

Лекция №1

Теоретические основы управления жизненным циклом информационных систем

УЧЕБНЫЕ ВОПРОСЫ:

1. Ведение в управление жизненным циклом ИС.
2. Жизненный цикл ИС и ее модели.
3. Требования к информационной системе и модели жизненного цикла.

Вопрос 1.
Ведение в управление
жизненным циклом ИС.

Цель дисциплины: получение студентами знаний о методах и средствах управления жизненным циклом информационных систем (ИС), основанных на CASE-технологиях, а также формирование навыков их самостоятельного применения при управлении жизненным циклом ИС в сфере экономики и управления.

Задачи дисциплины:

формирование целостного представления об основных моделях, методах и средствах управления жизненным циклом информационных систем в сфере экономики и управления;

овладение практическими навыками в использовании технологий управления жизненным циклом ИС в сфере экономики и управления;

формирование умений решения задач анализа, управления требованиями и конфигурациями, тестирования, выполнением проекта и документирования ИС в сфере экономики и управления, в том числе с применением современных программных комплексов.

Изучается дисциплина в 8 семестре и предусматривает:

Лекций 8 часов, Лабораторных занятий 10 часов,
Практических занятий 10 часов, самостоятельная работа 44 часа. Форма отчетности зачет.

Вопрос 2.
Жизненный цикл ИС и ее
модели.

Жизненный цикл ИС можно представить как ряд событий, происходящих с системой в процессе ее создания и использования.

Модель жизненного цикла отражает различные состояния системы, начиная с момента возникновения необходимости в данной ИС и заканчивая моментом ее полного выхода из употребления.

Модель жизненного цикла - структура, содержащая процессы, действия и задачи, которые осуществляются в ходе разработки, функционирования и сопровождения программного продукта в течение всей жизни системы, от определения требований до завершения ее использования.

Модели жизненного цикла:

Каскадная модель предусматривает последовательное выполнение всех этапов проекта в строго фиксированном порядке. Переход на следующий этап означает полное завершение работ на предыдущем этапе.

Поэтапная модель с промежуточным контролем. Разработка ИС ведется итерациями с циклами обратной связи между этапами. Межэтапные корректировки позволяют учитывать реально существующее взаимовлияние результатов разработки на различных этапах; время жизни каждого из этапов растягивается на весь период разработки.

Спиральная модель На каждом витке спирали выполняется создание очередной версии продукта, уточняются требования проекта, определяется его качество и планируются работы следующего витка. Особое внимание уделяется начальным этапам разработки - анализу и проектированию, где реализуемость тех или иных технических решений проверяется и обосновывается посредством создания прототипов (макетирования).

Каскадная модель жизненного цикла



Можно выделить следующие **положительные стороны** применения каскадного подхода:

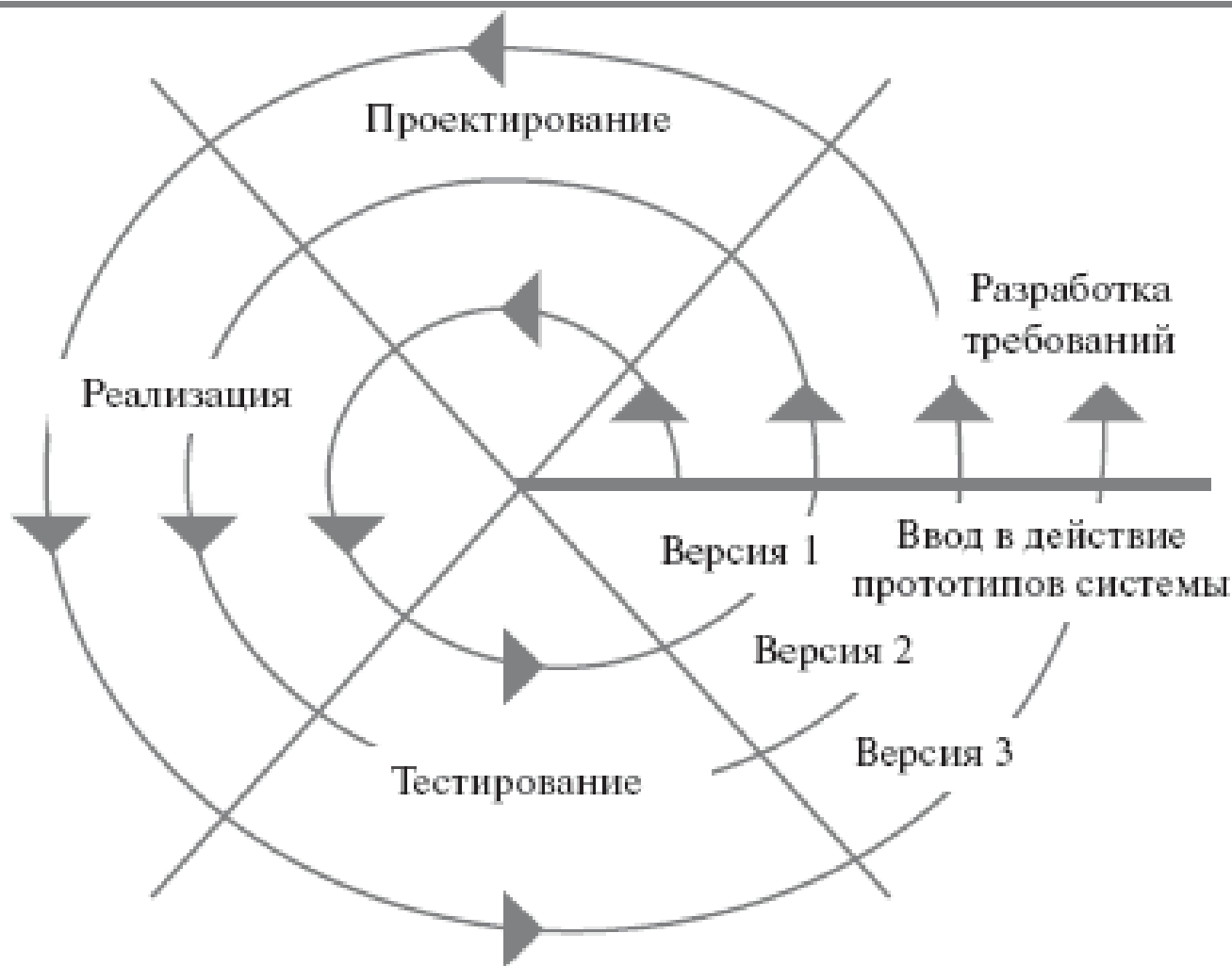
- на каждом этапе формируется законченный набор проектной документации, отвечающий критериям полноты и согласованности;
- выполняемые в логической последовательности этапы работ позволяют планировать сроки завершения всех работ и соответствующие затраты.

Основным недостатком этого подхода является то, что реальный процесс создания системы никогда полностью не укладывается в такую жесткую схему, постоянно возникает потребность в возврате к предыдущим этапам и уточнении или пересмотре ранее принятых решений.

Поэтапная модель с промежуточным контролем



Спиральная модель жизненного цикла



Основная **проблема** спирального цикла - определение момента перехода на следующий этап. Для ее решения вводятся временные ограничения на каждый из этапов *жизненного цикла*, и переход осуществляется в соответствии с планом, даже если не вся запланированная работа закончена.

На практике наибольшее распространение получили две основные *модели жизненного цикла*:

каскадная модель (характерна для периода 1970-1985 гг.);
спиральная модель (характерна для периода после 1986.г.).

Основные причины, по которым *каскадная модель* сохраняет свою популярность, следующие:

Привычка - многие ИТ-специалисты получали образование в то время, когда изучалась только *каскадная модель*, поэтому она используется ими и в наши дни.

Иллюзия снижения рисков участников проекта (заказчика и исполнителя). *Каскадная модель* предполагает разработку законченных продуктов на каждом этапе: технического задания, технического проекта, программного продукта и пользовательской документации. Разработанная документация позволяет не только определить требования к продукту следующего этапа, но и определить обязанности сторон, объем работ и сроки, при этом окончательная оценка сроков и стоимости проекта производится на начальных этапах, после завершения обследования.

Проблемы внедрения при использовании итерационной модели. В некоторых областях *спиральная модель* не может применяться, поскольку невозможно использование/тестирование продукта, обладающего неполной функциональностью (например, военные разработки, атомная энергетика и т.д.).

Среди наиболее известных стандартов можно выделить следующие:

ГОСТ 34.601-90 - распространяется на автоматизированные системы и устанавливает стадии и этапы их создания. Кроме того, в стандарте содержится описание содержания работ на каждом этапе. Стадии и этапы работы, закрепленные в стандарте, в большей степени соответствуют *каскадной модели* жизненного цикла.

- ISO/IEC 12207:1995 - стандарт на процессы и организацию *жизненного цикла*. Распространяется на все виды заказного ПО. Стандарт не содержит описания фаз, стадий и этапов.

- Custom Development Method (методика Oracle) по разработке прикладных информационных систем - технологический материал, детализированный до уровня заготовок проектных документов, рассчитанных на использование в проектах с применением Oracle. Применяется CDM для классической модели ЖЦ Rational Unified Process (RUP) предлагает итеративную модель разработки, включающую четыре фазы: начало, исследование, построение и внедрение

- Microsoft Solution Framework (MSF) сходна с RUP, так же включает четыре фазы: анализ, проектирование, разработка, стабилизация, является итерационной, предполагает использование объектно-ориентированного моделирования.

- Extreme Programming (XP). Экстремальное программирование (самая новая среди рассматриваемых методологий) сформировалось в 1996 году. В основе методологии командная работа, эффективная коммуникация между заказчиком и исполнителем в течение всего проекта по разработке ИС, а разработка ведется с использованием последовательно дорабатываемых прототипов.

В соответствии с базовым международным стандартом ISO/IEC 12207 все *процессы ЖЦ ПО* делятся на три группы:

1. Основные процессы:

- приобретение;
- поставка;
- разработка;
- эксплуатация;
- сопровождение.

2.Вспомогательные процессы:

- документирование;
- управление конфигурацией;
- обеспечение качества;
- разрешение проблем;
- аудит;
- аттестация;
- совместная оценка;
- верификация.

3.Организационные процессы:

- создание инфраструктуры;
- управление;
- обучение;
- усовершенствование.

Вопрос 3.
Требования к информационной
системе и модели жизненного
цикла.

Основные компоненты ИС включают в себя:
программное обеспечение,
информационное обеспечение, технические средства,
обслуживающий персонал.

Необходимость определения требований к информационной системе возникает в следующих случаях:

в момент выбора новой информационной системы,
при подготовке тендерной документации,
при заключении договора на разработку или настройку выбранной информационной системы,
при уточнении (детализации) потребностей бизнеса в процессе разработки или настройки системы, а так же при необходимости внесения изменений в систему в ходе эксплуатации.

На уровень детализации, область определения, а так же используемые методы описания влияют:

выбранная модель жизненного цикла разработки и внедрения;

характера разрабатываемого и внедряемого ПО (заказная разработка, настройка информационной системы).

используемые средств и методов проектирования (в случае заказной разработки).

В зависимости от этих параметров следует использовать ту или иную методологию моделирования, которая определяет выбор нотации, а выбор нотации, в свою очередь, определяет используемые инструментальные средства.

Функциональные требования к информационной системе, которые описываются, в том числе, и с помощью моделей процессов и структур данных, являются только частью общих требований, которые содержатся в техническом задании.

Раздел требований к информационной системе технического задания может содержать следующие подразделы:

требования к функциональным характеристикам

требования к надежности

настраиваемость

условия эксплуатации

требования к информационной и программной совместимости

требования к документации

Требования к функциональным характеристикам

Требования к надежности

В разделе должны быть определены требования к обеспечению надежного функционирования: контроль входной и выходной информации, время и механизмы восстановления после программных и аппаратных отказов. В этом разделе описывается организация системы безопасности, включая подсистемы контроля доступа, шифрования и т. п.

Настраиваемость

Определяются требования к адаптационным возможностям ПО, то есть указывается, какие изменения в методах управления и бизнес процессах должны быть предусмотрены.

Условия эксплуатации.

В этом разделе описывается необходимое обслуживание, которое требуется для работы системы, например, создание резервных копий, реиндексирование баз и т. п., а так же требования к квалификации персонала (пользователей и обслуживающего персонала).

Требования к составу и параметрам технических средств

Указывается необходимый состав технических средств с указанием их основных технических характеристик. Могут указываться требования к помещениям, в которых будет находиться оборудование. В этом разделе указываются требования к переносимости системы.

Требования к информационной и программной совместимости

Требования к информационным структурам на входе и выходе, методам решения, исходным кодам, языкам программирования и программным средствам, используемым программой.

Требования к программной документации

В этом разделе указывается предварительный состав программной документации, и при необходимости, специальные требования к ней.