



РАЗРАБОТКА ИС

Жизненный цикл ИС

- **Определение 1:** Жизненный цикл ИС — это процесс ее построения и развития.
- **Определение 2:** Жизненный цикл ИС — период времени, который начинается с момента принятия решения о необходимости создания информационной системы и заканчивается в момент ее полного изъятия из эксплуатации.
- Стандарт ГОСТ 34.601-90
- Стандарт ISO/IEC 12207:1995

Стандарт ГОСТ 34.601-90

1. Формирование требований к ИС
2. Разработка концепции ИС
3. Составление ТЗ
4. Эскизное проектирование
5. Технический проект
6. Рабочая документация
7. Ввод в действие
8. Сопровождение АС

Стандарт ISO/IEC 12207:1995

- Стандарт «Information Technology — Software Life Cycle Processes» определяет структуру жизненного цикла, содержащую процессы, действия и задачи, которые должны быть выполнены во время создания ИС.
- Каждый процесс разделен на набор действий, каждое действие — на набор задач. Каждый процесс, действие или задача инициируется и выполняется другим процессом по мере необходимости, причем не существует заранее определенных последовательностей выполнения.

Процессы жизненного цикла ИС по ISO

- *Основные:*
 - Приобретение
 - Поставка
 - Разработка
 - Эксплуатация
 - Сопровождение
- *Организационные*
 - Управление
 - Создание инфраструктуры
 - Усовершенствование
 - Обучение
- *Вспомогательные*
 - Документирование
 - Управление конфигурацией
 - Обеспечение качества
 - Верификация
 - Аттестация
 - Совместная оценка
 - Аудит
 - Разрешение проблем

Модели жизненного цикла

- Модель жизненного цикла ИС — структура, определяющая последовательность выполнения и взаимосвязи процессов, действий и задач на протяжении жизненного цикла.
- Модель жизненного цикла зависит от специфики, масштаба и сложности проекта и специфики условий, в которых система создается и функционирует.
- Стандарт ГОСТ Р и ISO не предлагает конкретную модель жизненного цикла.

I. Каскадная модель (модель водопада)

Следуя модели водопада, разработчик переходит от одной стадии к другой строго последовательно.

1. Анализ (определение требований)
2. Проектирование
3. Реализация (конструирование, кодирование)
4. Тестирование и отладка
5. Внедрение (инсталляция)
6. Сопровождение

1 Анализ

- Анализ требований — это процесс сбора требований к ПО, их систематизации, документирования, анализа, выявления противоречий, неполноты, разрешения конфликтов в процессе разработки ПО.
- Анализ требований включает три типа деятельности:
 - Сбор требований: общение с клиентами и пользователями, чтобы определить, каковы их требования.
 - Анализ требований: определение, являются ли собранные требования неясными, неполными, неоднозначными, или противоречащими, и затем решение этих проблем.
 - Документирование требований: Требования могут быть задокументированы в различных формах, таких как простое описание, сценарии использования, пользовательские истории, или спецификации процессов.

2 Проектирование ПО

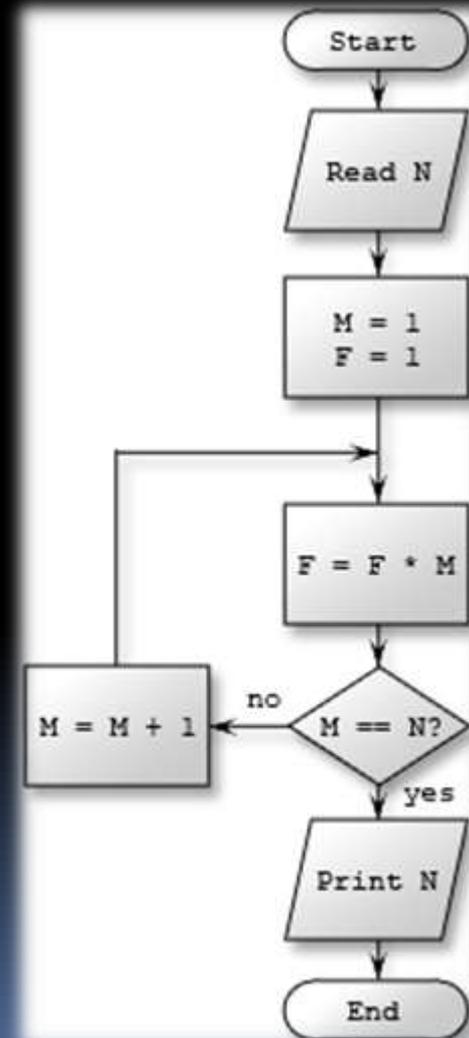
- Проектирование ПО — процесс создания проекта ПО, а также дисциплина, изучающая методы проектирования.
- Проектирование подразумевает выработку свойств системы на основе анализа постановки задачи, а именно: моделей предметной области, требований к ПО, а также опыта проектировщика.
- Модель предметной области накладывает ограничения на бизнес-логику и структуры данных.

Нотации при проектировании ПО

- В процессе проектирования ПО для выражения его характеристик используются различные нотации :
 - Блок схемы
 - ER-диаграммы
 - UML-диаграммы
 - DFD-диаграммы
 - Макеты

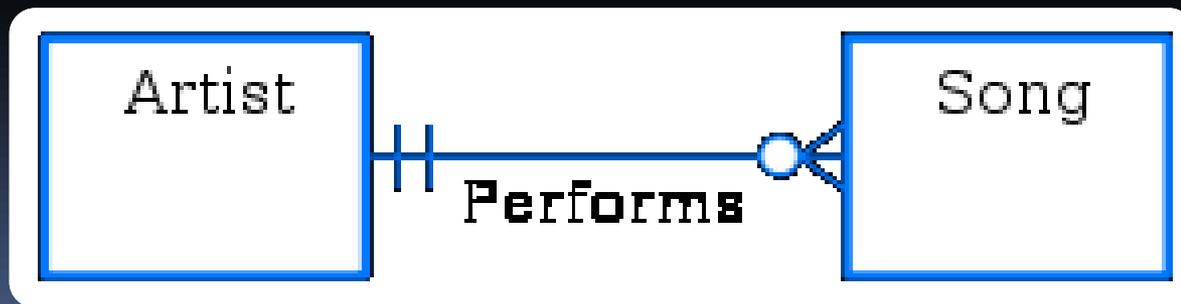
Блок схемы

- Блок-схема — распространенный тип схем, описывающий алгоритмы или процессы, изображая шаги в виде блоков различной формы, соединенных между собой стрелками.



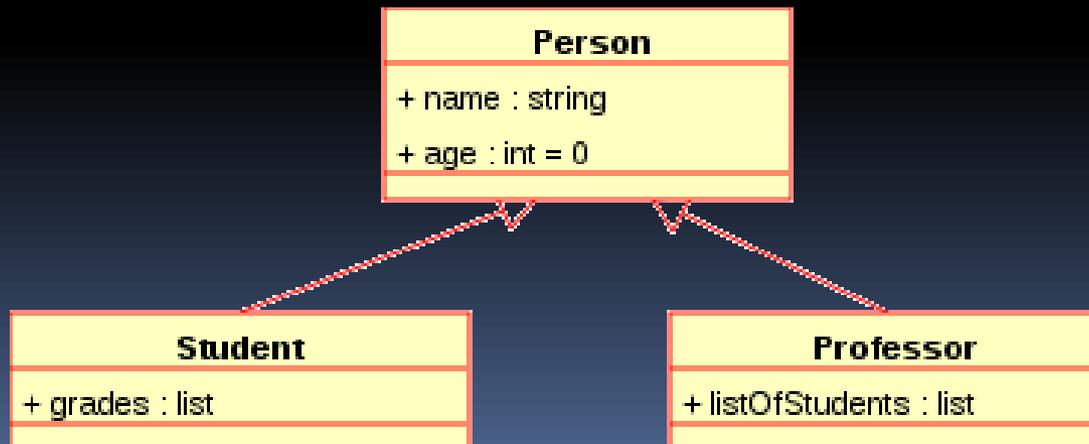
ER диаграммы

- Модель Сущность-Связь (ER-модель) — модель данных, позволяющая описывать концептуальные схемы. Предоставляет собой графическую нотацию, основанную на блоках и соединяющих их линиях, с помощью которых можно описывать объекты и отношения между ними какой-либо другой модели данных.



UML - диаграммы

- UML (англ. Unified Modeling Language — унифицированный язык моделирования) — язык графического описания для объектного моделирования в области разработки программного обеспечения.



DFD – диаграммы потоков данных

DFD (от англ. Data Flow Diagrams) — методология графического структурного анализа, описывающая внешние по отношению к системе источники и адресаты данных, логические функции, потоки данных и хранилища данных, к которым осуществляется доступ.



3 Реализация (конструирование) ИС

- Реализация ИС заключается в кодировании алгоритмов на заданном языке программирования, либо в создание БД в СУБД.

4 Тестирование

- Тестирование ПО — процесс исследования ПО с целью получения информации о качестве продукта.
- Функциональное тестирование — это тестирование ПО в целях проверки реализуемости функциональных требований, то есть способности ПО в определённых условиях решать задачи, нужные пользователям.
- Тестирование производительности — в инженерии программного обеспечения тестирование, которое проводится с целью определения, как быстро работает система или её часть под определенной нагрузкой.
- Юзабилити-тестирование — эксперимент, выполняемый с целью определения, насколько хорошо люди могут использовать ПО (БД, web-сайт и т.п.)
- Тестирование безопасности — оценка уязвимости ПО к различным атакам.
- Тестирование совместимости — метод, основной целью которого является обеспечение качественной работы конечного продукта с другим ПО.



Внедрение

- Внедрение ПО — процесс настройки программного обеспечения под определенные условия использования, а также обучения пользователей работе с программным продуктом.
- 

II. Спиральная модель проектирования ПО

- При использовании этой модели ИС создается в несколько итераций (витков спирали) методом прототипирования.
- Прототип — действующий компонент ИС, реализующий отдельные функции и внешние интерфейсы. Каждая итерация соответствует созданию фрагмента или версии ИС, на ней уточняются цели и характеристики проекта, оценивается качество полученных результатов и планируются работы следующей итерации.

III. Итерационная модель

- Итеративный подход — выполнение работ параллельно с непрерывным анализом полученных результатов и корректировкой предыдущих этапов работы. Проект при этом подходе в каждой фазе развития проходит повторяющийся цикл:
Планирование — Реализация — Проверка — Оценка



Стратегии и методы проектирования ПО

- Стратегия проектирования сверху-вниз
- Стратегия проектирования снизу-вверх
- Объектно-ориентированный подход
- Функциональное-ориентированное (структурное) проектирование
- Проектирование на основе структур данных
- Компонентное проектирование

Вертикальные стратегии

- При разработке ПО используются два подхода: проектирование сверху вниз, при котором разработка приложения начинается с определения основных функций и задач, и проектирование снизу вверх, при котором сначала проводится анализ данных и определение их структуры.
- Проектирование снизу вверх. Метод основан на создании базовых простейших элементов, на основе которых строятся более сложные.
- Проектирование сверху вниз. В этом методе сначала создается структура программы. Каждый элемент представлен моделью «черного ящика». Далее детально прорабатывается каждый элемент.

Структурное проектирование и на основе структур данных

- Структурное проектирование - метод проектирования, в котором декомпозиция сфокусирована на идентификации основных программных функций и, затем, детальной разработке и уточнении этих функций “сверху-вниз”
- При проектировании на основе структур данных фокус сконцентрирован в большей степени на структурах данных, которыми управляет система, чем на функциях системы.



Компонентное проектирование

- Программные компоненты являются независимыми единицами, которые обладают однозначно-определенными интерфейсами и зависимостями (связями) и могут собираться и развертываться независимо друг от друга.
 - Цель такого подхода заключается в повышении эффективности повторного использования разработанных компонент.
- 