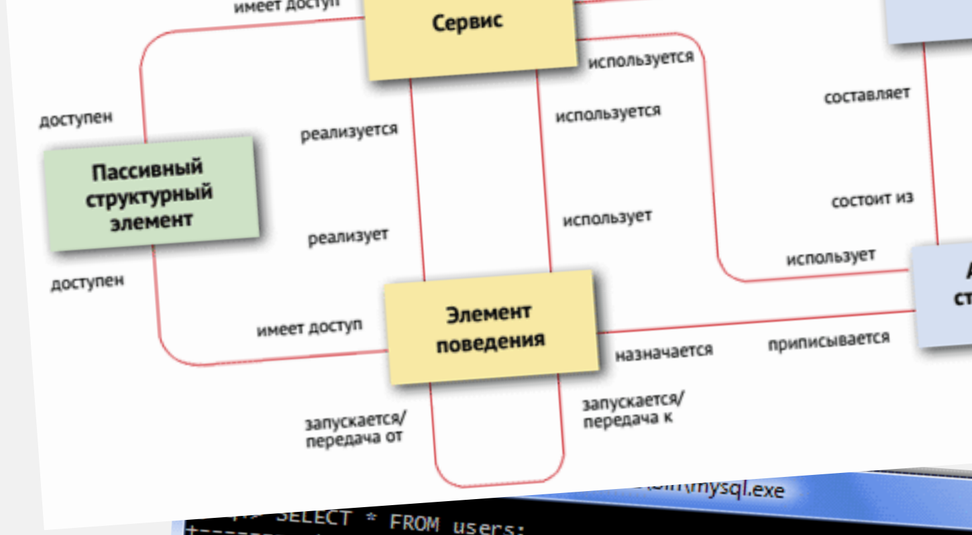
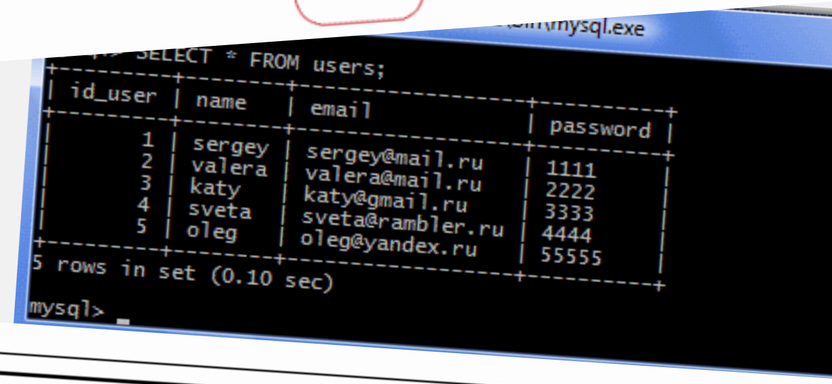
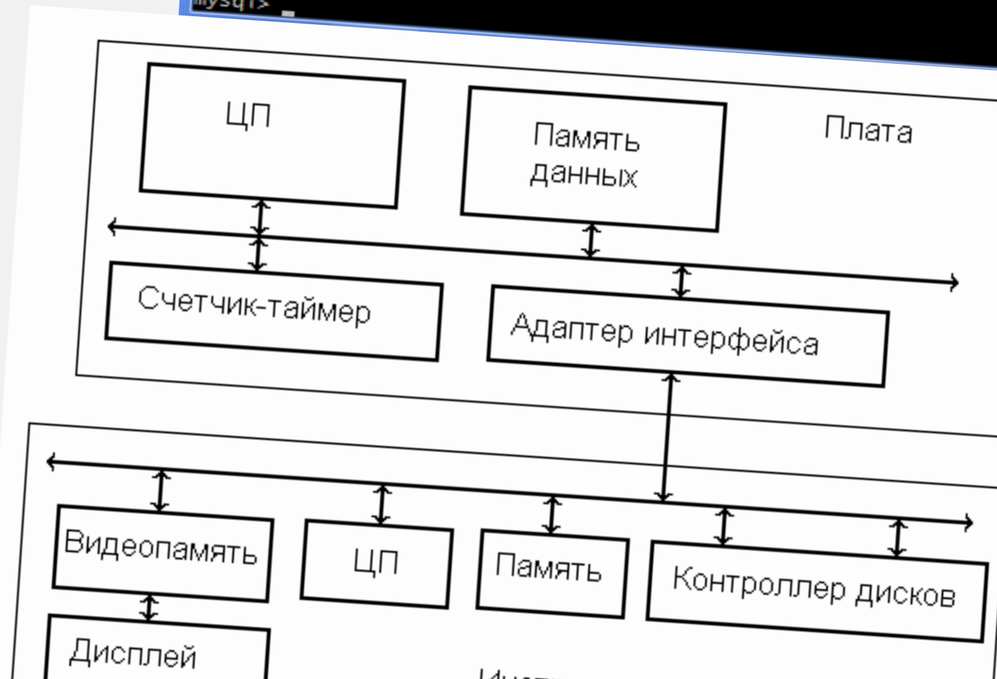
# Функциональные компоненты

Тестирование, отладка и интеграция функциональных компонентов

*испытание на внимательность и силу духа*



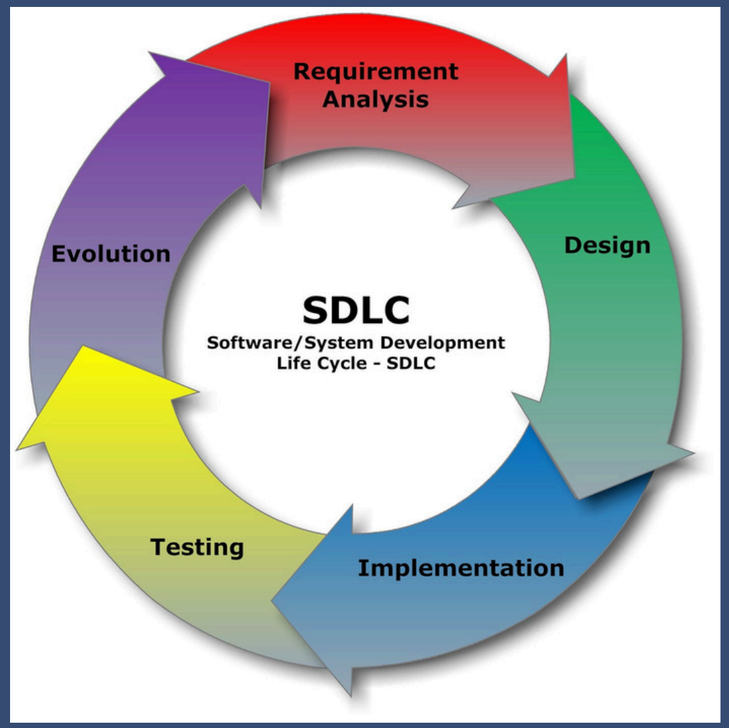



# Роль тестирования, отладки и интеграции ♖

Тестирование выявляет расхождения между ожидаемым и фактическим поведением.

Отладка ищет причины дефектов и обеспечивает их устранение.

Интеграция проверяет согласованную работу компонентов в составе системы

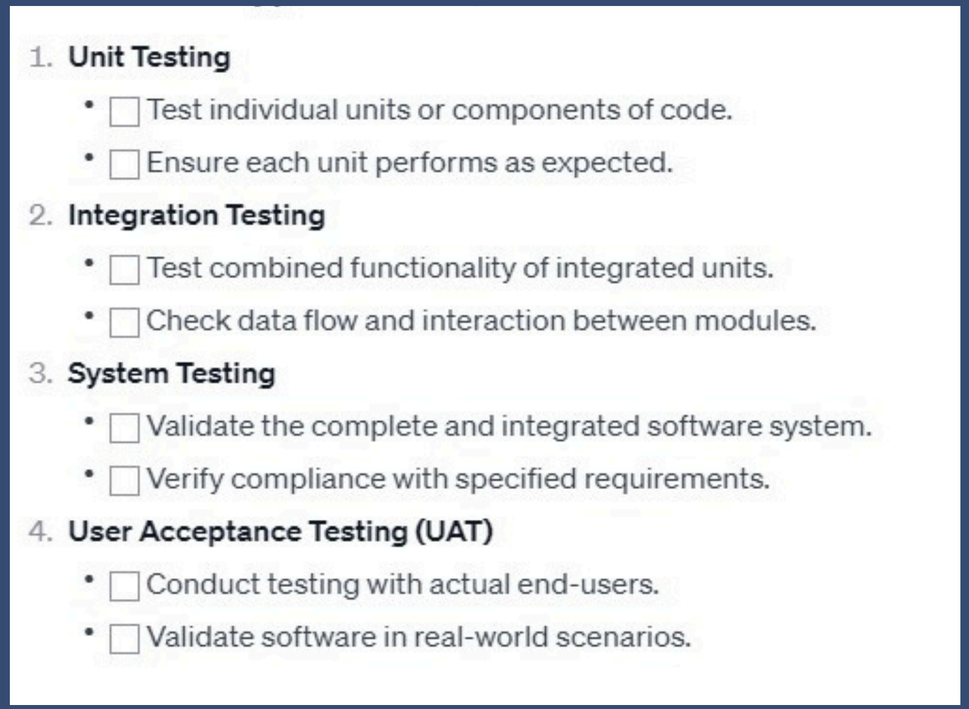


# Теоретические основы тестирования компонентов

Тестирование — это систематический процесс снижения рисков отказа.

Компонент проверяется по его контракту: входные данные, выходные значения, ошибки.

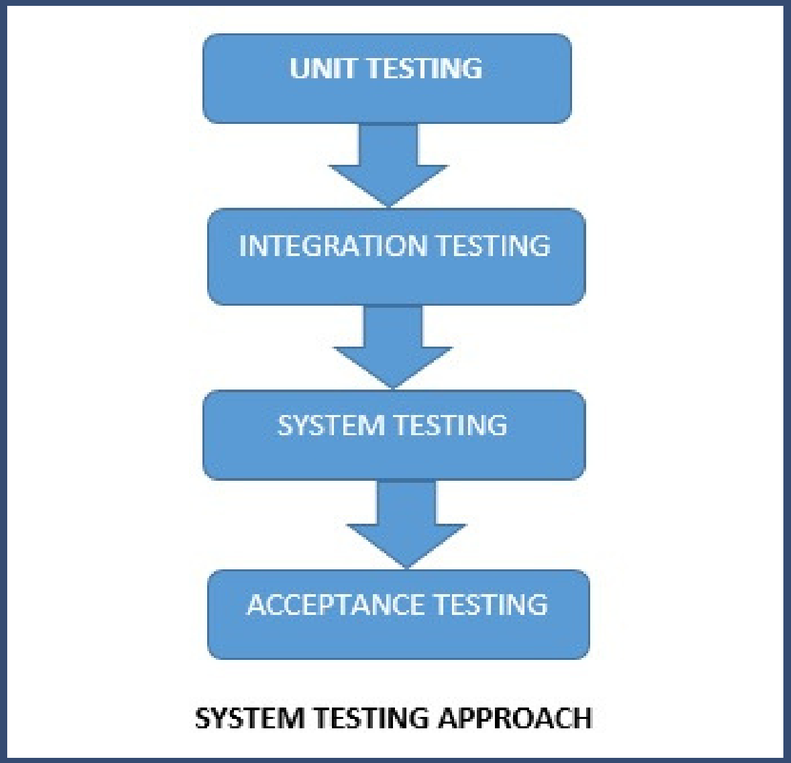
На практике важны как типичные, так и граничные случаи.



1. **Unit Testing**
   - ☐ Test individual units or components of code.
   - ☐ Ensure each unit performs as expected.
2. **Integration Testing**
   - ☐ Test combined functionality of integrated units.
   - ☐ Check data flow and interaction between modules.
3. **System Testing**
   - ☐ Validate the complete and integrated software system.
   - ☐ Verify compliance with specified requirements.
4. **User Acceptance Testing (UAT)**
   - ☐ Conduct testing with actual end-users.
   - ☐ Validate software in real-world scenarios.

# Уровни и виды тестирования♘

- Компонентное → проверка отдельного модуля
- Интеграционное → проверка взаимодействия компонентов
- Системное и приёмочное → проверка всей системы
- Подходы: «чёрный ящик», «белый ящик», «серый ящик»



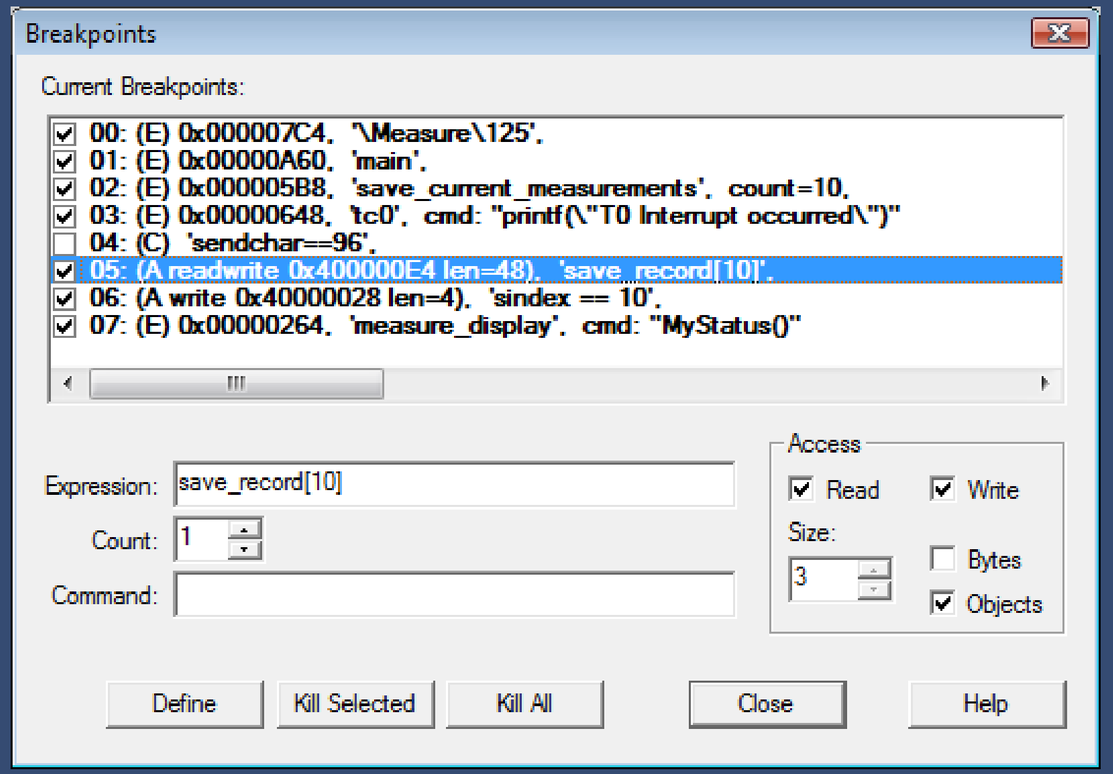
# Методы отладки ♗

Пошаговая отладка в IDE позволяет исследовать ход выполнения программы.

Логирование и трассировка помогают разбирать проблемы в распределённых системах.

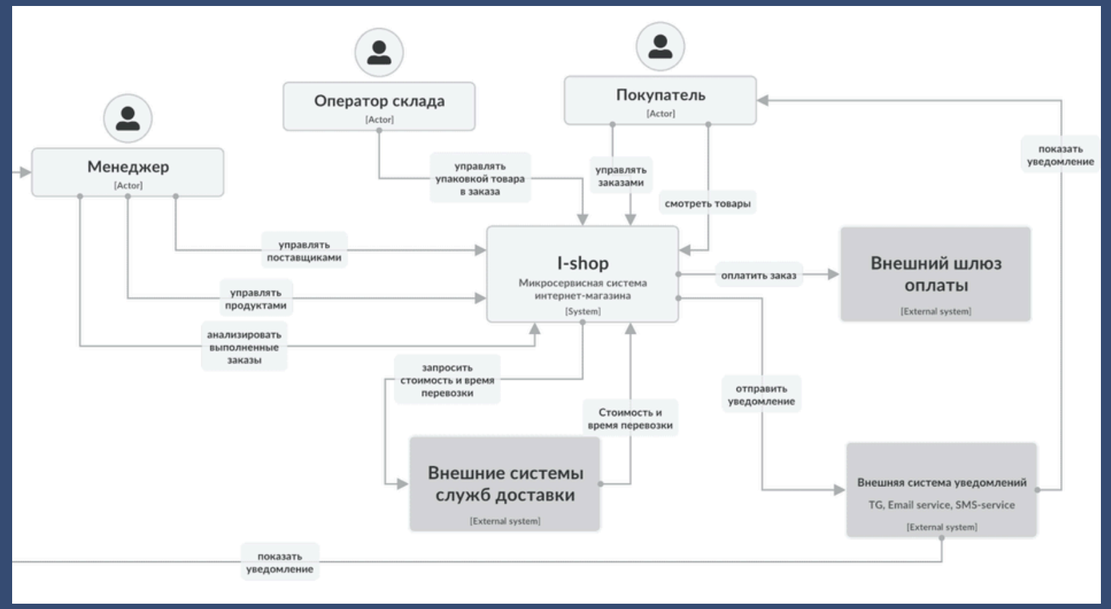
Профилирование выявляет узкие места по времени и ресурсам.

Анализ дампов используется при критических сбоях.



# Подходы к интеграции компонентов

- Монолит: вызовы модулей внутри процесса
- Микросервисы: REST, gRPC, очереди сообщений
- Стратегии интеграции: сверху вниз, снизу вверх, смешанная
- Влияние CI/CD: автоматическая сборка и проверка

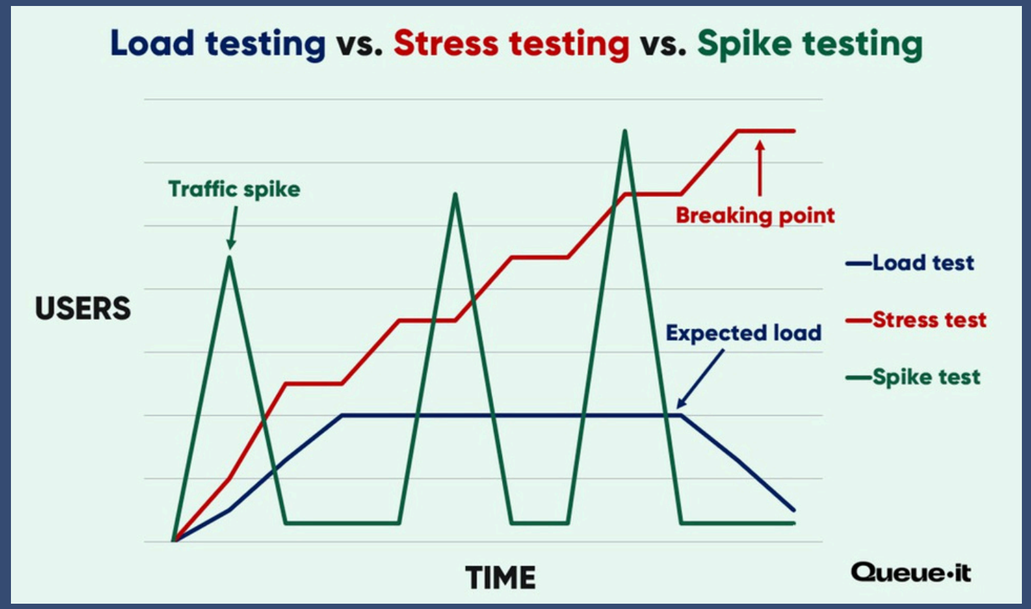


# Тестирование интегрированной системы ♖

End-to-end тесты проверяют полный бизнес-сценарий «от начала до конца».

Нагрузочные и стресс-тесты оценивают поведение под высокой нагрузкой.

Отказоустойчивость и безопасность проверяются в условиях сбоев и угроз.



# Итог

*Практические рекомендации и выводы*

Автоматизация компонентных и интеграционных тестов должна быть частью CI/CD.

Структурированное логирование и мониторинг упрощают отладку в эксплуатации.

Явные контракты API и версионирование снижают риски интеграционных дефектов.

Совместное применение этих практик повышает надёжность системы.