**1. СТБ ИСО/МЭК 12207-2003. Процессы жизненного цикла (ЖЦ) программных средств (ПС) и их классификация. Общие сведения.**

**СТБ ИСО/МЭК 12207-2003 – Информационная технология – Процессы жизненного цикла программных средств** (в РБ с **2004**). Данный стандарт является аутентичным аналогом международного стандарта **ISO/IEC 12207: 1995**.

Данный стандарт определяет жизненный цикл ПС и систем в целом. Однако стандарт охватывает процессы ЖЦ системы только в части ее ПС и не определяет процессы ЖЦ для ее остальных компонентов. В соответствии с данным стандартом ЖЦ ПС и систем имеет **трехуровневую иерархическую структуру**. Основу ЖЦ составляет набор **процессов**. Каждый процесс разделен на набор **работ**. Каждая работа разделена на набор **задач**.

Процессы ЖЦ ПС делятся на **три группы**:

5. основные;

6. вспомогательные;

7. организационные.

Нумерация приведена в соответствии со структурой стандарта.

**Основные процессы ЖЦ** – это процессы, которые реализуются под управлением основных сторон, участвующих в ЖЦ ПС. Состоят из **пяти процессов:** заказ; поставка; разработка; эксплуатация; сопровождение.

**Вспомогательные процессы ЖЦ** – это процессы, являющиеся целенаправленными составными частями других процессов. Состоят из **восьми процессов:** документирование, управление конфигурацией, обеспечение качества, верификация, аттестация, совместный анализ, аудит, решение проблем.

**Организационные процессы ЖЦ** – это процессы, предназначенные для создания в некоторой организации и совершенствования организационных структур, охватывающих процессы ЖЦ и соответствующий персонал. К ним относятся **четыре процесса**: управление; создание инфраструктуры; усовершенствование; обучение.

Следует обратить внимание на то, что взаимоотношения между процессами, определяемые в *СТБ ИСО/МЭК 12207-2003*, всегда статические. В реальной же жизни отношения между процессами и участниками программного проекта являются динамическими. Каждый процесс и выполняющая его организация включаются в проект **уникальным образом**.

1. Процесс заказа и заказчик – при определении системы, которая должна содержать программный продукт(ПП).

2. Процесс поставки и поставщик – при предоставлении программного продукта или услуги, от которых зависит система.

3. Процесс разработки и разработчик – при анализе системы для корректного выделения и определения ПП, при его разработке и обеспечении подключения к системе.

4. Процесс эксплуатации и оператор – при эксплуатации ПП в системной среде в интересах пользователя.

5. Процесс сопровождения и персонал сопровождения – при сопровождении и поддержке ПП в эксплуатационной готовности и для консультации пользователей.

6. Вспомогательный или организационный процесс – при необходимости обеспечения уникальных специализированных функций для других процессов.

**2.СТБ ИСО/МЭК 12207-2003. Основные и организационные процессы жизненного цикла программных средств.**

**Процессы жизненного цикла программных средств** (в РБ с **2004**). Данный стандарт является аутентичным аналогом международного стандарта **ISO/IEC 12207: 1995**.

Процессы ЖЦ ПС делятся на **три группы**:

5. основные;

6. вспомогательные;

7. организационные.

Нумерация приведена в соответствии со структурой стандарта.

**Основные процессы ЖЦ ПС:**

**1. Заказ** (**5 работ:** подготовка процесса заказа, подготовка заявки на подряд, подготовка и корректировка договора, надзор за поставщиком, приемка и закрытие договора)

**2.Поставка** (**5 работ:** подготовка процесса заказа, подготовка заявки на подряд, подготовка и корректировка договора, надзор за поставщиком, приемка и закрытие договора.)

**3.Разработка** (**13 работ:**подготовка процесса разработки, анализ требований к системе, проектирование системной архитектуры, анализ требований к ПС, проектирование программной архитектуры, техническое проектирование ПС, программирование и тестирование ПС, сборка ПС, квалификационные испытания ПС, сборка системы, квалификационные испытания системы, ввод в действие ПС, обеспечение приёмки ПС.)

**4.Эксплуатация (4 работы:**подготовка процесса эксплуатации, эксплуатационные испытания, эксплуатация системы, поддержка пользователя)

**5.Сопровождение** (**6 работ:**подготовка процесса сопровождения, анализ проблем и изменений, внесение изменений, проверка и приемка при сопровождении. Перенос. Снятие с эксплуатации)

**Организационные процессы ЖЦ ПС:**

**1.Управление** (**5 работ:** подготовка и определение области управления, планирование, выполнение и контроль, проверка и оценка, завершение)

**2.Процесс создания инфраструктуры** (**3 работы:** подготовка процесса создания инфраструктуры, создание инфраструктуры, сопровождение инфраструктуры)

**3. Процесс усовершенствования** (**3 работы:** создание процесса, оценка процесса, усовершенствование процесса)

**4. Процесс обучения** (**3 работы:** подготовка процесса обучения, разработка учебных материалов, реализация плана обучения)

**№3: СТБ ИСО/МЭК 12207-2003. Процесс разработки и его структура (работы 1-6).**

Основные процессы ЖЦ ПС включают в себя процесс разработки, который включает в себя 13 работ. Определяет работы и задачи разработчика. Общее число задач – 55.

1. **Подготовка процесса разработки:** Определение модели ЖЦ ПС со структурированием в неё процессов работ и задач СТБ/ИСО МЭК 12207-2003 (если модель не определена в договоре).

Документальное оформление выходных результатов (ВР), управление конфигурацией ВР, решение возникающих проблем, выполнение вспомогательных процессов в соответствии с процессами группы 6.

Выбор и адаптация стандартов, методов, инструментариев, языков программирования.

Разработка и выполнение планов проведения работ процесса разработки.

Поставка всех комплектующих изделий.

2. **Анализ требований к системе:** Анализ области применения системы и определение требований к ней. Оценка требований к системе.

3. **Проектирование архитектуры системы:** Определение общей архитектуры системы, распределение требований к ней между объектами технических и программных средств архитектуры и ручными операциями и дальнейшее уточнение требований.

Оценка архитектуры системы и требований к ее объектам.

4. **Анализ требований к программным средствам**: Определение требований к ПС, их оценка. Проведение совместных анализов в соответствии с процессом совместного анализа.

5. **Проектирование программной архитектуры:** Преобразование требований к программному объекту (ПО) в его архитектуру, их распределение между его компонентами и дальнейшее уточнение требований (разработка эскизного проекта).

Разработка эскизного проекта интерфейсов ПО и его компонентов.

Разработка эскизного проекта базы данных.

Разработка предварительных версий документации пользователя.

Разработка предварительных требований к испытаниям (тестированию) ПО и графика сборки программного продукта.

Оценка архитектуры ПО и эскизных проектов интерфейсов и базы данных по критериям.

Проведение совместных анализов в соответствии с процессом совместного анализа.

6. **Техническое проектирование программного средства:**

Разработка технического проекта для компонентов ПО (представление их в виде программных модулей), распределение технических требований к компонентам между программными модулями и дальнейшее уточнение требований.

Разработка технического проекта интерфейсов ПО и его компонентов.

Разработка технического проекта базы данных.

Уточнение документации пользователя.

Разработка требований к испытаниям и программе испытаний программных модулей.

Уточнение общих требований к испытанию (тестированию) и программе сборки программных средств.

Оценка технического проекта и требований к тестированию по критериям.

Проведение совместных анализов в соответствии с процессом совместного анализа.

**№4: СТБ ИСО/МЭК 12207-2003. Процесс разработки и его структура (работы 7-13).**

7. **Программирование и тестирование программных средств:** Программирование программных модулей, базы данных, разработка процедур и данных для их испытаний (тестирования).

Тестирование программных модулей и базы данных.

Уточнение документации пользователя.

программных модулей.

Уточнение общих требований к испытанию (тестированию) и программе сборки программного средства.

Оценка результатов программирования и тестирования по критериям.

8. **Сборка программных средств:** Разработка плана сборки для объединения программных модулей и компонентов в ПО.

Сборка программных модулей и компонентов и тестирование результатов сборки.

Уточнение документации пользователя.

Разработка плана и процедуры квалификационных испытаний программных средств.

Оценка планов и результатов сборки и тестирования, а также документации пользователя по критериям.

Проведение совместных анализов в соответствии с процессом совместного анализа.

9. **Квалификационные испытания программных средств:** Проведение квалификационных испытаний программных средств.

Уточнение документации пользователя.

Оценка результатов испытаний и документации пользователя по критериям.

Обеспечение проведения аудиторской проверки в соответствии с процессом аудита.

Доработка и подготовка программного продукта к последующим работам, определение состояния конфигурации (базовой линии) проекта.

10. **Сборка системы:** Сборка объектов программной и технической конфигурации, ручных операций, других систем в единую систему, испытания собранной системы.

Разработка плана и процедуры квалификационных испытаний системы.

Оценка собранной системы по критериям.

11. **Квалификационные испытания системы:** Проведение квалификационных испытаний системы.

Оценка системы по критериям.

Обеспечение проведения аудиторской проверки в соответствии с процессом аудита.

Доработка и подготовка программного продукта к последующим работам, определение состояния конфигурации (базовой линии) проекта.

12. **Ввод в действие программных средств:** Разработка плана ввода в действие программного продукта в среде эксплуатации.

Ввод в действие программного продукта в соответствии с планом.

13. **Обеспечение приемки программных средств:** Обеспечение проведения заказчиком приёмочных испытаний.

Укомплектование и поставка программного продукта заказчику на условиях договора.

Обеспечение обучения и поддержка персонала заказчика на условиях договора.

**5. СТБ ИСО/МЭК 12207-2003. Вспомогательные процессы жизненного цикла программных средств.**

**Вспомогательные процессы ЖЦ**– это процессы, являющиеся целенаправленными составными частями других процессов. Состоят из **восьми процессов:** документирование, управление конфигурацией, обеспечение качества, верификация, аттестация, совместный анализ, аудит, решение проблем.

Их основное назначение – обеспечить успешную реализацию и качество выполнения программного проекта. Вспомогательный процесс инициируется и используется другим процессом.

Для управления качеством программных средств в ходе жизненного цикла служат процессы, выделенный серым цветом на рисунке. При этом процессы верификации, аттестации, совместного анализа и аудита могут реализовываться различными сторонами независимо или использоваться как методы процесса обеспечения качества. Ответственность за работы и задачи вспомогательного процесса несет организация, выполняющая данный процесс.

**Процесс документирования** – (идет параллельно разработке) предназначен для формализованного описания информации, созданной в процессе или работе ЖЦ. Он включает: Планирование, Проектирование, Разработка, Выпуск, Редактирование, Распространение, Сопровождение документов по программному проекту

**Процесс управления конфигурацией** – предназначен для определения состояния программных объектов в системе управления их изменения и выпуска.

**Процесс обеспечения качества** – предназначен для обеспечения гарантии того, что ПП и процессы в ЖЦ проекта соответствуют требованиям и планам.

**Процесс верификации** – предназначен для определения соответствия функционирования ПП требованиям и условиям, реализованным в предшествующих работах.

**Процесс аттестации** – предназначен для определения полноты соответствия установленных требований созданной системы или ПП их функциональному назначению.

**Процесс совместного анализа** – предназначен для оценки состояния и результатов работ по проекту. Данный процесс может выполняться 2-мя сторонами, участвующими в договоре.

**Процесс аудита** – предназначен для определения соответствия требованиям планов и условия договора.

**Процесс решения проблем** – предназначен для анализа и решения проблем, к/т обнаружены в ходе выполнения разработки, эксплуатации, сопровождения и др. процессов.

**6. ISO/IEC 12207: 2008. Процессы соглашения и повторного применения программных средств.**

ISO/IEC 12207:2008 системная и программная инженерия — Процессы жизненного цикла программных средств — стандарт ISO, описывающий процессы жизненного цикла программного обеспечения.

Данный стандарт, используя устоявшуюся терминологию, устанавливает общую структуру процессов жизненного цикла программных средств, на которую можно ориентироваться в программной индустрии. Стандарт определяет процессы, виды деятельности и задачи, которые используются при приобретении программного продукта или услуги, а также при поставке, разработке, применении по назначению, сопровождении и прекращении применения программных продуктов.

Процессы ЖЦ:

1) Процессы в контексте системы:

1. Процессы соглашения — 2;

* Поставка
* Приобретение

2. Процессы организационного обеспечения проекта — 5;

3. Процессы проекта — 7;

4. Технические процессы — 11;

2) Специальные процессы:

5. Процессы реализации программных средств — 7;

6. Процессы поддержки программных средств — 8;

7. Процессы повторного применения программных средств — 3.

* Процесс проектирования доменов;
* Процесс менеджмента повторного применения активов;
* Процесс менеджмента повторного применения программ.

**Процессы соглашения** – определяют работы, необходимые для достижения договоренностей между 2-мя организациями. Соглашения могут разрабатываться:

1. Между поставщиком и разработчиком

2. Между заказчиком и разработчиком

· **Процесс приобретения** предназначен дляполучения продукта, который удовлетворяет потребностям, приобретающей стороны.

· **Процесс поставки** предназначен для предоставления приобретающей стороне продукта, удовлетворяющего установленным требованиям.

**Процессы повторного применения ПС** – предназначены для поддержки возможности повторного использования программных элементов в различных проектах, выполняемых в организации.

**7. ISO/IEC 12207: 2008. Процессы организационного обеспечения проекта.**

**Процессы организационного обеспечения проекта** управляют возможностью организации приобретать или поставлять продукты или услуги с помощью инициализации, поддержки и управления проектами. Эти процессы обеспечивают ресурсы и инфраструктуру, необходимую для поддержки проектов и гарантируют достижения организационных целей и имеющихся договоренностей.

Процессы организационного обеспечения проекта

1. Процесс менеджмента модели жизненного цикла;(определение сопровождения процессов ЖЦ моделей ЖЦ, относящихся к области действия должного стандарта для использования организации)

2. Процесс менеджмента инфраструктуры;(обеспечение услугами в течении ЖЦ)

3. Процесс менеджмента портфеля проектов;(поддержка необход.достаточ.количества проектов)

4. Процесс менеджмента людских ресурсов;(поддержка сотрудников)

5. Процесс менеджмента качества (гарантия того, что продукты ,услуги и реализация процесса ЖЦ соответствует качеству и удовлетворяет треб заказчика)

**8. ISO/IEC 12207: 2008. Процессы проекта.**

**Процессы проекта** связаны с планированием, оценкой и управлением проекта.

1) **Процессы менеджмента проекта** используются для планирования, выполнения, оценки и управления хода проекта. Эти процессы применяются для создания и отслеживания планов проекта, для оценки фактического выполнения хода проекта, для управления выполнением проекта до его завершения.

1. Процесс планирования проекта;(разработка и взаимная увязывание эффектив. И осуществ.планов проектов)

2. Процесс управления и оценки проекта.(опред.состояние проекта и гарантир,того,что проект выполняется в соответствии с планом предела бюджета и он удовлетворяет технически)

2) **Процессы поддержки проекта** поддерживают специализированные цели управления; они необходимы при управлении любым проектом от его организации в целом и до отдельных процессов жизненного цикла.

1. Процесс менеджмента решений;(непрерыв идентифин анализ,разрешения и мониторинг рисков)

2. Процесс менеджмента рисков;(непрерывное идентифин,анализ,разрешения и мониторинг рисков)

3. Процесс менеджмента конфигурации;(поддержка цености всех индентифицир выходных результатов проектов и обеспечение их допустимости заинтресованным сторонам )

4. Процесс менеджмента информации;(обеспечение заинтресова сторон значимой своевременной полной правильной и если необходимо конфидициональной информацией в начале и конце жц пс)

5. Процесс измерений.(сбор анализ и описание данных связанных с разраб продуктами для поддержки)

**9. ISO/IEC 12207: 2008.Технические процессы.**

Предназначены для определения требований к системе, преобр. треб. к системе в эффект. продукт, эксплуатации продукта, ликвидации продута при его снятии к эксплуатации. Эти процессы обеспечивают возможность достижения высоких характеристик качества продукта, его доступность, своевременность и результативность затрат.

**1. Процесс определения требований правообладателя** (цель-определить требования к системе которые позволяют обеспечить предоставление услуг необходимых пользователям или другим правообладателям в заданной среде окружения)

2. **Процесс анализа системных требований** (цель-преобразование сформированных требований правообладателя в набор технических требований к системе)

3. **Процесс проектирования архитектуры системы** (распределение системных требований межу элементами системы)

4. **Процесс реализации** (цель-реализация заданного элемента системы)

5. **Процесс сборки системы** ( цель-объединение системных элементов для создания полной системы удовлетворяющей проекту системы и ожидаемых результатов потребителя)

6. **Процесс квалификационного тестирования** (цель-подтверждение того что реализация каждого системного элемента протестирована на соответствие и готова к поставке)

7. **Процесс инсталляции ПС** (цель- установка программного продукта соответствующего заданным требованиям в целевой среде окружения)

8. **Процесс поддержки приемки ПС** ( цель- содействие заказчика в достижение уверенности что продукт соответствует требованиям)

9. **Процесс эксплуатации** (цель-эксплуатация программного продукта в заданной среде окружения и обеспечение поддержки потребителя)

10. **Процесс сопровождения** (цель- обеспечение эффективной по затратам поддержки поставляемого программного продукта)

11. **Процесс снятия ПС с эксплуатации** (цель: завершение существования системного программного элемента. Инсталлирует и выводит из работы программный продукт в среде окружения, оставляя среду в приемлемом состоянии.)

**10. ISO/IEC 12207: 2008. Процессы реализации программных средств.**

Используются для изготовления заданного элемента системы реализованного в виде ПС. Эти процессы преобразуют заданные свойства интерфейса и ограничения на реализованные действия по разработке.

Результат этих действий – системный элемент удовлетворяет соответствующим требованиям. Процессы 2-7 являются процессами нижнего уровня.

Работа стратегия реализации ПС (осуществляется выбор или разработка модели ЖЦ, в данную модель встраивают стадии разработки, выбираются работы и задачи, адаптация стандартов, методы, языки программирования и т.д.)

1. **Процесс анализа требования к ПС** (установка требований к программным элементам системы)

2. **Процесс проектирования архитектуры** (создание проекта реализации ПС который может быть верифицирован по отношению к требованиям)

3. **Процесс детального проектирования** (создание проекта реализации ПС который может быть верифицирован по отношению к требованиям и архитектуре ПС и достаточно детализирован чтобы позволить дальнейшее тестирование и кодирование)

4. **Процесс конструирования ПС** (создание исполнимых программных модулей соответствующих плану ПС)

5. **Процесс сборки ПС** (сборка программных модулей и компонентов полученных при сборке в соответствии с проектов ПС и демонстрация того что функциональные и нефункциональные требования к ПС удовлетворяются в эквивалентной или заданной среде окружения)

6. **Процесс квалификационного тестирования** (доказательство того что собранный программный продукт соответствует заданным требованиям)

**11. Качество программных средств. Основные понятия и определения (атрибут, качество, метрика, модель качества, характеристика качества программных средств, подхарактеристика качества программных средств, ошибка, отказ, ранжирование, уровень ранжирования, шкала).**

На процесс разработки и деятельность по оценке качества ПС оказывают влияние следующие обобщённые показатели ПС:

1. Область применения и назначение ПС;

2. Тип решаемых задач;

3. Объём и сложность ПС;

4. Необходимый состав и требуемые значения характеристик качества ПС и величина допустимого ущерба из-за их недостаточного качества;

5. Степень связи решаемых задач с реальным масштабом времени или допустимой длительностью эксплуатации и перспектива создания множества версий ПС;

6. Предполагаемый тираж производства и применения ПС;

7. Степень необходимости документированности ПС.

**Атрибут** – измеримое физическое или абстрактное свойство продукта. Могут быть внешними или внутренними.

**Качество** – совокупность характеристик программного продукта, относящихся к его способности удовлетворять установленным и подразумеваемым потребностям.

**Метрика** – метод и шкала измерения. Могут быть внутренними, внешними или метриками качества в использовании; прямыми или косвенными. Включают методы для категоризации качественных данных.

**Модель качества** – набор характеристик и связей между ними обеспечивающих основу для определения требований к качеству и для оценки качества.

**Характеристика качества программных средств** – набор свойств программного средства, с помощью которых описывается и оценивается его качество.

**Подхарактеристика качества программных средств** – характеристика качества программного средства, входящая в состав другой характеристики.

**Ошибка** – некорректный шаг процесс или определение данных в программе.

**Отказ** – прекращение способности продукта выполнять требуемую функцию или его неспособность работать в пределах заданных ограничений.

**Ранжирование** – действие по отнесение измеренного значения к соответствующему уровню ранжирования.

**Уровень ранжирования** – точка на порядковой школе которая используется для категории шкалы измерения.

**Шкала** – набор значений с определенными свойствами.

**12. Типы шкал. Примеры шкал.**

**Шкала** – набор значений с определёнными свойствами.

При оценке качества ПС обычно используется 4 типа шкал: номинальная, порядковая (упорядоченная),интервальная, шкала отношений.

1. **Номинальная шкала** соответствует набору категорий. Целью является классификация измеренных атрибутов, упорядочивание значений не предполагается. **Пример:** классификация дефектов ПС по их типам без упорядочивания.

2. **Порядковый тип шкал:** соответствует упорядоченному набору значений шкалы. Целью является назначение порядка или ранга измеренным атрибутам. Обычно содержит небольшое кол-во значений. **Пример:** упорядочивание программного продукта по серьёзности.

3. **Интервальный тип шкалы:** соответствует упорядоченной шкале с равноудаленными делениями; обычно содержит достаточно большое количество делений с количественными значениями. **Пример:** шкала с делениями (0,1,2,…,10)

4. **Относительная шкала:** соответствует упорядоченной шкале с равноудаленными делениями, оцененными в относительных единицах относительно некоторой абсолютной величины (обычно в диапазоне от 0 до 1)

Два первых типа шкал применяются для оценки качественных атрибутов ПС, которые нельзя измерить количественно, и для ранжирования измеренных значение, третий и четвертый типы – для оценки количественных атрибутов.

**13. СТБ ИСО/МЭК 9126-2003. Модель качества программных средств (характеристики и подхарактеристики).**

В соответствии со стандартом 28806 и 9126 модель качества ПС имеет трехуровневую структуру. На верхнем уровне 6 характеристик качеств, каждая характеристика делится на подхарктеристики (2 уровень), на 3 уровне метрики.

1. **Функциональность** (Functionality) – совокупность свойств ПС, определяемая наличием и конкретными особенностями набора функций, способных удовлетворять заданные или подразумеваемые потребности.

1. пригодность

2. правильность

3. способность к взаимодействию

4. согласованность

5. защищённость

2. **Надежность** (Reliability) – совокупность свойств, характеризующая способность ПС сохранять заданный уровень пригодности в заданных условиях в течение заданного интервала времени.

1. завершённость

2. устойчивость к ошибке

3. восстанавливаемость

3. **Удобство использования** (практичность, Usability) – совокупность свойств программного средства, характеризующая усилия, необходимые для его использования, и индивидуальную оценку результатов его использования заданным или подразумеваемым кругом пользователей.

1. понятность

2. обучаемость

3. простота использования

4. **Эффективность** (Efficiency) – совокупность свойств программного средства, характеризующая те аспекты его уровня пригодности, которые связаны с характером и временем использования ресурсов, необходимых при заданных условиях функционирования.

1. времяёмкость

2. ресурсоёмкость

5. **Сопровождаемость** (Maintainability) – совокупность свойств программного средства, характеризующая усилия, которые необходимы для его модификации.

1. анализируемось

2. изменяемость

3. стабильность

4. тестируемость

6. **Мобильность** (Portability) – совокупность свойств программного средства, характеризующая приспособленность для переноса из одной среды функционирования в другие.

1. адаптируемость

2. простота установки

3. соответствие

4.взаимозаменяемость

**14. СТБ ИСО/МЭК 9126-2003. Метод оценки качества программных средств.**

Определяет метод оценки качества ПС, основанный на трёхуровневой иерархической модели качества. На верхнем уровне 6 характеристик качеств, каждая характеристика делится на подхарктеристики (2 уровень), на 3 уровне метрики.



Процесс оценки состоит из трех стадий: определение требований к качеству ПС, подготовка к оцениванию и процедура оценивания. Данный процесс может применяться после любой подходящей работы жизненного цикла для каждого компонента программного продукта.

**Определение требований к качеству ПС**

Цель: установка требований в терминах характеристик и подхарактеристик качества.

**Подготовка к оцениванию**

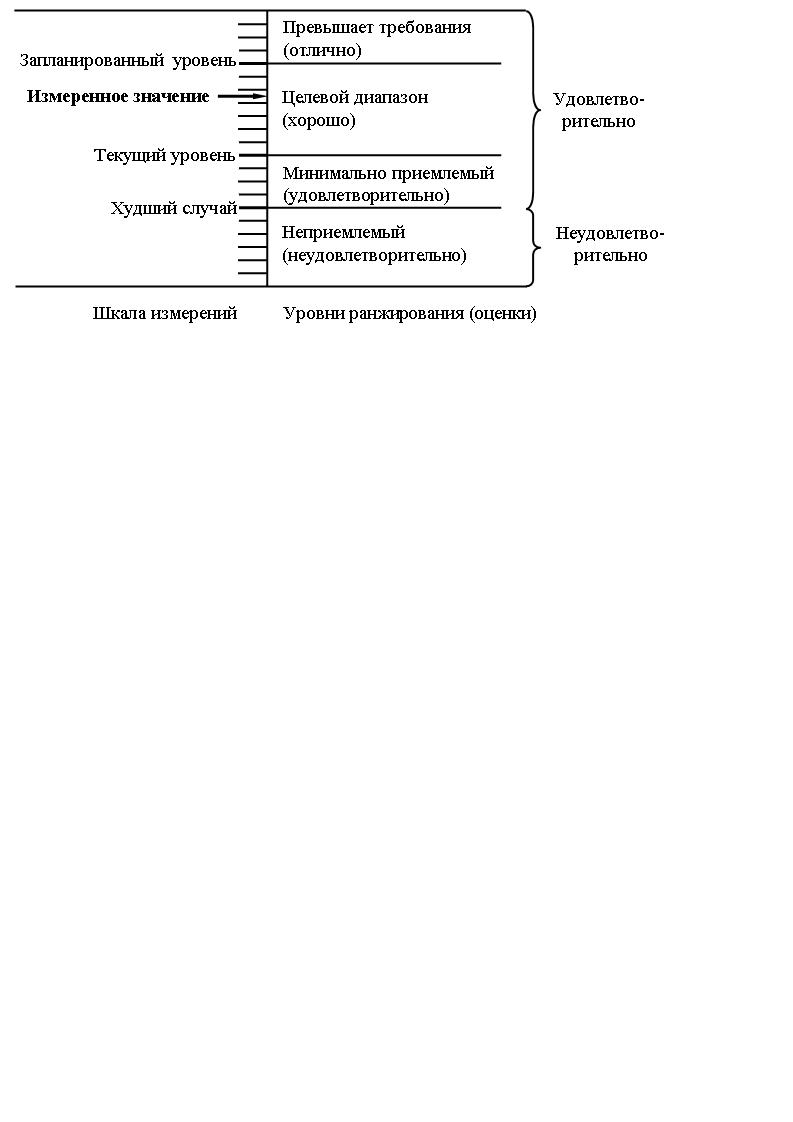
Цель: подготовка основы для оценивания.

Этапы:

1. Выбор метрик качества.
2. С учетом регламентированной в *СТБ ИСО/МЭК 9126–2003* иерархической модели качества уровень характеристик качества ПС определяется уровнем входящих в них подхарактеристик, а значения подхарактеристик в свою очередь определяются значениями входящих в них метрик.

2. Определение уровней ранжирования.

Для измерения количественных признаков ПС используются метрики качества. екущий уровень определяется для управления тем, чтобы новая система не становилась хуже по сравнению с существующей. Запланированный уровень определяет уровень, который считается достижимым при доступных ресурсах. Уровень худшего случая определяет границу принятия пользователем в случае, если изделие не удовлетворяет запланированному уровню Так как качество ПС связано с конкретными потребностями, общие уровни ранжирования невозможны и должны определяться для каждого конкретного оценивания.



3. Определение критерия оценки.

Для определения общего качества ПС должна быть учтена вся совокупность результатов оценивания различных метрик. Оценщик должен подготовить для этого процедуры, таблицы решений или средние взвешенные значения.

**Процедура оценивания**

Этапы:

1. Измерение

Для измерения выбранные метрики применяются к ПС. Результатом являются значения в масштабах метрик.

2. Ранжирование На этапе ранжирования устанавливается уровень ранжирования для измеренного значения

3. Оценка является последним этапом процесса оценивания ПС, на котором обобщается множество установленных уровней. Результатом является заключение о качестве ПС (приемлемый или неприемлемый уровень качества).

К **недостаткам** данного метода следует отнести отсутствие рекомендуемых вариантов метрик и представление метода лишь в общем виде (в виде модели), что затрудняет его контекстное использование.

**15. ГОСТ 28195-99. Фазы процессов разработки и применения. Иерархическая модель сопровождаемости для фазы анализа программных средств.**

Базируется на следующих процессах и фазах ЖЦ ПС:

Процесс разработки:

1. Фаза анализа

2. фаза проектирования

3. фаза реализации

4. фаза тестирования

5. фаза изготовления

Процесс применения:

1. фаза внедрения

2. фаза эксплуатации

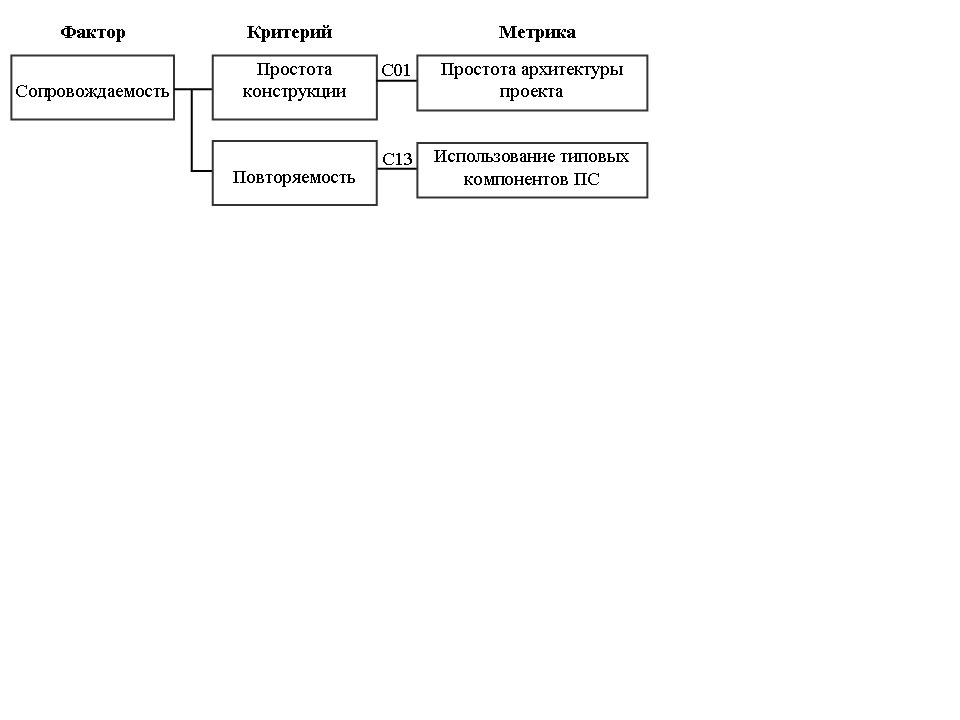
3. фаза сопровождения

Фазы представляют собой временные периоды, соответствующие работам, совокупностям работ или процессам жизненного цикла программных средств.

*четырехуровневая иерархическая модель качества.*



модель сопроваждаемости для фазы анализа

**Метрики критерия:**

использование типовых компонентов программного средства

Выбор оценочных элементов метрики зависит от функционального назначения и формируется с учетом полученных ранее данных при проведении испытаний программного средства.

Для выбора оценочных элементов предпологается перечень таблиц содержащих наименование элементов, метод оценки.

Первый символ (буква) указывает на принадлежность элемента фактору). Два следующих символа – номер метрики, которой принадлежит оценочный элемент (для *Сопровождаемости* номера метрик обозначены на Четвертый и пятый символы – порядковый номер данного оценочного элемента в метрике.

| Код элемента | Наименование | Метод оценки |
| --- | --- | --- |
| С0101 | Наличие модульной схемы программы | Экспертный |

Чем больше уникальных модулей тем и ниже сопровождаемость

**16. ГОСТ 28195-99. Метод количественной оценки качества программных средств.**

1. На фазе анализа проводится выбор показателей и их базовых значений.

2. Для показателей качества на всех уровнях принимается единая шкала

оценки (от 0 до 1).

3. В процессе оценки качества на каждом уровне (кроме уровня оценочных

элементов) проводится вычисление двух величин:

1. абсолютного показателя качества ***Pij* ;**

2. относительного показателя качества ***Rij* ,**

где ***j*** – порядковый номер показателя данного уровня для ***i***-го показателя вышестоящего уровня. Безымянный2.png

4. Каждый показатель качества второго и третьего уровней характеризуется двумя параметрами:

1. количественным значением;

2. весовым коэффициентом ***Vij*** .

Сумма весовых коэффициентов всех показателей некоторого уровня, относящихся к показателю вышестоящего уровня, постоянна и равна 1.

5. Определение усредненной оценки ***mkq*** оценочного элемента по нескольким его значениям (измерениям) ***mqt*** осуществляется по следующей формуле

Безымянный1.png

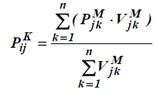
6. Итоговая оценка ***k***-ой метрики ***j***-го критерия определяется по формуле

Безымянный3.png

где ***М*** – признак метрики; ***Q*** – число оценочных элементов, реально используемых при оценке ***k***-й метрики.

7. Абсолютные показатели ***j***-го критерия ***i***-го фактора вычисляются по фор-

муле



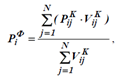
где ***n*** – число метрик, относящихся к ***j***-му критерию, реально используемых при оценке; ***К*** – признак критерия

8. Относительные значения *RijK* j-го критерия i-го фактора *PijK*по отношению к базовому значению *PijKа*определяются по формуле:  *RijK = PijK/ PijKа*

.

9. Абсолютные и относительные значения ***i***-го

фактора качества определяются по формулам



где ***F*** – признак фактора; ***N*** – число критериев качества, относящихся к ***i***-му фактору, реально используемых при оценке.

10. Общая оценка качества в целом формируется экспертами по набору

полученных значений факторов качества.

**17. ISO/IEC 9126-1:2001 Модель внешнего и внутреннего качества программных средств (характеристики и подхарактеристики).**

Модель внутренних и внешних характеристик качества ПС состоит из **шести групп** базовых показателей, каждая из которых детализирована несколькими нормативными **подхарактеристиками**:

**Функциональность** **(Functionality)** – способность программного продукта обеспечивать функции, удовлетворяющие установленные и подразумеваемые потребности при применении программного средства в заданных условиях..

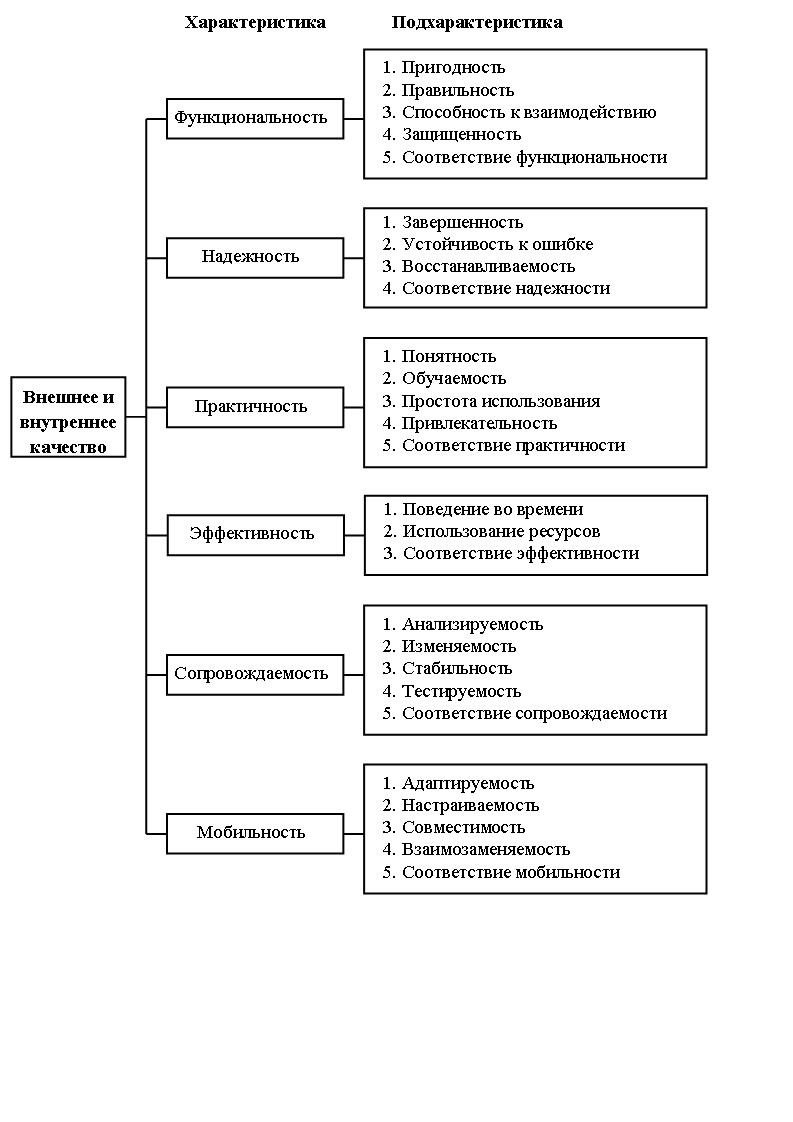
**Надежность (Reliability)** – способность программного продукта поддерживать заданный уровень качества функционирования при его использовании в заданных условиях. Ограничения надежности в процессе эксплуатации вызваны ошибками в требованиях, проектировании и кодировании.

**Практичность (Usability)** – способность программного продукта быть понятным, изученным, использованным и привлекательным для пользователя при применении в заданных условиях. Очевидно, что практичность зависит от некоторых аспектов таких характеристик качества, как функциональность, надежность и эффективность.

**Эффективность (Efficiency)** – способность программного продукта обеспечить соответствующую производительность в зависимости от количества используемых вычислительных ресурсов в заданных условиях. Ресурсы могут включать другие программные продукты, конфигурацию программных и аппаратных средств системы и материалы.

**Сопровождаемость (Maintainability)** – способность программного продукта к модификации. Модификации могут включать исправления, усовершенствования или адаптацию ПС к изменениям в среде применения, в требованиях и функциональных спецификациях.

**Мобильность (Portability)** – способность программного продукта к переносу из одной среды в другую. Среда может включать организационное, аппаратное и программное окружение.

Дополнительно каждая характеристика сопровождается **подхарактеристикой** согласованность, которая должна отражать отсутствие противоречий с иными стандартами и нормативными документами, а также с другими показателями в данном стандарте. В стандарте ISO 9126 отсутствуют методики количественного измерения характеристик и сопоставления с требованиями спецификаций, а также рекомендации, на каких этапах ЖЦ ПС их целесообразно применять.

**18. ISO/IEC 9126-1:2001. Модель качества в использовании.**

**Качество в использовании** (Quality in use) – это способность программного продукта позволять заданным пользователям достигать заданные цели с результативностью, продуктивностью, безопасностью и удовлетворением в заданном контексте использования. Качество в использовании – это восприятие пользователем качества окружающей среды, содержащей ПП. Оно измеряется больше на основе результатов использования ПП в среде, чем на основе собственных внутренних свойств ПП.

Модель качества в использовании является **иерархической**. На ее верхнем уровне находятся четыре характеристики. Характеристики определяются непосредственно метриками (второй уровень модели). В отличие от модели внешнего и внутреннего качества уровень подхарактеристик в модели качества в использовании отсутствует.

Качество в использовании разделяется на четыре **характеристики**:

1. **Результативность** (Effectiveness) – это способность программного продукта, позволяющая пользователям достигать заданные цели с точностью и полнотой в заданном контексте использования.

2. **Продуктивность** (Productivity) – это способность программного продукта, позволяющая пользователям расходовать количество ресурсов, соответствующее результативности, достигаемой в заданном контексте использования. Ресурсы могут включать время выполнения задачи, усилия пользователя, материалы, стоимость использования.

3. **Безопасность** (Safety) – это способность программного продукта достигать приемлемых уровней риска причинения вреда людям, бизнесу, программному обеспечению, имуществу или окружающей среде в заданном контексте использования. Обычно риски – это результат дефектов в функциональности (включая защищенность), надежности, практичности и сопровождаемости.

4. **Удовлетворенность** (Satisfaction) – это способность программного продукта удовлетворять пользователя в заданном контексте использования. Удовлетворенность определяется реакцией пользователя на взаимодействие с программным продуктом и включает отношение к применению продукта.

**19.** **ISO/IEC TR 9126-3:2003. Внутренние метрики программных средств.**

**Внутренние метрики** – это метрики, измеряющие собственные свойства программного средства. Они измеряются в процессе разработки программного средства на основе спецификации требований, результатов проектирования, исходного кода или другой документации программного средства. Внутренние метрики дают возможность оценить качество промежуточных программных продуктов разработки, предсказывая качество конечного программного средства.

**Внутренние метрики функциональности** предназначены для предсказания того, удовлетворяет ли разрабатываемый программный продукт требованиям к функциональности и предполагаемым потребностям пользователя. Примеры метрик:

1. Полнота функциональной реализации, оценивает пригодность ПС.
2. Точность, оценивает правильность ПС.(ФОРМУЛА 1)Х=А/В

|  |  |  |
| --- | --- | --- |
| Правильность | Точность  (1) | ***А*** – количество элементов данных, реализованных с заданными уровнями точности, подтвержденное при оценке; ***В*** – количество элементов данных, для которых в спецификации заданы уровни точности |

3) Соответствие интерфейсов, оценивает способность к взаимодействию.

4) Предотвращение разрушения данных, оценивает защищенность.

5) Соответствие функциональности, оценивает соответствие функциональности.

**Внутренние метрики надежности** используются во время разработки программного продукта для предсказания того, удовлетворяет ли программный продукт заявленным потребностям в надежности. Примеры метрик:

1. Полнота тестирования, оценивает завершенность.(1)

|  |  |  |
| --- | --- | --- |
| Завершенность | Полнота  тестирования (1) | ***А*** – количество тестовых комбинаций, спроектированных в плане тестирования и подтвержденных при проверке; ***В*** – количество требуемых тестовых комбинаций |

2) Предотвращение некорректных действий, оценивает устойчивость к ошибке.

3) Способность к восстановлению, оценивает восстанавливаемость.

4) Соответствие надежности, оценивает соответствие надежности.

**Внутренние метрики практичности** используются во время разработки программного продукта для предсказания степени, в которой программный продукт может быть понят, изучен, управляем, привлекателен и соответствует договоренностям и руководствам по практичности

1) Способность к демонстрации, оценивает понятность.

2) Полнота документации пользователя, оценивает обучаемость.

|  |  |  |
| --- | --- | --- |
| Обучаемость | Полнота документации пользователя и/или возможности электронной справки help (1) | ***А*** – количество описанных функций; ***В*** – общее количество предоставляемых функций |

3) Отменяемость действий пользователя, оценивает простоту использования.

4) Настраиваемость вида интерфейса пользователя, оценивает привлекательность.

5) Соответствие практичности, оценивает соответствие практичности.

**Внутренние метрики эффективности** используются во время разработки программного продукта для предсказания эффективности поведения программного продукта во время тестирования или эксплуатации. Примеры метрик:

1. Пропускная способность, оценивает поведение во времени.

|  |  |  |
| --- | --- | --- |
| Поведение во времени | Пропускная способность ***Х = А*** | ***А*** – число задач в единицу времени, подтвержденное при проверке |

2) Использование памяти, оценивает использование ресурсов.

3) Соответствие эффективности, оценивает соответствие эффективности.

**Внутренние метрики сопровождаемости** используются для предсказания уровня усилий, необходимых для модификации программного продукта. Примеры метрик:

1) Готовность диагностических функций, оценивает анализируемость.

2) Регистрируемость изменений, оценивает изменяемость.

3) Влияние изменений, оценивает стабильность.

4) Полнота встроенных функций тестирования, оценивает тестируемость.

|  |  |  |
| --- | --- | --- |
| Тестируемость | Полнота встроенных функций  тестирования (1) | ***А*** – количество реализованных встроенных функций тестирования из заданных в спецификации, подтвержденное при проверке; ***В*** – требуемое количество встроенных функций тестирования |

5) Соответствие сопровождаемости, оценивает соответствие сопровождаемости.

**Внутренние метрики мобильности** используются для предсказания воздействия программного продукта на поведение исполнителя или системы при проведении работ по переносу. Примеры метрик:

1) Адаптируемость структур данных, оценивает адаптируемость.

2) Объем работ по установке, оценивает настраиваемость.

|  |  |  |
| --- | --- | --- |
| Настраиваемость | Объем работ по установке  (1) | ***А*** – количество автоматических шагов установки (инсталляции), подтвержденное при проверке; ***В*** – требуемое количество шагов инсталляции |

3) Доступная совместимость, оценивает совместимость.

4) Преемственность данных, оценивает взаимозаменяемость.

5) Соответствие мобильности, оценивает соответствие мобильности.

**20.** **ISO/IEC TR 9126-2:2003. Внешние метрики программных средств.**

**Внешние метрики** – это метрики, предназначенные для измерения качества программного продукта путем измерения поведения системы, частью которой является данный продукт. Внешние метрики могут быть использованы в процессе эксплуатации и на стадиях тестирования или испытаний в процессах разработки и сопровождения программного средства, когда уже созданы исполнимые коды программного продукта.

**Внешние метрики функциональности** должны измерять свойства (атрибуты) функционального поведения системы, содержащей программное средство. Примеры:

1. Полнота функциональной реализации, оценивает пригодность ПС.

|  |  |  |
| --- | --- | --- |
| Пригодность | Полнота функциональной  реализации (1) | ***А*** – количество отсутствующих функций, обнаруженных при проверке; ***В*** – количество функций, описанных в спецификации требований |

2) Точность, оценивает правильность ПС.

3) Способность к обмену данными (основанная на успешных попытках пользователя), оценивает способность к взаимодействию.

4) Предотвращение разрушения данных, оценивает защищенность.

5) Соответствие функциональности, оценивает соответствие функциональности.

**Внешние метрики надежности** должны измерять свойства, связанные с поведением системы, содержащей программное средство, во время тестирования, чтобы показать степень надежности программного средства в системе в процессе эксплуатации.

1. Плотность ошибок, оценивает завершенность.

|  |  |  |
| --- | --- | --- |
| Завершенность | Плотность ошибок  ***Х = А / Z*** | ***А*** – количество ошибок, обнаруженных в течение определенного испытательного срока; ***Z*** – размер продукта |

2) Предотвращение некорректных действий, оценивает устойчивость к ошибке.

3) Способность к восстановлению, оценивает восстанавливаемость.

4) Соответствие надежности, оценивает соответствие надежности.

**Внешние метрики практичности** показывают, в какой мере программное средство может быть понято, изучено, управляемо, привлекательно и соответствует договоренностям и руководствам по практичности.

1. Полнота описания, оценивает понятность.

|  |  |  |
| --- | --- | --- |
| Понятность | Полнота описания (1) | ***А*** – количество функций (или классов функций), понятных после прочтения документации на программный продукт; ***В*** – общее количество функций (или классов функций), реализуемых программным продуктом |

2) Эффективность документации пользователя, оценивает обучаемость.

3) Исправление ошибок при использовании, оценивает простоту использования.

4) Изменяемость вида интерфейса, оценивает привлекательность.

5) Соответствие практичности, оценивает соответствие практичности.

**Внешние местрики Эффективность** - совокупность свойств программного средства, характеризующая те аспекты его уровня пригодности, которые связаны с характером и временем использования ресурсов, необходимых при заданных условиях функционирования.

1) Использование устройств ввода/вывода.

2) Соответствие эффективности.

3) Время отклика.

**Сопровождаемость** - совокупность свойств программного средства, характеризующая усилия, которые необходимы для его модификации.

1. Поддержка диагностическими функциями.

|  |  |  |
| --- | --- | --- |
| Анализируемость | Поддержка  диагностическими функциями (1) | ***А*** – количество отказов, при которых персонал сопровождения с помощью диагностических функций может диагностировать причину; ***В*** – общее число зарегистрированных отказов |

2) Возможность управления изменением ПС.

3) Название метрики, формула для её оценки.

4) Возникновение отказа после изменения.

5) Доступность встроенных функций тестирования.

6) Соответствие сопровождаемости.

**Мобильность** - совокупность свойств программного средства, характеризующая приспособленность для переноса из одной среды функционирования в другие.

1. Адаптируемость структурных данных.

|  |  |  |
| --- | --- | --- |
| Адаптируемость | Адаптируемость  структур данных  (1) | ***А*** – количество работоспособных данных, которые не требуют сопровождения при адаптации; ***В*** – ожидаемое число работоспособных данных в окружающей среде, к которой ПС адаптировано |

2) Простота установки.

3) Доступная совместимость.

4) Переемственность данных.

5) Соответствие мобильности.

6) Название метрики, формула для её оценки.

**21. Группы стандартов серии SQuaRE.**

Активно ведется разработка серии стандартов Системная и программная инженерия - Требования к качеству и оценка программного продукта.

**Мера (**measure) – переменная, которой присваивается значение в резуль- тате измерений. Термин «мера» используется собирательно для указания базо- вых мер, производных мер и указателей.

**Серия SQuaRE разделена на:**

1) **Группа управления качеством** (ISO/IEC 2500n)

Стандарты данной группы определяют общие модели , термины и определения, которые используются в остальных стандартах SQuaRE; содержит руководства по использованию стандартов серии SQuaRE.

2) **Группа модели качества** (ISO/IEC 2501n)

В стандартах этой группы представлены модели качества ПС, кач-ва в использовании, кач-ва данных. Содержит практическое руководство использования данных моделей.

3) **Группа измерения качества** (ISO/IEC 2502n)

Содержит эталонную модель измерения качества ПС, математические определения мер качества и практическое руководство по их применению.

4) **Группа требований к** качеству (ISO/IEC 2503n)

Стандарты данной группы помогают определить требования к качеству, основываясь на моделях и мерах качества. Эти требования могут использоваться для выявления требований к качеству разрабатываемого ПС и входных данных для оценки.

5) **Группа оценки качества** (ISO/IEC 2504n)

Стандарты этой группы содержат руководства по оценке ПП оценщиками, заказчиками и руководством.

6) **Группа расширения** (ISO/IEC 25050-25099)

Содержит требования к качеству коммерческих готовых ПП и общие промышленные форматы для отчетов по практичности.

**22. Модель качества продукта по стандарту ISO/IEC 25010:2011 (характеристики и подхарактеристики).**

Данная модель подразделяет свойства качества системы или программного продукта на восемь **характеристик**.

1. **Функциональное соответствие** - степень обеспечения программным продуктом или системой функций, отвечающих заданным или подразумеваемым потребностям при использовании в заданных условиях. **Подхарактеристики**: функциональная полнота, функциональная правильность, функциональная пригодность.

2. **Эффективность функционирования** - зависимость функционирования от кол-ва ресурсов, используемых в заданных условиях. Ресурсы могут включать другие программные продукты, программную и аппаратную конфигурацию системы, материалы. **Подхарактеристики**: поведение во времени, использование ресурсов, емкость.

3. **Совместимость** - степень возможностей программного продукта, системы или компонента обмениваться информацией с другими продуктами, системами или компонентами и/или выполнять свои заданные функции при совместном использовании одной и той же аппаратной или программной среды. **Подхарактеристики**: сосуществование, способность к взаимодействию.

4. **Практичность** - степень применимости программного продукта или системы заданными пользователями для достижения заданных целей с результативностью, эффективностью и удовлетворенностью в заданном контексте использования. **Подхарактеристики**: распознаваемость пригодности, обучаемость, простота использования, защита от ошибок пользователей, эстетичность пользовательского интерфейса, доступность.

5. **Надежность** - степень выполнения системой, программным продуктом или компонентом заданных функций в заданных условиях в течении заданного периода времени. Подхарактеристики: завершенность, готовность, устойчивость к ошибке, восстанавливаемость.

6. **Защищенность** - степень защиты программным продуктом или системой информации и данных так, чтобы люди, другие продукты или системы имели степень доступа к данным, соответствующую типам и уровням их авторизации. **Подхарактеристики**: конфиденциальность, целостность, неопровержимость, идентифицируемость, аутентичность.

7. **Сопровождаемость** - степень результативности и продуктивности модификации программного продукта или системы запланированным персоналом сопровождения. Модификации включают: исправления, улучшения, адаптация к новым требованиям. **Подхарактеристики**: модульность, повторная используемость, анализируемость, модифицируемость, тестируемость.

8. **Мобильность** - степень результативности и эффективности переноса системы, программного продукта или компонента из одной аппаратной, программной или иной эксплуатационной или используемой среды в другую. **Подхарактеристики**: адаптируемость, простота установки, взаимозаменяемость.

**23. Модель качества в использовании по стандарту ISO/IEC 25010:2011.**

**Качество в использовании** - степень применимости продукта или системы заданными пользователями для удовлетворения их потребностей в достижении заданных целей с результативностью, эффективностью, свободой от риска и удовлетворенностью в заданных контекстах использования.

Качество в использовании характеризует влияние, которое продукт оказывает на правообладателей. Оно определяется качеством программного обеспечения, аппаратных средств и эксплуатационной среды, а также характеристиками пользователей, задач и социального окружения

Данная модель состоит из 5 **характеристик**:

1. **Результативность** - точность и полнота, с которой пользователи достигают заданных целей.

2. **Эффективность** - ресурсы, затрачиваемые в зависимости от точности и полноты, с которыми пользователь достигает целей.

3. **Удовлетворенность** - степень удовлетворения потребностей пользователя при применении продукта или системы в заданном контексте использования. **Подхарактеристики**: применимость, доверие, удовольствие, комфорт.

4. **Свобода от риска** - степень уменьшения продуктом или системой потенциального риска для экономического статуса, человеческой жизни, здоровья или окружающей среды. **Подхарактеристики**: уменьшение экономического риска, уменьшение риска для здоровья и безопасности, уменьшение риска для окружающей среды.

5. **Покрытие контекста** - степень применимости продукта или системы с результативностью, эффективностью, свободой от риска и удовлетворенностью как в заданных контекстах использования, так и вне них. **Подхарактеристики**: покрытия контекста являются полнота контекста, гибкость.

**24 Основные понятия в области технического нормирования, стандартизации и оценки соответствия.**

В Законе «О техническом нормировании и стандартизации» определены следующие **технические нормативные правовые акты** в области технического нормирования и стандартизации:

1. технические регламенты;

2. технические кодексы установившейся практики;

3. государственные стандарты Республики Беларусь;

4. технические условия.

**Технический регламент (ТР)** – это технический нормативный правовой акт, разработанный в процессе технического нормирования, устанавливающий134 непосредственно и/или путем ссылки на технические кодексы установившейся практики и/или государственные стандарты Республики Беларусь обязательные для соблюдения технические требования, связанные с безопасностью продукции и процессов ее жизненного цикла.

**Технический кодекс установившейся практики (ТКП)** – это технический нормативный правовой акт, разработанный в процессе стандартизации, содержащий основанные на результатах установившейся практики технические требования к процессам жизненного цикла продукции или оказанию услуг.

**Стандарт** – это технический нормативный правовой акт, разработанный в процессе стандартизации на основе согласия большинства заинтересованных субъектов технического нормирования и стандартизации и содержащий технические требования к продукции, процессам ее жизненного цикла или оказанию услуг.

**Технические условия (ТУ)** – это технический нормативный правовой акт, разработанный в процессе стандартизации, утвержденный юридическим лицом или индивидуальным предпринимателем и содержащий технические требования к конкретным типу, марке, модели, виду реализуемой ими продукции или оказываемой услуге, включая правила приемки и методы контроля.

В Законе определено, что обязательные требования устанавливаются в технических регламентах, принимаемых органами власти, а стандарты применяются в добровольном порядке.

В области оценки соответствия приняты следующие основные термины и их определения:

**Аккредитация** - вид оценки соответствия, результатом осуществления которого является официальное признание компетентности юридического лица в выполнении работ по подтверждению соответствия и (или) проведении испытаний продукции.

**Аккредитованная испытательная лаборатория (центр)** - юридическое лицо, аккредитованное для проведения испытаний продукции в определенной области аккредитации.

**Аккредитованный орган по сертификации** - юридическое лицо, аккредитованное для выполнения работ по подтверждению соответствия в определенной области аккредитации.

**Аттестат аккредитации** - документ, удостоверяющий компетентность юридического лица в выполнении работ по подтверждению соответствия или в проведении испытаний продукции в определенной области аккредитации.

**Декларация о соответствии** - документ, в котором изготовитель (продавец) удостоверяет соответствие производимой и (или) реализуемой им продукции требованиям технических нормативных правовых актов в области технического нормирования и стандартизации.

**Декларирование соответствия** - форма подтверждения соответствия, осуществляемого изготовителем (продавцом).

**Национальная система подтверждения соответствия Республики Беларусь** - установленная совокупность субъектов оценки соответствия, нормативных правовых актов и технических нормативных правовых актов в области технического нормирования и стандартизации, определяющих правила и процедуры подтверждения соответствия и функционирования системы в целом.

**Область аккредитации** - сфера деятельности, в которой аккредитованному органу по сертификации или аккредитованной испытательной лаборатории (центру) предоставлено право на выполнение работ по подтверждению соответствия или проведение испытаний продукции.

**Оценка соответствия** - деятельность по определению соответствия объектов оценки соответствия требованиям технических нормативных правовых актов в области технического нормирования и стандартизации.

**Подтверждение соответствия** - вид оценки соответствия, результатом осуществления которого является документальное удостоверение соответствия объекта оценки требованиям технических нормативных правовых актов в области технического нормирования и стандартизации.

**Сертификат компетентности** - документ, удостоверяющий профессиональную компетентность физического лица в выполнении определенных работ, услуг.

**Сертификат соответствия –** документ, удостоверяющий соответствие объекта оценки требованиям технических нормативных правовых актов в области технического нормирования и стандартизации.

**Сертификация** - форма подтверждения соответствия, осуществляемого аккредитованным органом по сертификации.

**Система управления качеством (система менеджмента качества)** - часть общей системы управления, включающая организационную структуру, планирование, ответственность, методы, процедуры, процессы, ресурсы, необходимые для обеспечения качества продукции и (или) услуг; как правило, система управления качеством является частью системы управления предприятием или организацией.

**Схема подтверждения соответствия** - установленная последовательность действий, результаты которых рассматриваются в качестве доказательств соответствия объекта оценки требованиям технических нормативных правовых актов в области технического нормирования и стандартизации.

**Форма подтверждения соответствия** - установленный порядок документального удостоверения соответствия объекта оценки требованиям технических нормативных правовых актов в области технического нормирования и стандартизации.

**25. Оценка соответствия в Республике Беларусь.**

Основными **объектами** оценки соответствия являются:

1. продукция;

2. процессы жизненного цикла продукции (разработка, производство, эксплуатация, хранение, перевозка, реализация и утилизация);

3. оказание услуг;

4. система управления качеством;

5.система управления окружающей средой;

6. юридические лица;

7. персонал.

**Субъектами** оценки соответствия являются:

1. государство в лице уполномоченных государственных органов;

2. аккредитованные органы по сертификации;

3. аккредитованные испытательные лаборатории (центры);

4. заявители на аккредитацию;

5. заявители на подтверждение соответствия.

**Виды оценки:** аккредитация и подтверждение соответствия

**Подтверждение соответствия** может носить обязательный или добровольный характер.

К **документам** об оценке соответствия относятся:

1. аттестат аккредитации;

2. сертификат соответствия;

3. декларация о соответствии;

4. сертификат компетентности.

26**. Основы сертификации программных средств.**

Существует два **вида** сертификации программных средств – обязательная сертификация и добровольная сертификация. Результатом положительного прохождения сертификации является выдача сертификата соответствия.

**Обязательной сертификации** подвергаются программные средства, выполняющие особо ответственные функции, в которых недостаточное качество и ошибки представляют потенциальную опасность для жизни, здоровья и наследственности человека, имущества и окружающей среды.

**Добровольная сертификация** применяется для удостоверения качества программного средства с целью повышения их конкурентоспособности и создания благоприятных условий для обеспечения свободного перемещения продукции на внутреннем и внешнем рынках.

**Процесс сертификации программных средств включает:**

1) анализ и выбор разработчиком или заказчиком компетентного в данной области органа по сертификации;

2) подачу заявителем заявки на сертификацию в орган сертификации;

3) принятие органом сертификации решения по заявке, выбор испытательной лаборатории и схемы сертификации;

4) обзор и идентификацию версий ПС, подлежащих испытаниям;

5) сертификационные испытания ПС испытательной лабораторией;

6) анализ полученных результатов и принятие решения лабораторией или органом сертификации о возможности выдачи заявителю сертификата соответствия;

7) выдачу органом сертификации заявителю сертификата на сертифицированную версию программного средства;

8) осуществление инспекционного контроля органом сертификации за

сертифицированным программным средством;

9) проведение заявителем корректирующих мероприятий при нарушении соответствия программного средства установленным требованиям;

10) регистрацию и публикацию информации о результатах сертификации программного средства.