12. Какой показатель следует зафиксировать ещё в ТЗ для объективной приёмки после внедрения?
- a) Цветовая схема интерфейса
 - b) **Целевые SLA/SLO: время отклика, доступность, ошибки на транзакцию**
 - c) Фамилии всех пользователей
 - d) Набор скриншотов
13. Что из перечисленного относится к проверке *валидации* на стадии пилота?
- a) **Подтверждение, что система решает бизнес-задачу на реальных процессах**
 - b) Проверка соответствия коду стандартам оформления
 - c) Анализ покрытия модульными тестами
 - d) Линтинг SQL-скриптов
14. Какая роль должна утвердить глоссарий и границы системы до формализации?
- a) Архитектор без участия бизнеса
 - b) Разработчик интерфейсов
 - c) **Заказчик (владелец процессов) совместно с системным аналитиком**
 - d) Команда сопровождения
15. Какой набор артефактов оптимален для передачи в разработку после формализации?
- a) Набор презентаций и скриншотов
 - b) **Формальная модель, спецификация интерфейсов, модель данных, критерии тестирования**
 - c) Только диаграммы классов
 - d) Только текст ТЗ без интерфейсов
16. При интеграции с внешними ИС на какой стадии нужно зафиксировать контракты API и версии?
- a) На сопровождении, по факту ошибок

- b) На этапе пилота, не документируя
 - c) **В ТЗ/проектировании с управлением конфигурацией и версионированием**
 - d) Во время тестовой эксплуатации без контроля изменений
17. Что является корректным выходом стадии «внедрение»?
- a) Черновые инструкции для отдельных пользователей
 - b) **Принятая система по актам приёмки с выполненными критериями и обучением персонала**
 - c) Только запуск сервера
 - d) Устные договорённости о сроках доработок
18. Какой риск наиболее типичен при переходе от содержательного описания к формальной модели?
- a) Снижение прозрачности требований
 - b) **Потеря смысла терминов при некорректном мэппинге в формальные конструкции**
 - c) Избыточная автоматизация тестов
 - d) Недостаток серверных мощностей
19. Для снижения стоимости исправлений где рационально сосредоточить основные проверки согласованности?
- a) **На стадии формализации/проектирования: статический анализ и симуляция моделей**
 - b) Только в нагрузочных тестах на проде
 - c) На этапе сопровождения
 - d) В визуальном дизайне UI
20. Какая метрика подтверждает готовность к промышленной эксплуатации после пилота?
- a) Количество прототипов UI
 - b) **Стабильные метрики надёжности/производительности на пилотных данных, совпадающих по профилю с боевыми**
 - c) Число страниц ТЗ
 - d) Количество задействованных разработчиков
21. Как обеспечить переносимость результатов формализации в реализацию?
- a) Свободное толкование формальных артефактов программистами
 - b) **Генерация схем/конфигураций/каркасов из метамодели + проверки соответствия (model-to-text/consistency checks)**
 - c) Хранение модели в закрытых форматах без экспорта

d) Удаление формальных ограничений при кодировании

22. Что должно быть определено до начала пилота в части данных?

- a) **Стратегия миграции и обезличивания/маскирования, эталонные наборы тестовых данных**
- b) Только дашборды
- c) Эскизы экранных форм
- d) План командировок

23. Какой признак указывает, что стадия «ТЗ» выполнена формально и несёт риски на внедрении?

- a) **Отсутствуют измеримые критерии приёмки для нефункциональных требований**
- b) В ТЗ описаны интерфейсы и ограничения
- c) Есть матрица трассируемости
- d) Определён регламент изменения требований

24. Какой подход корректен для управления изменениями между пилотом и внедрением?

- a) Вносить правки напрямую в прод без версий
- b) **Процедура изменения требований (change control), обновление ТЗ/моделей и повторная приёмка**
- c) Фиксировать изменения устно
- d) Отключить автоматические тесты на время

25. Какой артефакт завершает цикл «требование→реализация» на стадии внедрения?

- a) Техническое задание в черновике
- b) **Протокол приёмочно-сдаточных испытаний с ссылками на требования и результаты тестов**
- c) Набор макетов интерфейса
- d) План отпусков команды