***18.01.2022***

***Лекция №1***

**Жизненный цикл ПО. Стадии жизненного цикла.**

**Жизненный цикл ПО(ЖЦПО)** — это непрерывный процесс, начинающийся с момента принятия решения о необходимости создания ПО, и заканчивающийся в момент его полного изъятия из эксплуатации.

Жизненный цикл ИС представляет собой модель ее создания и использования. Можно отражает различные состояния ИС, начиная с момента возникновения необходимости в данной ИС, и заканчивая моментом ее полного выхода из употребления у всех пользователей.

**Основные этапы ЖЦПО:**

1. Анализ требований

2. Проектирование

3. Кодирование (программирование)

4. Тестирование и отладка

5. Эксплуатация и сопровождение

**Жизненный цикл коммерческих программ определяется началом продаж**

**Падение продаж и интереса является сигналом к:**

1. Изменению программного продукта и услуг

2. Изменение цены на них

3. Проведению модификации или снятию с продажи и предоставления

**Основной НД регламентирующий ЖЦПО, это стандарт ISO/IEC 12207**

Данный стандарт определяет структуру жизненного цикла, содержащую процессы, действия и задачи, выполняемые во время создания программного обеспечения.

**Структура ЖЦПО:**

1. Основные процессы

(приобретение, поставка, разработка, эксплуатация, сопровождение)

2. Вспомогательные процессы(документирование, управление конфигурацией, обеспечение качества, верификация, аттестация, оценка, аудит, решение проблем)

3. Организационные процессы(управление проектами, создание инфраструктуры проекта, определение, оценка и улучшение самого жц, обучение)

Разработка ПО - это как правило, анализ, проектирование и реализация (программирование). Она включает все работы по созданию ПО и его компонент в соответствии с заданными требованиямии, в том числе оформление проектной и эксплуатационной документации, подготовку материалов, необходимых для проверки работоспособности и соответствующего качества программных продуктов, материалов, для организации обучения персонала и т. Д.

Эксплуатация включает работы по внедрению компонентов ПО в эксплуатацию, в том числе конфигурирование БД и рабочих мест пользователей, обеспечение эксплуатационной документацией проведение обучение персонала и т. д.

Управление проектом связано с вопросами планирования и организации работ, создания коллективов разработчиков; контроля за сроками и качеством выполняемых работ.

Техническое и организационное обеспечение проекта включает выбор методов и инструментальных средств для реализации проекта, определение методов описания промежуточных состояний разработки, разработку методов и средств испытаний ПО, обучение персонала и т. п.

Обеспечение качества проекта связано с проблемами верификации, проверки и тестирования ПО.

Верификация – это процесс определения насколько текущее состояния разработки, достигнутое на данном этапе, отвечает требованиям этого типа.

Оценка качества (ГОСТ 28195-89) осуществляется на всех этапах жизненного цикла программных средств (ПС) при:

* Планировании показателей качества ПС;
* Контроле качества на отдельных этапах разработки (техническое задание, технический проект, рабочий проект);
* Контроле качества в процессе производства ПС;
* Проверке эффективности модификации ПС в процессе сопровождения.

Проверка позволяет оценить соответствие параметров разработки с исходными требованиями. Проверка частично совпадает с тестированием, связанным с выявлением различий между действительными и ожидаемыми результатами а также оценкой соответствия характеристик ПО исходным требованиям.

Под тестированием понимается процесс исполнения программы с целью обнаружения ошибок.

В процессе реализации проекта важное место занимают вопросы идентификации, описания и контроля конфигурации отдельных компонентов и всей системы в целом.

Управление конфигурацией – один из вспомогательных процессов, поддерживающих основные процессы жизненного цикла ПО, прежде всего процессы разработки и сопровождения ПО ИС.

Управление конфигурацией позволяет организовать, систематически учитывать и контролировать внесение изменений в ПО на всех стадиях ЖЦ. Общие принципы и рекомендации конфигурационного учёта, планирования и управления конфигурациями ПО отражены в проекте стандарта ISO/IEC 12207. Жизненный цикл информационных продуктов и услуг составляет основу жизненного цикла информационных технологий и, соответственно, информационных систем. Следовательно, всё сказанное выше относится и к информационным системам.

***19.01.2022***

***Лекция №2***

**Процессы ЖЦ ПО. Основные процессы**

ГОСТ Р ИСО/МЭК 12207-2010 – «Информационная технология. Системная и программная инженерия. Процессы жизненного цикла программных средств.»

Является отечественным аналогом ISO/IEC 12207.

**Процессы ЖЦ ПО**

1. Основные

* Приобретение
* Поставка
* Разработка
* Эксплуатация
* Сопровождение

1. Вспомогательные

* Документирование
* Управление конфигурацией
* Обеспечение качества
* Верификация
* Аттестация
* Совместная оценка
* Аудит
* Разрешение проблем

1. Организационные
   * Управление
   * Усовершенствование
   * Создание инфраструктуры

**Основные процессы ЖЦ ПО**

***Процесс приобретения*** определяет действия предприятия-покупателя, которое приобретает программный продукт

Данный процесс охватывает действия:

1. Инициирование приобретения (задачи: определение заказчиком своих потребностей в приобретении программного продукта; анализ требований к системе; принятие решения относительно приобретения; проверку наличия необходимой документации, гарантий, сертификатов, лицензий и поддержки в случае приобретения программного обеспечения; подготовку и утверждения плана приобретения, включающего требования к системе, тип договора, ответственность сторон).

***25.01.2022***

1. Подготовку заявочных предложений;
2. Подготовку и корректировку договора;
3. Надзор за деятельностью поставщика;
4. Приемку и завершение работ

*Процесс поставки* охватывает действия и задачи, выполняемые поставщиком, который снабжает заказчика программным продуктом или услугой.

Данный процесс включает следующие действия:

1. Инициирование поставки;
2. Планирование.

*Процесс разработки* определяет действия предприятия-разработчика, включает в себя стратегическое планирование, анализ, проектирование и реализацию (программирование), т.е. все работы по созданию программного обеспечения и его компонентов в соответствии с заданными требованиями, а также оформление проектной и эксплуатационной документации, подготовку материалов, необходимых для тестирования разработанных программных продуктов и обучения персонала.

Разработка является одним из важнейших процессов ЖЦ ПО и охватывает следующие этапы:

1) подготовительная работа;

2) анализ требований к системе;

3) проектирование архитектуры системы;

4) анализ требований к ПО;

5) проектирование архитектуры ПО;

6) детальное проектирование ПО;

7) кодирование и тестирование ПО;

8) интеграция ПО;

9) квалификационное тестирование ПО;

10) интеграция системы;

11) квалификационное тестирование системы;

12) установка ПО;

13) приемка ПО.

*Квалификационное тестирование* – набор критериев и условий, которые необходимо выполнить, чтобы квалифицировать программный продукт как соответствующий своим спецификациям и готовый к использованию в условиях эксплуатации.

*Приемка ПО* предусматривает оценку результатов квалификационного тестирования программного обеспечения и системы и документирование результатов оценки, которые проводятся заказчиком при помощи разработчика. Разработчик выполняет окончательную передачу ПО заказчику в соответствии с договором, обеспечивая при этом необходимое обучение и поддержку.

Процесс эксплуатации определяет действия персонала, обеспечивающие обслуживание программного обеспечения в процессе его функционирования в интересах пользователей. Основные эксплуатационные работы включают непосредственно эксплуатацию, локализацию проблем и устранение причин их возникновения, модификацию программного обеспечения, подготовку предложений по совершенствованию системы, развитие и модернизацию системы. В процесс эксплуатации входят также конфигурирование базы данных и рабочих мест пользователей, обеспечение пользователей эксплуатационной документацией, обучение персонала.

Процесс эксплуатации включает в себя ряд действий:

1) подготовительную работу;

2) эксплуатационное тестирование;

3) эксплуатацию системы:

4) поддержку пользователей.

Процесс сопровождения определяет действия персонала, обеспечивающие сопровождение программного продукта, что представляет собой управление модификациями программного продукта, поддержку ею текущего состояния и функциональную пригодность, а также включает в себя инсталляцию и удаление программного обеспечения. В процессе сопровождения в программное обеспечение вносят необходимые изменения, которые могут потребовать пересмотра проектных решений, принятых на любом этапе жизненного цикла. Изменения, вносимые в существующее программное обеспечение, не должны нарушать его целостность. Процесс сопровождения включает в себя перенос программного обеспечения и другую среду (миграцию) и заканчивается снятием программного обеспечения с эксплуатации.

Процесс сопровождения охватывает следующие действия:

1. Подготовительная работа
2. Анализ проблем и запросов на модификацию ПО;
3. Модификация ПО;
4. Проверка и приемка;
5. Перенос ПО в другую среду;
6. Снятие ПО с эксплуатации.

**Вспомогательные процессы ЖЦ ПО**

Вспомогательные процессы поддерживают реализацию основных процессов, будучи неотъемлемой частью всего жизненного цикла информационной системы, с определенной целью и обеспечивают должное качество информационной системы. Вспомогательные процессы используются и выполняются по мере необходимости и инициируются другими процессами. Вспомогательные процессы – это:

1. Документирование;
2. Управление конфигурацией;
3. Обеспечение качества;
4. Верификация;
5. Аттестация;
6. Совместная оценка;
7. Аудит;
8. Разрешение проблем.

Процесс документирования предусматривает формализованное описание информации, созданной в течение жизненного цикла программного обеспечения. Данный процесс состоит из набора работ, с помощью которых планируют, проектируют, разрабатывают, выпускают, редактируют, распространяют и сопровождают те документы, в которых нуждаются все заинтересованные лица, такие как администраторы, инженеры и пользователи системы или программного продукта.

Процесс документирования включает в себя действия:

- подготовительную работу

- проектирование и разработку;

- выпуск документации;

- сопровождение.

*Процесс управление конфигурацией* определяет действия по управлению конфигурацией. Под конфигурацией программного обеспечения понимается совокупность ее функциональных и физических характеристик, установленных в технической документации и реализованных в программном обеспечении. Управление конфигурацией позволяет организовать, систематические учитывать и контролировать внесение изменений в программное обеспечение на всех стадиях жизненного цикла программного обеспечения. Это тот вспомогательный процесс, который поддерживает основные процессы жизненного цикла, прежде всего процессы разработки и сопровождения.

Процесс управления конфигураций включает в себя действия:

- подготовительную работу;

- идентификацию конфигурации;

- контроль конфигурации;

- учет состояния конфигурации;

- оценку конфигурации;

- управление выпуском и поставку;

*Процесс обеспечения качества* определяет действия для объективной гарантии, что информационная система и процессы соответствуют определенным требованиям к ним и придерживаются установленным замыслам.

*Процесс верификации* определяет действия (для покупателя, поставщика или независимой стороны) для верификации программного обеспечения с различной глубиной зависимости от проекта.

Верификация в узком смысле означает формальное доказательство правильности программного обеспечения.

Целью верификации является достижение гарантии того, что верифицируемый объект (требования или программный код) соответствует требованиям, реализован без непредусмотренных функций и удовлетворяет проектным спецификациям и стандартам.

Процесс верификации включает в себя инспекции, тестирование кода, анализ результатов тестирования, формирование и анализ отчетов о проблемах. Таким образом, принято считать, что процесс тестирования является составной частью процесса верификации.

*Процесс аттестации* определяет действия (покупателя, поставщика, независимой стороны) для аттестации программного обеспечения. Под аттестацией обычно понимаются подтверждение и оценка достоверности проведенного тестирования. Аттестация должна гарантировать полное соответствие ПО спецификациям, требованиям и документации, а также возможность его безопасного и надежного применения пользователем. Аттестацию рекомендуется выполнять путем тестирования во всех возможных ситуациях и использовать при этом независимых специалистов. Аттестация может проводиться на начальных стадиях жизненного цикла программного обеспечения или как часть работы по приемке программного обеспечения.

*Процесс совместной оценки* предназначен для оценки состояния работ по проекту и ПО. Он сосредоточен в основном на контроле планирования и управления ресурсами, персоналом, аппаратурой и инструментальными средствами проекта.

Оценка применяется как на уровне управления проектом, так и на уровне технической реализации проекта и проводится в течение всего срока договора. Данный процесс может выполняться двумя любыми сторонами, участвующими в договоре, при этом одна сторона проверяет другую.

Процесс совместной оценки включает в себя действия:

- подготовительную работу;

- оценку управления проектом;

- техническую оценку

*Процесс аудита* представляет собой определение соответствия требованиям, планам и условиям договора. Аудит может выполняться двумя любыми сторонами, участвующими в договоре, когда одна сторона проверяет другую.

Аудит – ревизия(проверка), проводимая компетентным органом (лицом) в целях обеспечения независимой оценки степени соответствия ПО или процессов установленным требованиям.

Аудит служит для установления соответствия реальных работ и отчетов требованиям, планам и контракту. Аудиторы не должны иметь прямой зависимости от разработчиков программного обеспечения. Они определяют состояния работ, использование ресурсов, соответствие документации требованиям и стандартам, корректность тестирования.

*Процесс решения проблем* определяет процесс анализа и устранения проблем (включая несоответствия), какова бы ни была их природа или источник, которые были обнаружены на протяжении разработки, эксплуатации, сопровождения или других процессов. Каждая обнаруженная проблема должна быть идентифицирована, описана, проанализирована и разрешена.

**Организационные процессы жизненного цикла программного обеспечения**

К организационным процессам жизненного цикла программного обеспечения относят:

1. Управление;
2. Усовершенствование;
3. Создание инфраструктуры;
4. Обучение.

*Процесс управления* состоит из действий и задач, которые могут выполняться любой стороной, управляющей своими процессами. Данная сторона (менеджер) отвечает за управление выпуском продукта, управление проектом и управление задачами соответствующих процессов, таких как приобретение, поставка, разработка, эксплуатация, сопровождение и др.

Процесс управления включает в себя следующие действия:

- инициирование и определение области управления – менеджер должен убедиться, что необходимые для управления ресурсы (персонал, оборудование и технология) имеются в его распоряжении в достаточном количестве;

- планирование – составление графиков работ, оценку затрат, выделение требуемых ресурсов, распределение (ответственности, оценку рисков, связанных с конкретными задачами, создание инфраструктуры управления);

- выполнение и контроль;

- проверку и оценку;

- завершение.

*Процесс создание инфраструктуры* охватывает выбор и поддержку (сопровождение) технологии, стандартов и инструментальных средств, выбор и установку аппаратных и программных средств, используемых для разработки, эксплуатации или сопровождения программного обеспечения.

*Процесс совершенствования* предусматривает оценку, измерение, контроль и усовершенствование процессов жизненного цикла ПО.

*Процесс обучения* охватывает первоначальное обучение и последующее постоянно повышение квалификации персонала. Приобретение, поставка, разработка, эксплуатация и сопровождение ПО в значительной степени зависит от уровня знаний и квалификации персонала.

***26.01.2022***

***Лекция №3***

Требования – это условие или характеристика, которой должно удовлетворять ПО

Первичные требования заказчика (ПТЗ) включают в себя протоколы совещаний, интервью с заказчиками и пользователями, копии и оригиналы различных документов, сведения о существовании аналогичных программных продуктов и другие материалы.

**ПТЗ не должны противоречить друг другу,** не должно быть лишних документов, не содержащих полезную информацию

Следует вести учет собираемых требований и контролировать их обработку, оценку и реализацию или отказ от реализации.

Такая работа называется *работой по управлению требованиями*

*Управление требованиями* (**requirements management)** представляет собой:

1. Систематический подход к выявлению организации и документированию требований к ПО
2. Процесс, устанавливающий соглашения между заказчиками и разработчиками относительно изменения требований к ПО

Цели данной работы:

1. Достижение соглашения с заказчиком и пользователями о том, что должен делать программный продукт.
2. Улучшение понимания требований к ПО со стороны разработчиков.
3. Установление границ программного обеспечения
4. Определение базиса для планирования

При разработке устанавливаются приоритеты, разделяя все требования на три категории:

1. «Необходимо выполнить»
2. «Следует выполнить»
3. «Можно выполнить»

Виды требований:

1. Функциональные ― определяют действия, которые должно выполнять программное обеспечение, без учета ограничений, связанных с его реализацией.
2. Не функциональные ― не определяют поведение программного продукта, но описывают его атрибуты или атрибуты системного окружения. Их типы:
   1. Требования к применению
   2. Требования к производительности
   3. Требования к реализации
   4. Требования к надежности
   5. Требования к интерфейсу

Этап формирования требований включает в себя анализ первичных требований заказчика, структурирование требований и, возможно, конструирование типа.

Составление требований является циклическим процессом.

*Анализ и структурирование ПТЗ заказчика.*

Анализ ПТЗ должен проводиться обязательно совместно с заказчиком. При анализе ПТЗ необходимо выяснить возможно ли создание требуемого ПО и насколько реальны указанные сроки, а также каких деталей в описании не хватает и что еще следует прояснить.

Кроме того, необходимо определить сроки и способы получения недостающей информации, способы тестирования и условия приемки готового ПП (программного продукта).

**Структурирование требований** выполняется:

1. Указывают общее назначение программного продукта, особенности аппаратных средств на которых ПО должно быть реализовано.
2. Определяются особенности эксплуатации программного продукта.
3. Выделяют основные компоненты ПО
4. Выделяют детализацию обобщенной структуры требований

Завершается структурирование требований к программному обеспечению составлением описаний отедльных функций каждого структурного элемента. Для каждой фукнции должны быть указаны входные данные, а также способы тестирования.

**Анализ и моделирование предметной области**

Под моделью предметной области понимается некоторая система, имитирующая структуру или функционирование исследуемой предметной области и отвечающая основному требованию – быть адекватной этой области.

К модели предметной области предъявляются следующие требования:

* Формализация, обеспечивающая однозначное описание структуры предметной области;
* Понятность для заказчиков и разработчиков на основе применения графических средств отображения модели
* Реализуемость, подразумевающая наличие средств физической реализации модели предметно области;
* Обеспечение возможности оценки эффективности реализации модели предметной области на основе определенных методов и вычисляемых показателей;

Для реализации перечисленных требований необходимо построение системы моделей:

* Объектной модели, отражающей состав взаимодействующих в процессах материальных и информационных объектов предметной области;
* Функциональной модели, отражающей взаимосвязь функций (действий) по преобразованию объектов в процессах;
* Технической модели, описывающей топологию расположения и способы коммуникации комплекса технических средств.

Главный критерий адекватности модели предметной области заключается в функциональной полноте разрабатываемой ИС.

С моделированием непосредственно связана проблема выбора языка представления проектных решений

Язык моделирования – графическая нотация, которая используется для описания проектов.

Нотация представляет собой совокупность графических объектов, используемых в модели, и является синтаксисом языка моделирования.

Обычно модели строятся на трех уровнях:

1. На внешнем уровне (определение требований);
2. Концептуальном уровне (спецификации требований);
3. Внутреннем уровне (реализация требований).

Так, на внешнем уровне определяется состав основных компонентов программного обеспечения: объектов, функций, событий, организационных единиц, технических средств.

На концептуальном уровне определяется характер взаимодействия компонентов системы.

На внутреннем уровне при помощи модели дается ответ на вопрос: «С помощью каких программно-технических средств реализуются требования к системе?».

**Составление спецификаций по требованиям заказчика**

Спецификациями называют точно формализованное описание функций и ограничений разрабатываемого программного продукта.

Функциональная спецификация в разработке программного обеспечения ― документ, описывающий требуемые характеристики системы (функциональность).

Эксплуатационная спецификация ― описание правил использования ПО. Эксплуатационные спецификации определяют требования к техническим средствам, надежности, безопасности и т.д. Совокупность спецификаций представляет собой общую логическую модель проектируемого программного обеспечения.

Функциональные спецификации должны быть полными и точными.

Функциональная спецификация состоит из трех частей:

1. Описание внешней информационной среды, с которой будет взаимодействовать разрабатываемое программное обеспечение. Должны быть определены все используемые каналы ввода и вывода и все информационные объекты , к которым будет применяться разрабатываемое ПО, а также существенные связи между этими информационными объектами.
2. Определение функций программного обеспечения, определенных на множестве состояний этой информационной среды. Вводятся обозначения всех определяемых функций, специфицируются их входные данные и результаты выполнения, с указанием типов данных и заданий всех ограничений, которым должны удовлетворять эти данные и результаты. Определяется содержание каждой их этих функций.
3. Описание исключительных ситуаций, если таковые могут возникнуть при выполнении программ, и реакций на эти ситуации, которые должны обеспечить соответствующие программы. Должны быть перечислены все существенные случаи, когда программное обеспечение не сможет нормально выполнить ту или иную свою функцию. Для каждого такого случая должна быть определена реакция программы.

Все функциональные спецификации разрабатываемого программного обеспечения описывают перечень функций и состав обрабатываемых данных. Они различаются только системой приоритетов (акцентов), которая используется разработчиком в процессе анализа требований и определения спецификаций.

**Конструирование прототипа**

Разработка прототипа (прототипирование), по мнению некоторых разработчиков, является одним из самых важных этапов разработки программного обеспечения.

Можно сказать, что прототипирование ― быстрая «черновая» реализация базовой функциональности для анализа работы системы в целом.

Создаваемая на этапе прототипирования программная система может работать неэффективно, с ошибками и не в полной мере. Кроме того, прототипирование необязательно должно осуществляться с использованием тех же технологий, что и разрабатываемая система.

Прототипы могут быть горизонтальные и вертикальные, одноразовые и эволюционные, бумажные и электронные.

Процесс создания прототипа состоит из следующих шагов:

1. Определение начальных требований
2. Разработка первого варианта прототипа, который содержит только пользовательский интерфейс системы;
3. Этап изучения прототипа заказчиком и конечным пользователем. Получение обратной связи о необходимых изменениях и дополнениях;
4. Переработка прототипа с учетом полученных замечаний и предложений.

Прототип позволяет также получить обратную связь от будущих пользователей, причем именно тогда, когда это наиболее необходимо: в начале проекта еще есть возможность исправить ошибки проектирования практически без потерь.

Прототипирование также уменьшает общую стоимость разработки системы. По этим причинам оно часто используется в процессе разработки требований.

**Основные сведения о технологии проектирования ПО**

Задачей этапа проектирования является определение подробных спецификаций разрабатываемого программного обеспечения. Процесс проектирования программного обеспечения обычно включает:

1. Проектирование общей структуры ― определение основных частей (компонентов) и их взаимосвязей по управлению и данным;
2. Декомпозицию компонентов и построение структурных иерархий в соответствии с рекомендациями блочно-иерархического подхода;
3. Проектирование компонентов.

Результатом проектирования является детальная модель разрабатываемого программного обеспечения вместе со спецификациями его компонентов всех уровней.

Принято различать также два аспекта проектирования:

1. Логическое проектирование, включающее в себя те проектные операции, которые непосредственно не зависят от имеющихся технических и программных средств, составляющих среду функционирования будущего программного продукта.
2. Физическое проектирование, которое заключается в привязке к конкретным техническим и программным средствам среды функционирования.

Технология проектирования определяет