ФЕДЕРАЛЬНОЕ АГЕНТСТВО

ПО ТЕХНИЧЕСКОМУ РЕГУЛИРОВАНИЮ И МЕТРОЛОГИИ

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

ГОСТ Р 58976— 2020/ ISO/TR 80002-2:2017

Изделия медицинские

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

Часть 2

Валидация программного обеспечения, используемого в системах качества медицинских изделий

(ISO/TR 80002-2:2017, Medical device software — Part 2: Validation of software for medical device quality systems, IDT)

Издание официальное

Москва
Стандартинформ
2020

gost.e-ecolog.ru

Предисловие

1 ПОДГОТОВЛЕН Обществом с ограниченной ответственностью «МЕДИТЕСТ» (ООО «МЕДИТЕСТ») на основе официального перевода на русский язык англоязычной версии указанного в пункте 4 международного документа, который выполнен рабочей группой в составе представителей ООО «МЕДИТЕСТ» и Общества с ограниченной ответственностью «АУРИГА» (ООО «АУРИГА»)

2 ВНЕСЕН Техническим комитетом по стандартизации ТК 436 «Менеджмент качества и общие аспекты медицинских изделий»

3 УТВЕРЖДЕН И ВВЕДЕН В ДЕЙСТВИЕ Приказом Федерального агентства по техническому регулированию и метрологии от 27 августа 2020 г. № 566-ст

4 Настоящий стандарт идентичен международному документу ISO/TR 80002-2:2017 «Программное обеспечение медицинских изделий. Часть 2. Валидация программного обеспечения, используемого в рамках систем качества медицинских изделий (ISO/TR 80002-2:2017 «Medical device software — Part 2: Validation of software for medical device quality systems», IDT).

Наименование настоящего стандарта изменено относительно наименования указанного международного документа для приведения в соответствие с ГОСТ Р 1.5—2012 (пункт 3.5)

5 ВВЕДЕН ВПЕРВЫЕ

*Правила применения настоящего стандарта установлены в статье 26 Федерального закона от 29 июня 2015 г. № 162-ФЗ «О стандартизации в Российской Федерации». Информация об изменениях к настоящему стандарту публикуется в ежегодном (по состоянию на 1 января текущего года) информационном указателе «Национальные стандарты», а официальный текст изменений и поправок — в ежемесячном информационном указателе «Национальные стандарты». В случае пересмотра (замены) или отмены настоящего стандарта соответствующее уведомление будет опубликовано в ближайшем выпуске ежемесячного информационного указателя «Национальные стандарты». Соответствующая информация, уведомление и тексты размещаются также в информационной системе общего пользования — на официальном сайте Федерального агентства по техническому регулированию и метрологии в сети Интернет (*gost.e-ecolog.ru

© ISO, 2017 — Все права сохраняются
© Стандартинформ, оформление, 2020

Настоящий стандарт не может быть полностью или частично воспроизведен, тиражирован и распространен в качестве официального издания без разрешения Федерального агентства по техническому регулированию и метрологии

Содержание

1 Область применения . . . 1
2 Нормативные ссылки . . . 1
3 Термины и определения . . . 1
4 Общие аспекты и подход к валидации программного обеспечения . . . 1
4.1 Определение термина «валидация программного обеспечения» . . . 1
4.2 Деятельность по обеспечению доверия к функционированию программного обеспечения: инструменты из набора инструментов . . . 2
4.3 Подход на основе критического мышления . . . 2
5 Валидация программного обеспечения и подход на основе критического мышления . . . 2
5.1 Краткий обзор . . . 2
5.2 Установление необходимости валидации программного обеспечения . . . 7
5.2.1 Идентификация процесса и применяемого программного обеспечения . . . 7
5.2.2 Оценка применимости регулирующих требований . . . 7
5.2.3 Процессы и программное обеспечение, не подпадающие под регулирующие требования к медицинским изделиям . . . 8
5.3 Стадия разработки . . . 8
5.3.1 Планирование валидации . . . 8
5.3.2 Определение/Идентификация . . . 8
5.3.3 Имплементация, тестирование и развертывание . . . 12
5.4 Стадия обслуживания . . . 15
5.4.1 Переход в стадию обслуживания . . . 15
5.4.2 Планирование обслуживания . . . 15
5.4.3 Типы обслуживания, проводимые в рамках стадии . . . 16
5.4.4 Изменения процесса: переход к мерам по управлению риском . . . 16
5.4.5 Экстренные изменения . . . 16
5.4.6 Поддержка предусмотренного применения . . . 17
5.5 Стадия вывода из эксплуатации . . . 17
6 Документация . . . 18
7 Обязательные процессы . . . 19
Приложение А (справочное) Набор инструментов . . . 20
Приложение В (справочное) Менеджмент риска и риск-ориентированный подход . . . 26
Приложение С (справочное) Примеры . . . 29
Библиография . . . 74

Введение

Настоящий стандарт был разработан с целью оказания заинтересованным сторонам практической помощи в определении необходимой деятельности по валидации процесса применения программного обеспечения, используемого в системах качества медицинских изделий, с применением риск-ориентированного подхода, использующего философию критического мышления.

Настоящий стандарт распространяется на программное обеспечение, используемое в системе менеджмента качества, на программное обеспечение, используемое при производстве и предоставлении услуг, а также на программное обеспечение, используемое для мониторинга и измерения требований, в соответствии с требованиями ИСО 13485:2016 (4.1.6, 7.5.6 и 7.6).

Настоящий стандарт является результатом усилий и консолидации опыта работников промышленности медицинских изделий, которые занимаются валидацией такого программного обеспечения и в рамках деятельности своих организаций создают подвергаемую аудиту соответствующую документацию. Настоящий стандарт был разработан с учетом определенных имеющихся вопросов и проблем, с которыми приходится сталкиваться при валидации процесса применения программного обеспечения, используемого в рамках систем качества медицинских изделий, среди которых следующие: Что должно быть сделано? Будет ли этого достаточно? Как проводить анализ рисков? После долгих обсуждений был сделан вывод, что для обеспечения уверенности в способности программного обеспечения функционировать в соответствии с его предусмотренным применением (назначением), в каждом конкретном случае должен определяться некий перечень деятельности (с применением подходящих инструментов из набора/совокупности инструментов). Этот перечень может варьироваться в зависимости от множества факторов, например от сложности программного обеспечения, от риска причинения вреда и происхождения (например, качество, стабильность) поставляемого вендор-поставщиком программного обеспечения. Цель настоящего стандарта заключается в оказании помощи заинтересованным сторонам, включая изготовителей, аудиторов и регулирующие органы, в понимании и применении требований по валидации применения программного обеспечения, установленных в ИСО 13485:2016 (4.1.6, 7.5.6 и 7.6).

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

Изделия медицинские

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ

Часть 2

Валидация программного обеспечения, используемого в системах качества медицинских изделий

Medical devices. Software. Part 2. Validation of software for medical device quality systems

Дата введения — 2021—08—01

1 Область применения

Настоящий стандарт применим к любому программному обеспечению, используемому при проектировании, испытаниях (тестировании), приемке узлов и компонентов, производстве, маркировании, упаковывании, распространении изделий, включая обработку претензий, а также для автоматизации любых других аспектов системы качества медицинских изделий, как это установлено в ИСО 13485.

Настоящий стандарт применим:

- к программному обеспечению, используемому в рамках системы менеджмента качества;
- к программному обеспечению, используемому при производстве и предоставлении услуг;
- к программному обеспечению, используемому для мониторинга и измерения требований.

Настоящий стандарт не применим:

- к программному обеспечению, используемому в качестве компонента, части или принадлежности к медицинскому изделию;
- к программному обеспечению, которое само по себе является медицинским изделием.

2 Нормативные ссылки

Настоящий стандарт не содержит нормативных ссылок.

3 Термины и определения

В настоящем стандарте применены термины по ИСО 9000 и ИСО 13485.

ИСО и МЭК ведут терминологические базы для использования в стандартизации по следующим адресам:

- IEC Electropedia: размещена по адресу: gost.e-ecolog.ru ;
- онлайн-платформа ISO: размещена по адресу: gost.e-ecolog.ru .

4 Общие аспекты и подход к валидации программного обеспечения

4.1 Определение термина «валидация программного обеспечения»

Термин «валидация программного обеспечения» может рассматриваться как в широком, так и в узком смысле, от простого испытания (тестирования) до масштабных действий, включающих проведение тестирования. В настоящем стандарте термин «валидация программного обеспечения» используется для идентификации всей деятельности, которая на заданном уровне обеспечивает уверенность в

Издание официальное

том, что используемое программное обеспечение соответствует его предусмотренному применению, а также является надежным и заслуживающим доверия. Любая обозначенная деятельность (выбранные действия/мероприятия) должна обеспечивать соответствие программного обеспечения предъявляемым к нему требованиям и его предусмотренному применению.

4.2 Деятельность по обеспечению доверия к функционированию программного обеспечения: инструменты из набора инструментов

Применение подходящих инструментов из набора инструментов (см. таблицы А.1—А.5) включает деятельность, выполняемую в течение жизненного цикла программного обеспечения в системе менеджмента качества, снижающую риск и обеспечивающую доверие к его функционированию.

4.3 Подход на основе критического мышления

Настоящий стандарт стимулирует применение подхода на основе метода критического мышления в целях идентификации деятельности, которую необходимо выполнить для адекватной валидации конкретного программного обеспечения. Применение метода критического мышления заключается в использовании процесса анализа и оценивания различных аспектов программного обеспечения, а также предполагаемой среды его применения с целью определения наиболее значимого набора деятельности по обеспечению доверия к программному обеспечению для их применения при валидации. Критическое мышление помогает избежать применения подхода, именуемого как «one-size-fits-all» («всех под одну гребенку»), в котором принимается универсальное решение по валидации без тщательного оценивания свидетельств того, что валидируемый процесс действительно приводит к нужному результату. Использование критического мышления предполагает, что решения по валидации могут сильно различаться от одного программного обеспечения к другому, а также позволяет применять различные решения по валидации к одному и тому же программному обеспечению в аналогичной ситуации. Критическое мышление подвергает оценке предлагаемые решения по валидации с целью обеспечения их соответствия требованиям системы менеджмента качества и учитывает все ключевые заинтересованные стороны и их потребности. Критическое мышление также может быть использовано для пересмотра ранее принятого решения по проведению валидации в случае изменения характеристик и предусмотренного применения программного обеспечения или в случае появления новой подходящей информации.

Критическое мышление приводит к решениям по валидации, которые устанавливают соответствие для производителя, обеспечивают безопасность программного обеспечения для использования, приводят к документированным свидетельствам (которые рецензенты сочтут подходящими и адекватными) и приводят к созданию условий, при которых участвующий в проведении валидации персонал, осуществляющий деятельность по валидации, будет осознавать, что прилагаемые им усилия повышают ценность организации и являются наиболее эффективным способом достижения необходимых результатов.

В приложении С представлены примеры исследований, демонстрирующих возможное применение подхода на основе критического мышления при валидации программного обеспечения, используемого в рамках систем качества медицинских изделий, в различных ситуациях, с разными уровнями сложности, происхождением и уровнем риска.

5 Валидация программного обеспечения и подход на основе критического мышления

5.1 Краткий обзор

Для программного обеспечения, используемого в рамках систем менеджмента качества медицинских изделий, должны быть предусмотрены соответствующие средства управления, обеспечивающие правильную работу на протяжении всего жизненного цикла программного обеспечения в системе менеджмента качества. Применение критического мышления и реализация выбранных действий по укреплению доверия приводят к созданию и поддержанию валидированного состояния (статуса валидации) программного обеспечения. На рисунке 1 приведено концептуальное представление типовых видов деятельности и соответствующего управления, которые являются частью жизненного цикла программного обеспечения в системе менеджмента качества с момента принятия решения об автоматизации процесса и до тех пор, пока программное обеспечение не устареет или больше не будет использоваться в

рамках систем менеджмента качества медицинских изделий. На рисунке 1 изображена последовательная модель, в действительности же этот процесс носит цикличный характер, поскольку определяются элементы, идентифицируются риски и применяется подход на основе критического мышления.

При разработке программного обеспечения, используемого в рамках систем менеджмента качества медицинских изделий, одним из основных видов деятельности по укреплению доверия к функционированию программного обеспечения, выбираемой из набора инструментов, является выбор модели жизненного цикла разработки программного обеспечения. Выбранная модель уже должна содержать основанные на критическом мышлении действия, которые позволяют выбирать другие подходящие инструменты для использования на различных стадиях жизненного цикла. С целью обеспечения функционирования программного обеспечения в соответствии с его предусмотренным применением, результаты анализа и оценивания используются для выбора наиболее подходящего набора действий по укреплению доверия. Настоящий стандарт не подразумевает и не предписывает использование какой-либо конкретной модели разработки программного обеспечения. Однако для простоты остальная часть настоящего стандарта разъясняет концепцию критического мышления в контексте Каскадной модели разработки с использованием общих формулировок. Другие модели разработки программного обеспечения (например, итеративные, спиральные) также могут быть использованы при условии, что к ним применяются соответствующие инструменты и используется подход на основе критического мышления.



Рисунок 1 — Управление жизненным циклом программного обеспечения

При рассмотрении применения программного обеспечения в процессе следует идентифицировать факт его использования как части процесса системы качества посредством проведения анализа предусмотренного применения для рассматриваемого программного обеспечения. Если факт такого использования идентифицирован, то такое программное обеспечение должно быть валидировано на соответствие предусмотренному применению. Несмотря на то что настоящий стандарт описывает

валидацию программного обеспечения, используемого в рамках систем качества медицинских изделий, такой же подход является хорошим примером для оценивания программного обеспечения на соответствие любым установленным требованиям. Самая важная часть валидации программного обеспечения — это разработка/закупка подходящего программного средства с целью использования его как инструмента для поддержания процессов, как предусмотрено изготовителем. Это подразумевает точное установление требований с целью оценивания того, подходит ли разработанное/закупаемое программное обеспечение для выполнения требований предусмотренного применения. Одинаково важны как технические требования, подходящие для верификации, так и технологические требования к процессу, подходящие для валидации. При рассмотрении вопроса о применении программного обеспечения в процессах следует учитывать, что программное обеспечение может взаимодействовать или иметь интерфейсы с другим программным обеспечением.

С целью сбора информации и принятия решений на стадии разработки жизненного цикла программного обеспечения в системе менеджмента качества выполняются задачи по менеджменту риска и планированию валидации в следующих четырех областях:

- уровень прилагаемых усилий, а также экспертиза документации и результатов работ и отчетных материалов;
- объем содержания результатов работ и отчетных материалов;
- выбор инструментов (из набора инструментов) и методов применения этих инструментов;
- уровень трудозатрат и усилий при применении этих инструментов.

Основными факторами, влияющими на принятие решений в этих четырех областях, являются риски, связанные с выполнением процесса, и риски, связанные с функционированием программного обеспечения. На принимаемые решения могут влиять и другие факторы (например, сложность программного обеспечения, сложность процесса, тип и происхождение программного обеспечения).

Процесс планирования проведения валидации состоит из двух отдельных составных частей. Первая часть включает в себя установление уровней тщательности в отношении документирования и проведения анализа полученных результатов. Решения в этой части главным образом определяются результатами анализа рисков процесса. Вторая часть включает в себя установление механизма выбора подходящих инструментов из набора инструментов для реализации тестирования и развертывания программного обеспечения. Выбор подходящих инструментов определяется главным образом посредством анализа риска программного обеспечения. Настоящий стандарт разбивает процесс планирования валидации на разные составные части, исходя из различий в деятельности, связанной с видом проводимого анализа риска. Тем не менее часто конкретные действия объединяются в одну деятельность, которая включает в себя различные аспекты анализа риска и вытекающие из него варианты продолжения проведения валидации.

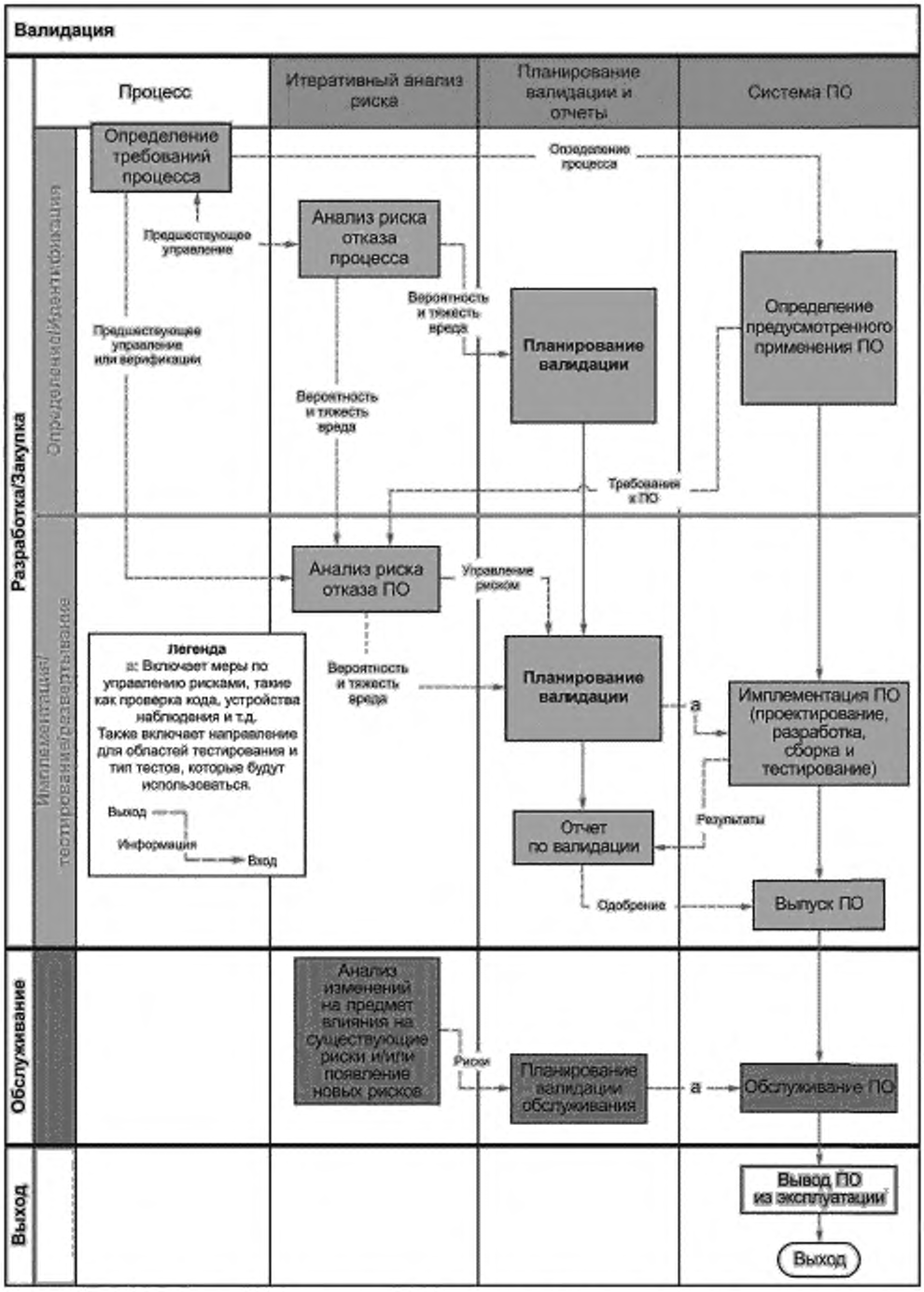
На стадии разработки жизненного цикла программного обеспечения в системе менеджмента качества задачи менеджмента риска и планирования проведения валидации используются для установления соответствующего уровня трудозатрат и необходимости привлечения иных ресурсов, которые будут применены к программному обеспечению, а также для выбора и идентификации подходящих для применения инструментов по укреплению доверия. Использование такого подхода приводит как к выполнению поставленных задач по верификации, так и к реализации дополнительной деятельности, которая является основой для успешного установления валидированного состояния процесса применения программного обеспечения. Как только эта деятельность и задачи будут выполнены, инструменты и связанные с ними результаты вносятся в отчет по валидации программного обеспечения в качестве подтверждения заключения о том, что программное обеспечение является валидированным.

После развертывания программное обеспечение переходит в стадию своего жизненного цикла в системе менеджмента качества — обслуживание. На этой стадии программное обеспечение подвергается мониторингу, а также улучшается и обновляется в соответствии с потребностями бизнеса или изменением регулирующих требований. Деятельность по управлению изменениями на данной стадии основывается на тех же принципах, что и на стадии разработки жизненного цикла программного обеспечения. Однако изменения теперь оцениваются с точки зрения их влияния на предусмотренное применение, риск возникновения отказа, меры по управлению риском, которые применялись в ходе первоначальной разработки, а также на любые функциональные возможности самого программного обеспечения.

Стадия снятия с эксплуатации состоит из прекращения использования программного обеспечения посредством удаления процесса либо путем замены программного обеспечения, используемого для процесса.

На рисунке 1 показана основная деятельность по управлению жизненным циклом программного обеспечения в системе менеджмента качества. Прочие последовательности действий, не отраженные на рисунке 1, включают менеджмент проекта, разработку процессов, управление вендор-поставщиками (если применимо), а также другую деятельность в зависимости от рассматриваемого программного обеспечения.

Рисунок 2 отображает деятельность по управлению жизненным циклом программного обеспечения в системе менеджмента качества с учетом критического мышления и в контексте последовательности действий, не включенных в рисунок 1. Подход на основе критического мышления представлен на рисунке 2 в колонке «Итеративный анализ риска», а также в последовательности действий по валидации. Важно иметь четкие и документированные определения последовательности таких действий в рамках бизнес-модели организации с целью обеспечения правильного менеджмента программного обеспечения как с точки зрения бизнеса, так и с точки зрения регулирования.



Примечание — При использовании термина «разрабатывать» или «разработка» речь идет о разработке валидированного состояния программного обеспечения.

Рисунок 2 — Деятельность по управлению на основных стадиях жизненного цикла программного обеспечения

Стадии жизненного цикла программного обеспечения в системе менеджмента качества и блоки, расположенные в строке на рисунке 2, соответствуют той стадии жизненного цикла программного обеспечения в системе менеджмента качества, которая показана на общей блок-схеме на рисунке 1. Пунктирные линии показывают выходную информацию от одной деятельности, которая обеспечивает входные данные или помогает принимать решения при осуществлении другой деятельности. Диаграмма демонстрирует, что последовательность деятельности определяется наличием входной информации, необходимой для ее реализации. Важно отметить, что вся деятельность выполняется независимо от размера или сложности внедряемого программного обеспечения. Однако для более крупного или более сложного программного обеспечения такая деятельность, скорее всего, будет носить дискретный характер; для меньшего или более простого программного обеспечения некоторые виды деятельности будут объединены или выполняться одновременно.

Таким образом, подход на основе критического мышления устанавливает системный метод для идентификации и включения соответствующей деятельности или инструментов, обеспечивающих доверие к программному обеспечению, в различные последовательности действий для обоснования сделанного заключения о валидации программного обеспечения при выпуске, а также того, что статус валидации программного обеспечения будет поддерживаться до вывода его из эксплуатации.

В следующих подпунктах настоящего стандарта представлены дополнительные детали по управлению на каждой из стадий жизненного цикла программного обеспечения в системе менеджмента качества, изображенных на рисунке 1. В этих подпунктах используется изображение последовательности действий итеративного анализа рисков; валидации программного обеспечения; действий с программным обеспечением, показанных на рисунке 2. Далее приведены описание областей, в которых необходимо принятие решений; факторы, влияющие на принятие решений, которые включают в себя подход на основе критического мышления.

5.2 Установление необходимости валидации программного обеспечения

5.2.1 Идентификация процесса и применяемого программного обеспечения

Первым шагом в определении того, считается ли программное обеспечение используемым в рамках систем качества медицинских изделий, являются идентификация и документирование процессов системы качества высокого уровня с использованием в нем программного обеспечения. Эта задача может показаться малозначительной, если уже известно, что программное обеспечение применяется в системе качества медицинских изделий, и уже осуществляется определение полного предусмотренного применения программного обеспечения. Для менее очевидных ситуаций документирование процесса, использующего программное обеспечение, позволяет четко установить факт нахождения рассматриваемого программного обеспечения в области применения настоящего стандарта (т. е. необходима ли валидация этого программного обеспечения). Для идентифицированного программного обеспечения, выходящего за пределы области применения настоящего стандарта, такая деятельность может помочь обосновать данное решение.

5.2.2 Оценка применимости регулирующих требований

Оценка применимости регулирующих требований может быть использована для отнесения программного обеспечения к «программному обеспечению, используемому в рамках систем качества медицинских изделий» и, следовательно, подпадает ли оно в область применения настоящего стандарта. Начать следует с идентификации конкретных регулирующих требований, которые применяются к процессам, использующим программное обеспечение, а также к записям, которые управляются программным обеспечением. Ряд вопросов может быть использован для полного понимания роли, которую программное обеспечение играет в реализации регулирующих требований. Должны быть рассмотрены следующие вопросы.

a) Могут ли отказ или скрытые недостатки программного обеспечения повлиять на безопасность или качество медицинских изделий?

b) Рассматриваемое программное обеспечение автоматизирует или выполняет деятельность, которая установлена регулирующим требованиями (в частности, требованиями к системам менеджмента качества медицинского изделия)? Примеры могут включать сбор цифровых подписей и/или записей, поддержание прослеживаемости продукции, выполнение и сбор результатов испытаний, ведение журналов данных (например, по корректирующим и предупреждающим действиям, по несоответствиям, претензиям, калибровке и т. д.).

Ответ «да» на любой из вопросов идентифицирует программное обеспечение, которое требует проведения валидации и входит в область применения настоящего стандарта.

Иногда бывает сложно определить, является ли процесс и соответствующее программное обеспечение частью системы качества. Некоторые инструменты могут не оказывать непосредственного влияния на реальное медицинское изделие. Поэтому каждая организация должна тщательно учитывать обстоятельства, связанные с применением такого пограничного программного обеспечения, и должна полностью понимать влияние отказа программного обеспечения на процессы и в конечном итоге на безопасность и результативность любых производимых медицинских изделий. При возникновении каких-либо сомнений лучшее решение состоит в том, чтобы рассматривать применение такого программного обеспечения как подлежащее валидации и использовать подход, установленный в настоящем стандарте.

5.2.3 Процессы и программное обеспечение, не подпадающие под регулирующие требования к медицинским изделиям

Если процессы или программное обеспечение содержат функциональные возможности, которые выходят за рамки действия регулирующих требований в отношении медицинских изделий, то следует проанализировать, какие компоненты программного обеспечения находятся в области применения настоящего стандарта, а какие — нет. Такие решения должны быть обоснованы исходя из степени интеграции между различными компонентами, модулями и структурами данных программного обеспечения и в соответствии с применимыми к организации требованиями соответствия. Приведение такого обоснования особенно важно в случае программного обеспечения, используемого для поддержки системы качества [например, программного обеспечения для планирования корпоративных ресурсов (ERP)]. Программное обеспечение ERP может включать функциональные возможности для процессов, не связанных с медицинским изделием, таких как бухгалтерский учет и финансы. Функциональность может иметь решающее значение для деловых операций и должна соответствовать определенным государственным требованиям (например, требованиям Закона Сарбейнса — Оксли).

5.3 Стадия разработки

5.3.1 Планирование валидации

Первая часть деятельности по планированию валидации с применением критического мышления использует результаты анализа риска процесса (см. приложение В) с целью установления масштаба усилий, которые следует применять к созданию документации, а также для выбора инструментов из раздела «Набор инструментов» блока «Определение/Идентификация» (см. таблицы А.1—А.5). Вторая часть использует результаты анализа риска программного обеспечения с целью выбора инструментов (из набора инструментов) для имплементации, тестирования и развертывания. После выполнения деятельности и установления валидированного состояния программного обеспечения свидетельства валидации документируются в отчете по валидации.

На данной стадии могут применяться различные модели разработки жизненного цикла программного обеспечения в системе менеджмента качества. Настоящий стандарт не рекомендует какую-либо модель, однако предполагает использование методологии управления разработкой. Такая методология будет основана на концепции определения требований (включая предусмотренное применение) перед имплементацией, тестированием и развертыванием, которые имеют основополагающее значение при установлении валидированного состояния программного обеспечения для его предусмотренного применения.

5.3.2 Определение/Идентификация

5.3.2.1 Требование блока «Определение/Идентификация»

Деятельность, выполняемая в рамках блока «Определение/Идентификация» (см. рисунок 1), включает определение/идентификацию процесса, определение предусмотренного применения программного обеспечения в этом процессе, а также планирование масштаба усилий по валидации на основе рисков, идентифицированных в рассматриваемом процессе. Рисунок 3 иллюстрирует эту часть стадии разработки в рамках выбранного примера Каскадной модели.

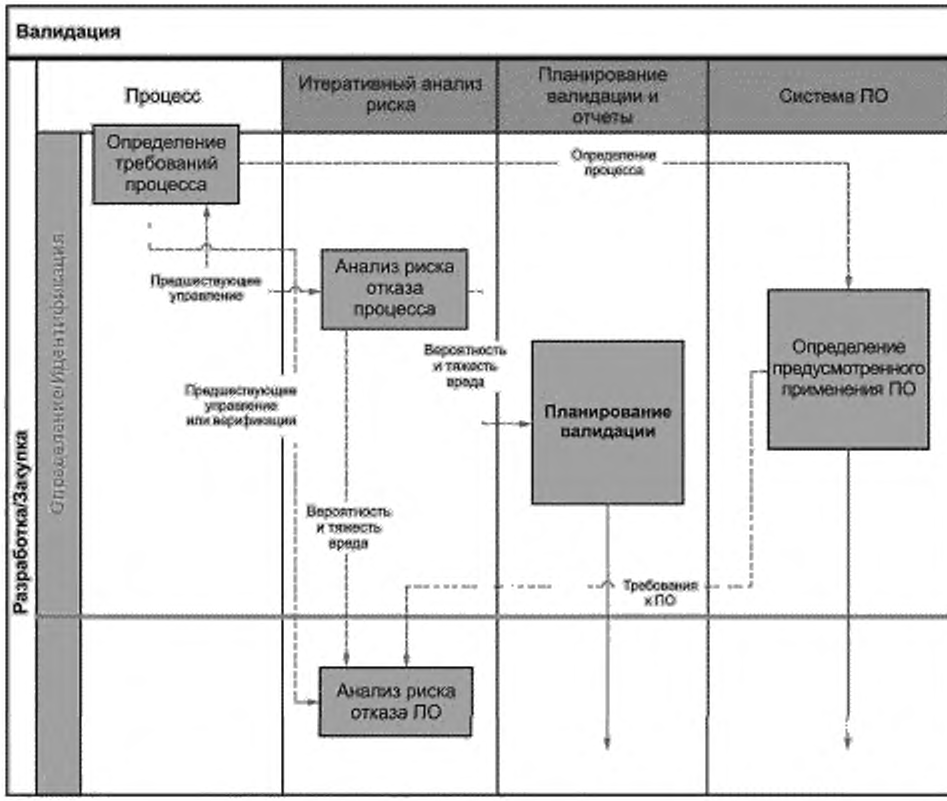


Рисунок 3 — Стадия жизненного цикла программного обеспечения в системе менеджмента качества: последовательность действий блока «Определение/Идентификация»

5.3.2.2 Требования к процессу

Первым шагом в применении средств управления жизненным циклом программного обеспечения в системе менеджмента качества является определение цели и функции всего процесса, особенно частей, управляемых программным обеспечением. Эту задачу лучше всего выполнять с привлечением соответствующих экспертов, включая все аспекты и связанную с ними деятельность независимо от того, все ли они будут управляться программным обеспечением. Преимущества такого подхода обосновываются следующим:

- могут быть четко определены регулирующие требования;
- может быть четко определено предусмотренное применение конкретного программного обеспечения в контексте процесса;
- элементы процесса и деятельность, которые не управляются конкретным программным обеспечением, могут быть четко идентифицированы и учтены в процедурах или другими способами;
- предшествующие программному обеспечению и последующие элементы процесса идентифицированы и могут учитываться при оценке рисков отказа программного обеспечения, а также при разработке мер по управлению риском отказа программного обеспечения.

Деятельность по определению процесса закладывает основу для решений, принимаемых на более поздних стадиях жизненного цикла программного обеспечения в системе менеджмента качества, и имеет важное значение для ориентации усилий на деятельность с добавленной стоимостью с учетом риск-ориентированного подхода.

5.3.2.3 Анализ риска отказа процесса

Связь программного обеспечения с итоговой безопасностью и результативностью медицинской продукции рассматривается в рамках процесса анализа риска. Следует также учитывать следующее:

- риск причинения вреда людям: это включает прямой вред пациентам и пользователям, а также косвенный вред, когда происходит отказ/сбой программного обеспечения, управляющего изготовлением или качеством изделия, что приводит к неисправности самого изделия и причинению вреда;

- регуляторный риск: риск несоблюдения регулирующих требований следует учитывать, если отказ программного обеспечения может привести к потере записей (например, корректирующих и предупреждающих действий, претензий, записей, относящихся к техническому файлу или файлу проектирования и разработки), требуемых регулирующими органами, или к отклонениям от процедур системы качества и производственных процедур;

- риск влияния на среду применения: риск для среды, как физической, так и виртуальной, в которой используется программное обеспечение.

В данную модель могут быть включены и другие типы рисков, однако область применения настоящего стандарта, как и рассматриваемые инструменты для снижения риска, их не учитывают. Настоящий стандарт фокусируется на определении рисков нарушения безопасности человека, регуляторных рисков и рисков воздействия на среду применения, связанных с отказом программного обеспечения в контексте отказа процесса.

Результаты анализа риска должны быть четко документированы, поскольку они являются важными факторами, влияющими на выбор инструментов из набора инструментов, а также на масштаб усилий, прилагаемых к деятельности по валидации.

5.3.2.4 Планирование валидации

Степень подтверждения и объем объективных свидетельств, необходимых для обеспечения последовательного выполнения требований к программному обеспечению, зависят от критической значимости программного обеспечения в рамках всего процесса. Таким образом, первое действие по планированию валидации, касающееся масштаба прилагаемых усилий и тщательности рассмотрения практических результатов, должно быть основано исключительно на выходных данных анализа риска отказа процесса.

Завершение первого действия по планированию валидации приводит к первой итерации документации по планированию валидации. Такое планирование включает как выбор «масштаба прилагаемых усилий» (т. е. решений), так и обоснование данного выбора (т. е. факторов принятия решений). Обоснование должно базироваться на риске причинения вреда в результате отказа процесса. План валидации должен предоставить объективные свидетельства применения подхода на основе критического мышления к процессу планирования валидации.

5.3.2.5 Предусмотренное применение программного обеспечения

5.3.2.5.1 Элементы предусмотренного применения

Предусмотренное применение программного обеспечения является основой для составления полной картины функциональности программного обеспечения и цели его использования в рамках процесса. В частности, предусмотренное применение программного обеспечения предназначено для описания и объяснения того, как программное обеспечение вписывается в общий процесс, который оно автоматизирует; что делает программное обеспечение; что можно ожидать от программного обеспечения; насколько можно полагаться на программное обеспечение для проектирования, производства и обслуживания медицинских изделий с точки зрения безопасности. Предусмотренное применение является ключевым инструментом в понимании того, какие потенциальные риски связаны с использованием программного обеспечения.

Предусмотренное применение программного обеспечения содержит три основных элемента:

- цель и назначение, которые связаны.

- с использованием программного обеспечения (например, кто, что, когда, почему, где и как);

- с соблюдением регулирующих требований при использовании программного обеспечения;

- с границами применения программного обеспечения в рамках процесса или с другим программным обеспечением и/или пользователями;

- требования к использованию программного обеспечения. Поскольку с увеличением сложности, как правило, возрастает риск, этот элемент предоставляет более подробную информацию относительно применения программного обеспечения (например, варианты использования, требования к пользователю);

- требования к программному обеспечению. По мере того как сложность и риск возрастают до такой степени, что разработчикам программного обеспечения необходимы четкие предписания, этот элемент предоставляет более конкретную и подробную информацию относительно ожиданий от программного обеспечения.

Предусмотренное применение программного обеспечения должно быть официально одобрено и находиться под управлением квалифицированного и опытного персонала в области регулирующих требований, требований системы качества, а также связанного с программным обеспечением процесса.

Поскольку валидация проводится для «предусмотренного применения», важно понимать, что полноценное проведение валидации невозможно без полноценного установления предусмотренного применения для программного обеспечения.

Далее приведены дополнительные сведения об элементах предусмотренного применения программного обеспечения.

5.3.2.5.2 Цель и назначение программного обеспечения

Описание цели и назначения программного обеспечения содержит информацию, охватывающую три аспекта: использование программного обеспечения, регуляторные аспекты использования и границы применения программного обеспечения.

a) Использование программного обеспечения

- При определении использования программного обеспечения необходимо учитывать следующие вопросы: «что?», «почему?», «как?», «кто?», «где?» и «когда?». Ответы помогают понять, как программное обеспечение используется для соответствия требованиям процесса. Как показано в таблице 1, такой механизм помогает выявить базовую информацию для определения предназначения программного обеспечения.

- Ответы, которые имеют значение для описания программного обеспечения, должны быть включены в документированное определение предполагаемого использования.

Таблица 1 — Примеры вопросов и ответов

Вопрос	Ответ
На решение какой проблемы направлено программное обеспечение?	Проблемы в эффективном и точном обобщении данных о дефектах продукта для анализа тенденций
Почему программное обеспечение полезно?	Программное обеспечение позволяет обобщать и анализировать данные из различных мест по всему миру
Как программное обеспечение решает проблему?	Программное обеспечение управляет процессом сбора данных и автоматически обобщает и анализирует информацию о тенденциях, или программное обеспечение не управляет процессом, но обеспечивает пассивный сбор данных, используемых для обобщения и анализа информации о тенденциях
Кто использует программное обеспечение?	Отделы обеспечения качества и эксплуатации
Где используется программное обеспечение?	Доступ к программному обеспечению осуществляется в США, Европе и Японии
Когда используется программное обеспечение?	Доступ к программному обеспечению осуществляется в рабочее время по всему миру (т. е. ежедневно, с понедельника по пятницу)
Примечание — Данные примеры вопросов не являются исчерпывающими.	

b) Регуляторные аспекты использования программного обеспечения

- При оценке регуляторных аспектов использования можно дополнительно изучить ответы на вопросы, чтобы определить, входит ли программное обеспечение в область применения настоящего стандарта (см. 5.2). Следует более подробно объяснить все ответы «да» с целью определения причины таких выводов. Когда программное обеспечение определено как входящее в область применения настоящего стандарта, необходимо определить любой потенциальный вред для людей (не являющихся пользователями медицинского изделия) или для окружающей среды. Предложенные далее вопросы акцентируют внимание пользователя настоящего стандарта на элементы, которые являются частью регулирующих требований, таких как область здравоохранения, безопасность и действительность или подлинность электронных записей и подписей.

- Как отказ или скрытые недостатки программного обеспечения могут повлиять на безопасность или качество медицинских изделий?

- Как программное обеспечение автоматизирует или выполняет действия, установленные в регулирующих требованиях, в частности требованиях к системам менеджмента качества медицинских изделий?

- Как программное обеспечение может причинить вред людям (не являющимся пользователями медицинского изделия) или окружающей среде?

c) Границы применения программного обеспечения

- Определение частей процесса, которые должны управляться с помощью программного обеспечения (границы внутри процесса), и областей, где существуют программные интерфейсы (границы с другим программным обеспечением), способствует результативности и эффективности усилий по валидации. Например, часто бывает более эффективно валидировать несколько программных продуктов как группу, а не проводить их валидацию по отдельности. Также следует рассмотреть различные стратегии группирования, которые могут влиять на эффективность текущих действий в последующей стадии обслуживания.

- Границы программного обеспечения в процессе

  - Идентификация границ программного обеспечения в рамках процесса четко устанавливает аспекты, которые должны быть включены в предусмотренное применение. Программное обеспечение может автоматизировать только часть или весь процесс, а также может служить хранилищем данных для процесса. Понимание роли, которую программное обеспечение играет в этом процессе, помогает определить риски, связанные с возможным отказом программного обеспечения.

- Границы с другим программным обеспечением

  - При внешнем взаимодействии с другим программным обеспечением, используемым в рамках систем качества медицинских изделий, или с программным обеспечением медицинских изделий (включая случаи, когда программное обеспечение само по себе является медицинским изделием) важно идентифицировать все интерфейсы между приложениями. Хотя масштаб усилий и трудозатраты по валидации обычно включают внутренние интерфейсы как неотъемлемую часть используемого метода, внешние интерфейсы программного обеспечения не должны игнорироваться. Все интерфейсы между программными приложениями должны быть охвачены процессом критического мышления.

5.3.2.5.3 Требования к использованию программного обеспечения

Требования к использованию программного обеспечения состоят из хорошо документированных и прослеживаемых элементов, которые обеспечивают дополнительный уровень детализации целей и задач программного обеспечения. Такие требования обеспечивают понимание сценариев использования системы с точки зрения потребностей пользователя или особенности требований к продукции. Требования пользователя могут быть выражены в форме пользовательских требований, вариантов использования или в другой ориентированной на пользователя форме. Перспектива развития требований к продукции может повлиять на требования к медицинским изделиям, на которые воздействует данная система, и может в некоторых случаях включать ссылки на конкретные требования к медицинским изделиям или краткую характеристику линейки медицинских изделий, на которые может повлиять программное обеспечение.

5.3.2.5.4 Требования к программному обеспечению

Установление требований к программному обеспечению состоит из документированной и прослеживаемой деятельности или действий по определению элементов. Результатом такой деятельности являются спецификации, устанавливающие, что программное обеспечение должно делать, чтобы соответствовать цели, назначению и требованиям к использованию программного обеспечения. Требования к программному обеспечению включают входные данные для проектирования системы, конфигурации системы, а также входные данные для деятельности по испытаниям (тестированию).

5.3.3 Имплементация, тестирование и развертывание

5.3.3.1 Требуемая деятельность

Деятельность, выполняемая в рамках блока имплементации, тестирования и развертывания, включает:

a) планирование уровня строгости проведения валидации в разрабатываемом проекте;

b) разработку и конфигурирование;

c) сборку программного обеспечения;

d) тестирование программного обеспечения с учетом идентифицированных рисков.

5.3.3.2 Анализ рисков отказа программного обеспечения

Ключевым моментом анализа риска отказа программного обеспечения являются определение и документирование свойственных рисков, связанных с отказом программного обеспечения, а также

идентификация любых мер по его управлению (включая элементы управления в отношении процесса и программного обеспечения вне анализируемого программного обеспечения). Затем этот анализ используется для нахождения практичного и результативного подхода к валидации.

При анализе рисков, связанных с отказом программного обеспечения, учитываются любые элементы управления процессом вне анализируемого программного обеспечения, которые представляют собой меры по управлению риском этого процесса. Эти меры по управлению риском процесса способствуют уменьшению критичности отказа, тем самым снижая зависимость от программного обеспечения, что, в свою очередь, снижает зависимость от тестирования (экспертизы) и документирования (сбора объективных свидетельств), проводимых в целях обеспечения безопасности функционирования программного обеспечения. Включение в анализ таких аспектов способствует обеспечению уверенности в том, что программное обеспечение рассматривается в рамках всего процесса в целом.

Модель, представленная в приложении В, не является всеобъемлющей формулой. Проведенный анализ предоставляет входные данные для выбора инструментов из набора инструментов, которые будут использоваться для валидации программного обеспечения.

5.3.3.3 Планирование валидации

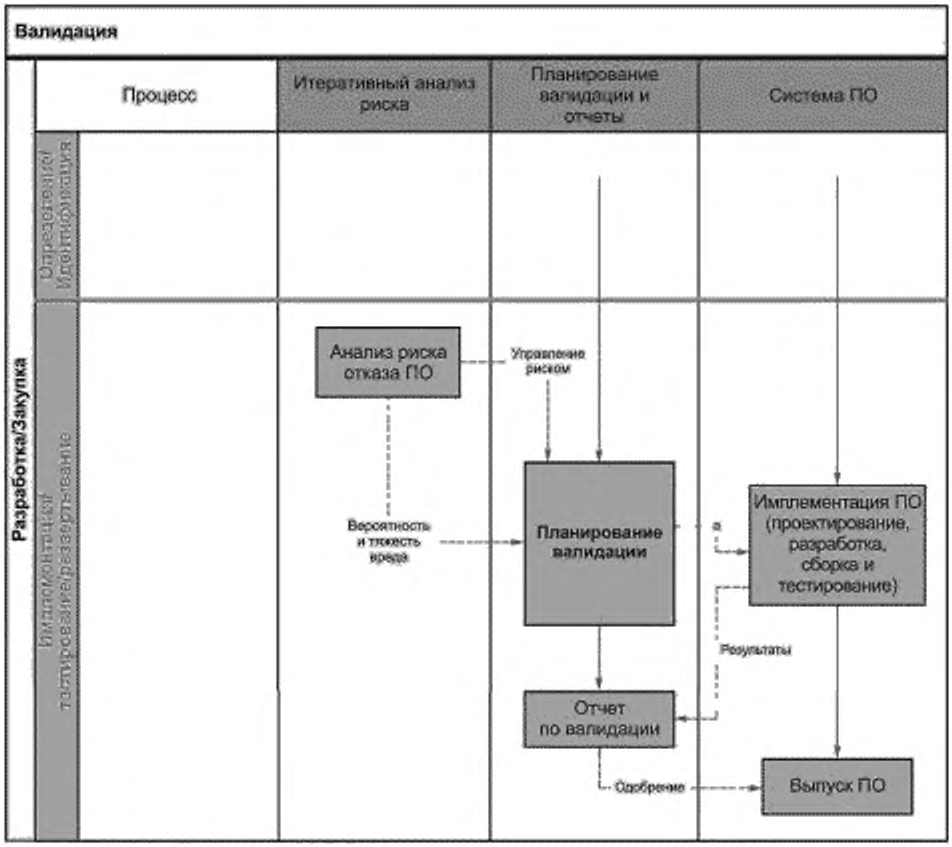
Планирование валидации использует выходные данные деятельности по определению предусмотренного применения и анализа риска программного обеспечения в качестве входных данных для идентификации мер по управлению риском и выбора инструментов из набора инструментов, которые будут использоваться для валидации программного обеспечения.

Важно обеспечить участие в процессе выбора инструмента квалифицированных специалистов, понимающих влияние отказов программного обеспечение на автоматизируемый процесс, а также связанные с ними риски, хотя эти специалисты не обязательно должны быть экспертами в области программного обеспечения. В процесс планирования сложного программного обеспечения или программного обеспечения, имеющего высокий риск, связанный с его отказом, должны быть вовлечены специалисты из разных областей (например, по вопросам регулирования, качества, клинических аспектов и т. п.).

Результатом планирования валидации является документированный план, в котором описываются принятые решения и причины их принятия. Планирование валидации предоставляет документированные свидетельства, обосновывающие выбор уместных дополнительных мероприятий по укреплению доверия к программному обеспечению и его функционированию, как предусмотрено.

5.3.3.4 Имплементация программного обеспечения (проектирование и разработка, сборка и тестирование)

Этот блок диаграммы включает применение многих инструментов из набора инструментов. Инструментарий (необходимая деятельность или действия, установленные в плане валидации) осуществляется на этапах проектирования, разработки, сборки и тестирования (см. рисунок 4).



а — включает меры по управлению риском, такие как проверка кода, таймер обеспечения безопасности и т. д. Также включает направление для областей тестирования и тип тестов, которые будут использоваться

Рисунок 4 — Стадия жизненного цикла программного обеспечения в системе менеджмента качества: последовательность деятельности по имплементации, тестированию и развертыванию программного обеспечения

5.3.3.5 Отчет по валидации

Вся деятельность, а в некоторых случаях и результаты этой деятельности должны быть указаны в окончательном отчете по валидации, когда завершены необходимые мероприятия в отношении укрепления доверия, включая выбор инструментов из набора инструментов, для обеспечения надлежащего функционирования программного обеспечения. Официальные анализ и одобрение отчета представляют собой краткое описание ссылок на все документированные объективные свидетельства, подтверждающие сделанный вывод о том, что программное обеспечение было валидировано на предмет его предусмотренного применения.

5.3.3.6 Выпуск программного обеспечения

По завершении валидации должен быть реализован официально установленный метод выпуска программного обеспечения. Установленный способ управления выпуском программного обеспечения должен обеспечивать и подтверждать, что введенное в эксплуатацию программное обеспечение соответствует тому, которое оценивалось в ходе деятельности по достижению доверия к программному обеспечению, что указывается в отчете по валидации. В противном случае обоснования и используемые средства управления выпуском должны обеспечивать и подтверждать тот факт, что выпущенное программное обеспечение является в достаточной степени работоспособным в предназначенной среде применения.

5.4 Стадия обслуживания

5.4.1 Переход в стадию обслуживания

Критерий входа в стадию: стадия технического обслуживания (поддержания) начинается после выпуска программного обеспечения для применения.

Область деятельности: на данном этапе деятельность направлена на обеспечение валидированного состояния программного обеспечения в случае внесения в систему разного рода изменений, которые должны быть под управлением. Некоторые типы изменений могут включать только изменения процесса, в котором используется программное обеспечение.

Изменения в любой валидированной системе должны находиться под управлением в соответствии с применяемыми изготовителем политиками и процедурами.

В идеальной ситуации рекомендуется валидировать вносимые изменения в тестовой среде до перезапуска производственной эксплуатации системы. Если валидация изменений в тестовой среде невозможна и тестирование изменений должно проводиться в производственной среде, то с целью минимизации нежелательного воздействия на производственную среду или на продукцию должны быть предусмотрены соответствующие средства управления.

Выбор используемых при валидации изменений инструментов из набора инструментов должен определяться посредством анализа влияния изменений программного обеспечения на существующие меры по управлению риском, или посредством анализа возникновения новых рисков, или обоими методами.

На практике, несмотря на усилия по управлению изменениями, фактически используемое программное обеспечение или его конфигурация со временем подвергаются изменениям, в связи с чем может быть целесообразным использование специфичных для данной стадии инструментов поддержки (например, периодического мониторинга фактического применения или мониторинга конфигурации программного обеспечения в режиме реального времени). Если результатом изменения в предусмотренном применении становится более высокий уровень риска, то такое изменение может потребовать более обширного набора действий по валидации, чем было выполнено первоначально (даже при отсутствии изменений в самом программном обеспечении).

С целью обеспечения наличия объективных свидетельств поддержания валидированного состояния программного обеспечения, решения изготовителя в отношении выбора более обширных действий по валидации и сбора свидетельств их выполнения должны быть задокументированы как часть планирования валидации.

5.4.2 Планирование обслуживания

Записи, свидетельствующие о планировании проведения обслуживания, должны быть составлены до перехода программного обеспечения в стадию обслуживания.

В идеальной ситуации планирование обслуживания должно начинаться на стадии разработки. Нужно точно понимать, как изменения могут повлиять на статус валидации программного обеспечения, изучить влияние изменений на риск и спланировать деятельность для поддержания статуса валидации.

В случае большого и сложного программного обеспечения действия по ежедневному обслуживанию и настройке функционирования программного обеспечения могут выполняться, не затрагивая при этом способность программного обеспечения работать надлежащим образом в соответствии с предусмотренным применением. Планирование обслуживания на этапе разработки может определить, какие действия в условиях эксплуатации могут быть выполнены без ущерба статусу валидации, а какие изменения в них требуют проведения работ по поддержанию этого статуса. До перехода программного обеспечения в стадию обслуживания следует запланировать и обсудить способы определения периодичности выполнения дальнейших действий по валидации программного обеспечения, а также то, как изменения в базовом программном обеспечении (например, в операционной системе, системе управления базами данных) могут повлиять на валидированное состояние программного обеспечения. Полезно обучить операторов программного обеспечения распознавать границы и разницу между обычными действиями (деятельностью) по эксплуатации и любыми изменениями в них, требующими валидации.

Анализ прослеживаемости является полезным инструментом в отношении менеджмента деятельности по обслуживанию. Анализ прослеживаемости часто является фундаментом первоначальной валидации и часто осуществляется с помощью матрицы прослеживаемости. Такая матрица устанавливает связь между требованиями к проведению тестирования или другой деятельности по верификации с мерами по управлению риском и т. д. При разработке матрицы прослеживаемости во время первоначальной имплементации программного обеспечения анализ прослеживаемости становится ценным инструментом при

проведении обслуживания, помогая в идентификации влияния изменений и соответствующих действий по валидации изменений. В простом программном обеспечении такой анализ может состоять из одноуровневого прослеживания требований к имплементации и требований по верификации. Однако для сложного программного обеспечения может потребоваться многоуровневая матрица, которая подвергает декомпозиции функциональные возможности верхнего уровня на требования более низкого уровня, а затем на требования к исполнению и верификации. Также в матрицу прослеживаемости может быть включена другая информация (например, разделы программного обеспечения с наиболее высокими рисками, которые могут быть обозначены в ней с указанием дополнительной деятельности по валидации).

5.4.3 Типы обслуживания, проводимые в рамках стадии

Программное обеспечение может измениться после его выпуска для применения по нескольким причинам. Некоторые из наиболее распространенных типов изменений на стадии обслуживания включают следующее:

- корректирующие изменения при обслуживании для исправления ошибок и отказов в программном обеспечении;
- улучшающие изменения при обслуживании для повышения производительности, удобства обслуживания или других характеристик программного обеспечения;
- адаптивное сопровождение для обновления операционной среды программного обеспечения (например, изменения операционной системы, системного оборудования или приложений, с которыми данное программное обеспечение взаимодействует).

5.4.4 Изменения процесса: переход к мерам по управлению риском

Если изменениям подвергается процесс, который полностью или частично управляется программным обеспечением, то следует провести анализ влияния этих изменений с целью пересмотра мер по управлению риском.

Процесс, который полностью или частично управляется программным обеспечением, может изменяться независимо от программного обеспечения. При изменении процесса важно понимать, как это изменение повлияет на статус валидации (валидированное состояние) программного обеспечения. Изменение процесса может повлиять на предусмотренное применение программного обеспечения или на другую связанную с ним поддерживающую информацию.

Изменение процесса может повлиять на меры по управлению риском, применяемые к программному обеспечению, которые являлись частью обоснования решения о проведении валидации. Поскольку программное обеспечение является составной частью процесса, предпринимаемое последующее управляющее воздействие может быть важной мерой по управлению риском в отношении программного обеспечения. Анализ влияния предполагаемого изменения процесса упростится, если все предпринимаемые последующие действия по управлению риском правильно идентифицированы как часть обоснования валидации программного обеспечения и как часть определения (идентификации) процесса. Анализ влияния изменений важен для правильного проведения обслуживания с целью укрепления доверия как к процессу, так и к программному обеспечению.

5.4.5 Экстренные изменения

Экстренные изменения должны проходить в рамках официально одобренных процессов. Такие процессы должны требовать обоснования в отношении разработки и имплементации; механизмов получения и регистрации разрешения на развертывание изменения; положений, обеспечивающих надлежащую оценку и управление риском; любых действий, необходимых для осуществления экстренных изменений (например, обучение, коммуникация, анализ продукции и ее местонахождение). Выполнение положений для правильной оценки и управления риском в таких обстоятельствах сводится к минимально необходимому набору деятельности для обеспечения соответствия регулирующему требованию по валидации изменений перед выпуском.

Изменения в программном обеспечении могут выполняться при возникновении чрезвычайных обстоятельств. В основном такие изменения требуются, если была нарушена целостность программного обеспечения, операционной системы или данных, а также если можно снизить последствия возникновения ситуаций, ведущих к причинению вреда с высокой степенью тяжести.

Для проведения полного оценивания всех последствий экстренных изменений могут потребоваться дополнительные действия. В зависимости от совокупного риска, вызванного отказом процесса, его результаты (данные или продукция) могут нуждаться в дополнительном управлении до тех пор, пока не будут завершены все дополнительные действия после экстренных изменений.

Выявление проблемы с программным обеспечением, вызывающей прерывание процесса, не вызывает затруднений. Выявление скрытых проблем может быть более сложным. Периодический анализ

журналов ошибок, запросов службы поддержки, отзывов клиентов и других отчетов о дефектах может указывать на глубинные проблемы. Перечисленные методы мониторинга могут идентифицировать проблемы, которые не являются достаточно очевидными и не приводят к сообщению об ошибке, но которые могут указывать на исправимые проблемы программного обеспечения. В такой ситуации могут потребоваться действия по обслуживанию (поддержке), предпринимаемые для решения идентифицированных проблем посредством внесения исправлений в последующие выпуски программного обеспечения. Одновременно с этим могут быть оперативно урегулированы проблемы в уже выпущенном программном обеспечении, относящиеся к таким типам программных проблем.

После того как действия по обслуживанию (поддержке) приведут к исправлению проблем для будущих выпусков, следует проанализировать историю влияния выявленных дефектов в ранее выпущенном программном обеспечении и осуществлять менеджмент в отношении их последствий.

Периодическая оценка результативности обучения пользователя является еще одним методом мониторинга, помогающим поддерживать валидированное состояние, если валидация программного обеспечения зависит от обеспечения его правильного использования посредством обучения.

5.4.6 Поддержка предусмотренного применения

Любое изменение предусмотренного применения программного обеспечения должно быть валидировано для нового предусмотренного применения. В противном случае использование программного обеспечения должно быть сочтено как несанкционированное и его использование должно быть прекращено. В последнем случае необходимо провести оценку риска и убедиться, что в течение периода несанкционированного использования не возникло рисков.

Изменения в предусмотренном применении требуют особого внимания, т. к. они могут быть как совершенно очевидными, так и малозаметными и трудными в обнаружении. К малозаметному изменению можно отнести изменение цели и предназначения или требований к использованию программного обеспечения, что не обязательно приводит к изменению конкретных элементов требований к программному обеспечению (см. 5.3.2.5). Такое изменение может произойти как преднамеренно, так и в результате использования существующего программного обеспечения в новом режиме, без осознания факта работы в условиях измененного предусмотренного применения. Помимо указанных изменений сами пользователи могут начать применять программное обеспечение не так, как было предусмотрено изначально. Из-за этого изменения развернутое программное обеспечение больше не находится в валидированном состоянии.

При инициализации изменений в валидированном программном обеспечении должно анализироваться его предусмотренное применение с целью обеспечения того, что оно все еще согласуется с фактическим использованием программного обеспечения.

5.5 Стадия вывода из эксплуатации

На этой стадии следует задокументировать факт вывода из эксплуатации программного обеспечения и установить методы доступа к любым связанным электронным записям в течение любых требуемых сроков хранения записей.

Деятельность по выводу программного обеспечения из эксплуатации в значительной степени зависит от типа программного обеспечения, подлежащего выводу. Некоторые из них могут просто выполнять какие-либо действия и не сохранять никаких данных. Другое программное обеспечение может быть гораздо более сложным (например, система прослеживаемости или управления документами, в которых хранятся объемы данных, связанных с продукцией и с ее соответствием требованиям). В случае сохраняющего данные программного обеспечения следует составить план обработки данных. Некоторые вопросы для рассмотрения включают в себя следующее:

- Есть ли программное обеспечение, которое заменяет устаревшее программное обеспечение?
- Можно ли перенести имеющиеся данные в новое программное обеспечение?
- Должны ли данные быть перенесены в специальный формат для долгосрочного хранения?
- Каковы требования к хранению данных какого-либо типа?
- Будут ли данные храниться на постоянных носителях?
- Если да, каковы инструкции или процедуры хранения и могут ли быть получены целостные данные, содержащие все связанные требования к этим данным?
- Какова процедура поддержания постоянных носителей и программного обеспечения, способного их читать?
- Будет ли храниться аппаратное обеспечение для использования и восстановления выведенного из эксплуатации программного обеспечения?

- Как будет обслуживаться хранящееся аппаратное обеспечение?
- Нужно ли будет обращаться к устаревшему программному обеспечению в рамках изучения жалоб/претензий или в рамках работы с корректирующими и предупреждающими действиями?
- Потребуются ли описанное выше аппаратное обеспечение и вышедшее из эксплуатации программное обеспечение для повторной имплементации программного обеспечения?

6 Документация

Следует обеспечить управление и надлежащее документирование всей информации, связанной с деятельностью по управлению жизненным циклом программного обеспечения в системе менеджмента качества.

Существуют два важных преимущества разработки адекватной и качественной документации.

a) Четкая формулировка полного определения программного обеспечения в создаваемой документации позволит детально понять предусмотренное применение программного обеспечения, ожидаемые результаты, а также воздействие любых изменений, внесенных в программное обеспечение.

b) Записи по планированию и проведению валидации предоставят документированные свидетельства решений, принятых в результате использования критического мышления. Создание документации, сфокусированной на проведенное оценивание или анализ, а также на результаты выбора инструментов, нацеленных на риск-ориентированные и значимые действия по достижению доверия к программному обеспечению, обеспечивает четкое понимание проведенной валидации. Такая документация кратко излагает соответствие критериям приемки и предоставляет свидетельства того, что выполненные действия обеспечивают должную работу программного обеспечения, функционирование которого создает только допустимые уровни риска в отношении автоматизируемого процесса.

Объем составляемой документации напрямую связан с масштабом усилий, приложенных к валидации программного обеспечения, который должен быть соизмерим с риском. Таким образом, в рамках подхода к валидации программного обеспечения, рассматриваемого в настоящем стандарте, объем документации основывается на последствиях отказа процесса. Чем выше риск причинения вреда людям или окружающей среде от рассматриваемого процесса, тем значительней ожидаемый объем создаваемой документации. Кроме того, более высокий риск причинения вреда должен сопровождаться более тщательным анализом документации со стороны специалистов разного профиля и/или высшего руководства организации.

Структура информации по управлению жизненным циклом программного обеспечения в создаваемой документации в системе менеджмента качества может отличаться в зависимости от многих факторов (например, от используемой технологии, размера или сложности программного обеспечения).

Информация должна быть организована таким образом, чтобы облегчить аудит информации наряду со способностью поддерживать свидетельство валидированного состояния в течение стадии обслуживания жизненного цикла программного обеспечения в системе менеджмента качества.

Способ сбора и документирования информации по управлению жизненным циклом программного обеспечения в системе менеджмента качества зависит от предпочтений и установленных политик сторон, выполняющих валидацию. Способ оформления и представления объективных свидетельств управления жизненным циклом программного обеспечения в документации в системе менеджмента качества выбирается по усмотрению сторон, проводящих валидацию программного обеспечения. С точки зрения предстоящего анализа соответствия документация по планированию валидации и отчетности должна быть составлена таким образом, чтобы объединить информацию обо всех уместных мероприятиях по укреплению доверия к программному обеспечению, которые были запланированы и выполнены с целью обеспечения должного функционирования программного обеспечения. По сути, эта документация является главной записью о принятых решениях на основе входных данных (драйверов решений), которые воплощают в себе процесс критического мышления, используемого для подтверждения факта разработки законченного программного решения, соответствующего целям регулирования и учитывающего ключевые заинтересованные стороны и их потребности.

Примечание — Термин «документация» используют для обозначения объема записываемой информации, независимо от способа и носителя (в фактическом документе или в инструментах, которые собирают информацию, таких как инструменты управления требованиями).

7 Обязательные процессы

Представленная в настоящем стандарте методология предназначена для полноценного использования в рамках действующей и результативной системы менеджмента качества медицинских изделий.

Аспекты системы качества, которые могут оказать наиболее положительное влияние на успех применения методологии критического мышления, включают в себя: менеджмент ресурсов (человеческие и аппаратные), менеджмент изменений (включая менеджмент конфигурации) и управление поставщиками. Детализация этих аспектов выходит за рамки настоящего стандарта; каждый аспект рассматривается в других стандартах и отраслевых документах (см. Библиографию). Кроме того, настоящий стандарт не предназначен для сопоставления устанавливаемой в нем деятельности с процессами и ролью системы качества (например, обеспечение качества, менеджмент и производство). Философия и кадровая инфраструктура каждой конкретной организации будут определять приемлемые роли для выполнения деятельности по валидации.

Приложение А
(справочное)

Набор инструментов

А.1 Введение

Этот набор инструментов предоставляет перечень видов деятельности по укреплению доверия к функционированию программного обеспечения, которая может быть проведена для достижения цели установления требований по его валидации. Перечень не является исчерпывающим и представляет собой стартовый набор, основанный на текущих знаниях в области разработки программного обеспечения. Некоторые из этих видов деятельности частично совпадают или могут использоваться совместно (например, испытание в нормальных условиях часто является частью тестирования системы программного обеспечения, при условии, что основное внимание уделяется ценности этой деятельности). Эти виды деятельности должны использоваться в качестве основы для планирования и выполнения валидации.

Выбор и осуществление того или иного вида деятельности должны соответствовать риску, связанному с программным обеспечением. С целью обоснования сделанного выбора деятельность в наборе инструментов категорирована и обозначена согласно следующим схемам:

- полная: выполнение всей деятельности в любом случае;
- адаптированная: выбор и осуществление уместных/подходящих частей деятельности;
- выборочная: выбор и осуществление необходимой деятельности.

Инструменты могут быть подобраны для определения видов деятельности, подходящих для использования в рамках конкретной организации, и могут изменяться или совершенствоваться по мере изменения технологий и расширения опыта, внедряя, таким образом, новые передовые методы разработки программного обеспечения. Там, где применимо, некоторые виды деятельности также необходимо будет установить в стандартных процедурах.

А.2 Структура набора инструментов

Для удобства и простоты восприятия деятельность подразделяется на пять основных процессов жизненного цикла программного обеспечения в системе менеджмента качества. С целью выбора и идентификации наиболее подходящей деятельности, в зависимости от объема и характера программного обеспечения, на различных стадиях жизненного цикла программного обеспечения в системе менеджмента качества должен применяться подход на основе критического мышления.

В отношении каждой деятельности, приведенной в таблице А.1, приводится ее краткое описание с пояснением вклада, который эта деятельность привносит в процесс валидации. Правый столбец также содержит примеры методов, которые можно использовать для выполнения поименованной в левом столбце деятельности.

Таблица А.1 — Стадия разработки: определение

Деятельность	Определение
Определение требований к процессу (полная схема)	Деятельность по определению рассматриваемого производственного процесса или процесса системы качества для его частичной или полной автоматизации посредством использования программного обеспечения. Деятельность также описывает любые верификационные или предупреждающие меры в рамках процесса, которые можно учитывать при выполнении анализа риска процесса или программного обеспечения. Выходные данные этой деятельности могут быть документированы в виде схемы потока процесса или сформулированных требований, которые устанавливают деятельность, выполняемую в рамках бизнес-процесса, процесса производства или системы качества
Анализ риска отказа процесса (полная схема)	Деятельность по установлению степени влияния отказа процесса на безопасность и результативность изделия, производственный персонал, окружающую среду или систему качества
Предусмотренное применение (адаптированная схема)	Для простого программного обеспечения результаты этой деятельности могут состоять из нескольких предложений или пунктов. Для большого и сложного программного обеспечения эта деятельность может привести к созданию обширной документации и подробных требований к программному обеспечению. Риск также является важным фактором при установлении масштаба этой деятельности. Элементы предусмотренного применения: - цель и назначение программного обеспечения; - требования к использованию программного обеспечения; - требования к программному обеспечению

Окончание таблицы А.1

Деятельность	Определение
Планирование валидации (полная схема)	Планирование валидации выполняется в два этапа: - на стадии разработки — определения устанавливают уровень детализации и масштаб усилий, которые будут определены в документации по проведению валидации, а также уровень строгости проверки и выбора деятельности, которая будет включена в стадию определения; - в последующем, на этапе выполнения, выбирают подходящую деятельность по валидации на основе решений, принятых на этапе определения, и соответствующую деятельность по анализу рисков. Результатом планирования валидации является план, описывающий деятельность, которая будут выполняться для обеспечения уверенности в последовательном соответствии программного обеспечения требованиям его предусмотренного применения
Официальный анализ требований к программному обеспечению (выборочная схема)	Деятельность (процесс, обсуждение и т. д.), при осуществлении которой заинтересованные стороны рассматривают и согласовывают требования к программному обеспечению на основе его предусмотренного применения
Выбор модели жизненного цикла разработки программного обеспечения в системе менеджмента качества (выборочная схема)	Деятельность по определению методологии и средств управления жизненным циклом, которые будут использоваться на этапе разработки полного жизненного цикла программного обеспечения в системе менеджмента качества. Обычно требуется только для сложного или программного обеспечения с повышенным риском. Для некоторого программного обеспечения применение МЭК 62304:2006 (включая изменение 1:2015) в качестве стандарта на процессы может быть особенно полезным для некоторых видов программного обеспечения
Планирование менеджмента риска (полная схема)	Деятельность связана с планированием применения менеджмента риска к программному обеспечению. Результатом планирования менеджмента риска является план, устанавливающий подход к анализу проблемных областей программного обеспечения относительно риска, а также выбор методов, с помощью которых можно анализировать риски, например анализ видов и последствий отказов (FMEA), анализ дерева неисправностей или другие инструменты анализа
Идентификация мер по управлению риском в производственном или бизнес-процессе (полная схема)	Эта деятельность представляет собой механизм для идентификации мер по управлению риском или опасностей (например, процедурный контроль). Он предполагает постоянный мониторинг, проводимый с целью обеспечения того, что элементы управления реализованы и результативны

Таблица А.2 — Стадия разработки: имплементация

Деятельность	Определение
Анализ отказов программного обеспечения (анализ риска) (полная схема)	Анализ отказов (сбоев) программного обеспечения сводится к установлению степени их влияния на соответствующий процесс и проблемных аспектов, идентифицированных при анализе отказов процесса
Анализ документированной архитектуры программного обеспечения (выборочная схема)	Архитектура программного обеспечения устанавливает высокоуровневую структуру элементов программного обеспечения и взаимосвязь между ними. Анализ документированной архитектуры позволяет проверить правильность, полноту и способность выполнять реализуемые программным обеспечением функции
Спецификация проекта (выборочная схема)	Представляет собой точное изложение того, как будут реализованы требования к программному обеспечению. Обычно включает в себя структуру программного обеспечения или компонентов, алгоритмы, логику управления, структуры данных, информацию об использовании набора данных, форматы ввода и вывода, описания интерфейсов и т. п.
Анализ проектирования и разработки (выборочная схема)	Анализ проводится для оценки прогресса, технической адекватности и разрешения рисков выбранного подхода к проектированию для одного или нескольких элементов конфигурации

Окончание таблицы А.2

Деятельность	Определение
Идентификация мер по управлению риском в рамках разработки программного обеспечения (полная схема)	Эта деятельность идентифицирует меры по управлению риском или опасности, которые были идентифицированы в ходе оценки рисков. Идентификация мер по управлению риском должна быть итеративным процессом, позволяющим осуществлять непрерывный мониторинг и обеспечивать внедрение и результативность разработанных средств управления (например, процедурные средства контроля, аппаратное резервирование)
Анализ кода или верификация кода (выборочная схема)	Анализ основан на экспертной оценке исходного кода программного обеспечения, предназначенной для поиска и устранения дефектов и улучшения общего качества кода. Анализ кода и общее качество кода могут быть улучшены путем установления и соблюдения набора общих стандартов кодирования
Анализ прослеживаемости (выборочная схема)	Этот анализ касается прослеживаемости требований к проектированию, кодированию, тестированию, анализу рисков или опасностей, а также к мерам по управлению риском. Может также включать прослеживаемость в отношении требований к процессу
Аудит поставщиков (выборочная схема)	Такой аудит представляет собой оценку системы качества поставщиков (вендоров) программного обеспечения в объеме, необходимом для обеспечения уверенности в их способности поставлять безопасное и пригодное для использования программное обеспечение. Возможны различные методы аудита поставщиков

Таблица А.3 — Стадия разработки: испытания (тестирование)

Деятельность	Определение
Планирование проведения испытаний (выборочная схема)	Планирование проведения испытаний должно устанавливать общий подход к этой деятельности, результаты которой помогают укрепить уверенность в том, что программное обеспечение соответствует предусмотренному применению. Тем не менее одного тестирования программного обеспечения может быть недостаточно для установления уверенности в том, что программное обеспечение пригодно для использования по назначению. В целях обеспечения комплексного подхода к валидации, в сочетании с тестированием, могут потребоваться другие методы или способы верификации. Уровень тестирования должен основываться на факторах риска и обеспечивать соответствующий уровень уверенности в том, что программное обеспечение удовлетворяет установленным требованиям, а также проектным спецификациям в соответствии с подходящими методами испытаний. Такие испытания могут включать тестирование разработчиком, модульное тестирование, интеграционное тестирование, пользовательское тестирование, нагрузочное тестирование, эксплуатационное тестирование и т. п.
Модульное тестирование (выборочная схема)	Тестирование, которое проводится в целях верификации реализованного в проекте одного элемента программного обеспечения (таких как программный элемент или программный модуль) или набора элементов программного обеспечения
Верификация данных (выборочная схема)	Верификация данных относится к деятельности, выполняемой для подтверждения правильности данных. Может быть выполнена как часть процесса переноса, преобразования или тестирования данных, а также независимо. По возможности следует использовать статистическую выборку
Интеграционное тестирование (выборочная схема)	Представляет собой упорядоченную последовательность тестирования, в котором программные элементы и (или) аппаратные элементы объединяются и тестируются для оценивания их взаимодействия до тех пор, пока программное обеспечение не будет интегрировано
Тестирование по сценариям использования (выборочная схема)	Представляет собой вид функционального тестирования, который игнорирует внутренний механизм или структуру системы/компонента и фокусируется на выходных данных, генерируемых в ответ на выбранные входные данные и условия выполнения. Каждый сценарий использования (применения) может иметь соответствующие входные параметры, каждый параметр может иметь набор значений, определенных для имитации фактических условий использования. Ряд сценариев применения может быть связан с использованием заранее определенных потоков, которые описывают последовательность, выполняющую некоторую задачу

Продолжение таблицы А.3

Деятельность	Определение
Тестирование программных интерфейсов (выборочная схема)	Относится к утверждению интерфейса между программными приложениями с учетом полного пути передачи данных от выхода к входу. Может быть выполнено путем прямого тестирования или 100%-ной верификации данных. Деятельность по тестированию должна включать стратегии, которые обеспечивают должную работу интерфейса в рамках требований спецификации или в границах условий как для нормальных, так и для отклоняющихся от нормы случаев
Регрессионное тестирование (выборочная схема)	Заключается в повторном выполнении тестов, которые программа ранее выполняла правильно, с целью обнаружения ошибок, вызванных изменениями или исправлениями, которые произошли во время разработки и обслуживания программного обеспечения
Набор тестов, предоставленный поставщиком (выборочная схема)	Поставляемые поставщиком (вендором) наборы тестов могут тестировать все возможности программного решения и обеспечить уверенность в производительности программного обеспечения в среде конечного использования. Однако такие наборы должны оцениваться на предмет соответствия установленному предусмотренному применению и полноте тестирования, включая тестирование для любых установленных мер по управлению риском. Для использования такого набора может потребоваться договорное соглашение, требующее от поставщика поддерживать набор тестов в течение всего срока службы программного обеспечения
Тестирование системы программного обеспечения (выборочная схема)	Представляет собой процесс тестирования интегрированной системы аппаратного и программного обеспечения для верификации его соответствия установленным требованиям. Такое тестирование может проводиться как в среде разработки, так и в запланированной среде применения. Валидация программного обеспечения отличается от системного тестирования программного обеспечения. Валидация программного обеспечения подтверждает пригодность программного обеспечения для использования его в предусмотренной среде применения и только предусмотренными пользователями. Системное тестирование программного обеспечения верифицирует только то, что требования к программному обеспечению были успешно выполнены. Для производственных систем, управляемых программным обеспечением, процесс валидационного тестирования может охватывать некоторые или все из этих тестов. Для применяемого в системах качества программного обеспечения выполнение всех шагов (требуемых рабочей инструкцией по работе с программным обеспечением) может покрывать требования к тестированию программного обеспечения
Тестирование по сценариям использования (выборочная схема)	Относится к тестированию, выполняемому на основе сценариев использования, включающих альтернативные последовательности и ситуации возникновения ошибок, которые устанавливаются в этих сценариях использования
Испытание в нормальных условиях (выборочная схема)	Эта деятельность представляет собой тестирование с использованием обычных входных данных
Тестирование устойчивости (Стрессовое тестирование) (выборочная схема)	Тестирование устойчивости должно показать, что программный продукт функционирует должным образом при вводе неожиданных и (или) некорректных данных. Проводится для оценивания системы или компонента в пределах или за пределами указанных требований. Методы идентификации достаточного набора таких тестов включают разделение по классам эквивалентности, анализ граничных значений и идентификацию особых случаев (угадывание ошибок)
Тестирование с фокусом на выходные данные (выборочная схема)	Эта деятельность представляет собой выбор тестовых входных данных и оценивание того, что выбранные (или все) выходы правильно сгенерированы системой. Тестирование с фокусом на выходные данные включает в себя создание набора тестовых сценариев, предназначенных для получения определенных выходных данных в системе. Акцент делается на создании желаемых выходных данных, а не на входных данных, которые инициировали ответ системы
Тестирование набором входных данных (выборочная схема)	Метод тестирования, с помощью которого формируется комбинация входных данных, с которыми программный блок или система могут столкнуться во время работы

Окончание таблицы А.3

Деятельность	Определение
Бета-тестирование (выборочная схема)	Тестирование программного обеспечения изготовителем в реальной среде применения на небольшом наборе пользователей
Тестирование рабочих характеристик (выборочная схема)	Измеряет, насколько хорошо система работает в соответствии с требуемым временем отклика, использованием центрального процессора (ЦП) и другими количественными характеристиками

Таблица А.4 — Стадия разработки: развертывание

Действие	Определение
Анализ используемых пользователем процедур (выборочная схема)	Проверка пользовательских процедур и инструкций, связанных с использованием программного обеспечения. Такой анализ обеспечивает установление применения программного обеспечения надлежащим образом
Внутреннее обучение по применению (выборочная схема)	Относится к документируемым учебным мероприятиям, характерным для программного обеспечения
Установочная квалификация (выборочная схема)	Означает установление уверенности в том, что программное обеспечение установлено и функционирует в соответствии с документированными инструкциями по установке (IQ)
Операционная и эксплуатационная квалификация (в процессе осуществления валидации) (выборочная схема)	Операционная квалификация (QQ) означает достижение уверенности в том, что производственный процесс и связанные с ним системы способны последовательно функционировать в установленных пределах и допусках. Эксплуатационная квалификация (PQ) означает достижение уверенности в результативности и воспроизводимости процесса
Окончательные приемочные испытания (выборочная схема)	Окончательные приемочные испытания относятся к тестам, применяемым к системе непосредственно перед финальным развертыванием. Также известны как тестирование ввода в эксплуатацию
Аттестация оператора (выборочная схема)	Является подтверждением компетентности обученных лиц

Таблица А.5 — Стадия обслуживания

Деятельность	Определение
Планирование обслуживания (адаптированная схема)	Связанные с планированием способы технического обслуживания заключаются в следующем: - Перспективное планирование. Этот метод охватывает предварительное планирование и прогнозирование изменений в программном обеспечении. Может использоваться во время первоначальной реализации программного обеспечения перед входом в фазу обслуживания, а также в любое время в течение фазы обслуживания. - Планирование предстоящих изменений. Этот метод касается планирования ожидаемых изменений программного обеспечения. Обычно фокусируется на деятельности, характерной для ожидающихся изменений. Данный вид планирования выполняется на этапе обслуживания программного обеспечения
Анализ известных проблем (выборочная схема)	Процесс, в котором все известные поставщику (вендору) проблемы с программным обеспечением оцениваются с точки зрения их влияния на применение или валидированное состояние установленного программного обеспечения
Тестирование совместимости (выборочная схема)	Процесс установления способности двух или более программных систем обмениваться информацией
Анализ совместимости инфраструктуры (выборочная схема)	Процесс определения того, как изменения в программной инфраструктуре могут повлиять на установленное программное обеспечение. Изменения могут включать изменения в оборудовании или в расположении системы

Окончание таблицы А.5

Деятельность	Определение
Мониторинг системы (выборочная схема)	Включает в себя методы, используемые для оценки общего состояния системы программного обеспечения в стадии обслуживания жизненного цикла программного обеспечения в системе менеджмента качества. Методы мониторинга системы могут включать: - периодическую оценку изменений предусмотренного применения; - фактическое использование конечными пользователями; - оценку результативности обучения; - анализ дефектов; - аудит данных
Процессы резервного копирования и восстановления (выборочная схема)	Включают резервное копирование системы, хранение и сохранность резервных копий, а также процедуры по восстановлению данных с носителя резервного копирования
Операционные средства управления (выборочная схема)	В дополнение к процессам резервного копирования и восстановления, мониторингу и отчетности с целью обеспечения должного функционирования программного обеспечения могут использоваться операционные средства управления. Общие методы включают: - безопасность; - управление правами доступа; - администрирование базы данных; - архивирование; - планирование на случай чрезвычайных обстоятельств
Регрессионный анализ (выборочная схема)	Включает такие задачи, как анализ прослеживаемости или анализ воздействия. Проводится для установления необходимой деятельности для поддержания статуса валидации (валидированного состояния) системы

Приложение В
(справочное)

Менеджмент риска и риск-ориентированный подход

В.1 Введение

В основной части настоящего стандарта было упомянуто, что масштаб усилий и уровень строгости проведения валидации определяются риском, связанным с программным обеспечением.

Более детальная информация приведена в ИСО 14971. ИСО 14971 описывает процесс менеджмента риска, который должен применяться к медицинским изделиям. Базовые принципы и терминология могут также применяться к программному обеспечению, подпадающему под область применения ИСО 14971.

В.2 Терминология

Перечисленные ниже определения заимствованы из ИСО 14971 или основаны на определениях из ИСО 14971:

- опасность: потенциальный источник вреда;
- опасная ситуация: обстоятельства, при которых люди, имущество или окружающая среда подвергаются одной или нескольким опасностям;
- риск: сочетание вероятности причинения вреда и тяжести этого вреда;
- вред: физическая травма или ущерб здоровью людей, ущерб имуществу или окружающей среде;
- тяжесть: мера возможных последствий опасности;
- мера по управлению риском: меры, с помощью которых риски снижаются или поддерживаются в пределах установленных уровней.

В.3 Основные принципы

Основным принципом является снижение до допустимого уровня рисков, связанных с программным обеспечением. Для этого изготовителю необходимо идентифицировать возможные опасные ситуации, связанные с применением программного обеспечения, оценить связанные риски и их соответствие критериям допустимости, которые устанавливаются изготовителем (если не установлены регулирующими требованиями).

Само по себе программное обеспечение не может нанести вред самостоятельно, следовательно, менеджменту риска должен быть подвергнут весь процесс, который управляется программным обеспечением.

В.4 Идентификация опасных ситуаций и определение рисков

В соответствии с подходом ИСО 14971, после установления предусмотренного применения следует идентифицировать возможные опасности и опасные ситуации, а также определить соответствующие риски. Однако возможный вред, который следует принимать во внимание, весьма отличается от вреда, рассматриваемого в ИСО 14971.

Отказ в системе производства и системе качества редко приводит к прямому вреду пациенту или пользователю медицинского изделия, производство или качество которого находится под управлением программного обеспечения. Вред в этом контексте почти всегда косвенный. Наносимый самому медицинскому изделию вред в конечном счете становится источником вреда для пациента или пользователя медицинского изделия. Это не означает, что косвенный вред менее серьезен. Фактически в некотором смысле тяжесть отказа производственных систем и системы качества можно считать более серьезной просто потому, что один их отказ может привести к сбоям во множестве изделий, что в конечном итоге может причинить вред многим пациентам до того, как он будет обнаружен. Отказ программного обеспечения на одном медицинском изделии может нанести вред только одному пациенту за раз.

Как прямой, так и косвенный многократный вред может быть причинен в результате отказа систем производства или системы качества. Следует отметить, что приведенный в нижеследующем списке вред не является взаимоисключающим. Каждый вред из списка может стать источником косвенного вреда для пациентов или пользователей медицинского изделия. Примеры включают, но не ограничиваются следующим:

- вред в отношении медицинского изделия:
  - производственное оборудование не отслеживает критически важные допуски;
  - система калибровки неправильно калибрует медицинское изделие для введения лекарственных средств;
  - вследствие отказа контроллера стерилизатора производятся нестерильные компоненты;
  - вследствие отказа системы финального тестирования не обнаруживаются скрытые дефекты изделия;
- вред в отношении производственного процесса:
  - сбой процесса, управляемого программным обеспечением, снижает производительность, поскольку операции приходится выполнять вручную;
  - сбой процессов, управляемых программным обеспечением, приводит к значительному увеличению количества брака;
- вред в отношении соответствия регулирующим требованиям:
  - система обработки жалоб искажает статистику отказов, оставляя незамеченными дефекты, обнаруженные при эксплуатации;

- система обслуживания или ремонта медицинских изделий не в состоянии выявить тенденции в отношении проблем, которые могут указывать на ранее необнаруженные дефекты;
- потеря целостности базы данных имплантированных медицинских изделий;
- потеря записей по управлению качеством, связанных с проверками безопасности изготовляемых сборочных элементов/частей;
- потеря данных о соответствии;
- потеря данных по валидации изделий;
- потеря возможности управления и регистрации конфигурации программного обеспечения в изготовленных изделиях;
- отказ системы обеспечения прослеживаемости (MRP) приводит к невозможности уведомления потенциальных пользователей этой системы об отзыве изделий в рамках обеспечения безопасности;
- вред в отношении производственного персонала или окружающей среды:
- оператор травмируется;
- выделяются токсичные химические вещества.

При анализе рисков должны учитываться все категории вреда, связанные с применением программного обеспечения для автоматизации производства и систем качества.

Определение риска включает определение тяжести возможного вреда и вероятности возникновения этого вреда.

Определение тяжести обычно выполняется с помощью классификации (см., например, ИСО 14971:2007, приложение D или G.4), которая связана с уровнем допустимости (см. В.5).

Можно столкнуться с трудностями в определении вероятности вреда, особенно если рассматривать вероятность отказов программного обеспечения, способствующих возникновению вреда. В этом случае следует учитывать, что программный сбой является лишь одним из факторов (другие факторы вне программного обеспечения могут быть частью в последовательности событий), приводящих к причинению вреда. Будет целесообразно предположить наихудший случай в отношении как неизвестной вероятности событий, так и вероятности причинения вреда.

Аналогичный подход используется в МЭК 62304:2006 (включая изменение 1:2015). В качестве основы для принятия решения об уровне строгости при управлении процессом предполагают наиболее неблагоприятную вероятность отказов программного обеспечения, но учитывают более низкую вероятность вреда, связанного с событиями за рамками программного обеспечения (см. также МЭК/ТО 80002-1).

В.5 Оценивание риска

После определения рисков необходимо выполнить их оценивание с целью определения того, являются ли они допустимыми или нет. Если риски недопустимы, изготовитель должен идентифицировать и реализовать меры по управлению риском, которые он разработал для снижения риска допустимого уровня.

Деятельность по установлению допустимого уровня риска, возможно, является самой сложной в рамках всего менеджмента риска и во многом зависит от уровня тяжести возможного вреда. Каждый изготовитель должен установить критерии: для определения и документирования допустимости риска; для идентификации всех рисков в формате, который позволит выполнить их оценивание в соответствии с этими критериями. В целом если допустимый риск снижен до уровня, который можно обосновать коллегам, высшему руководству или аудиторам, то, возможно, этот риск установлен на соответствующем уровне.

Рекомендации о пороговых значениях допустимости риска не входят в область применения настоящего стандарта. Тем не менее некоторые рекомендации по их установлению являются целесообразными:

- конкретность. Критерии допустимости, такие как «как можно ниже» или «настолько безопасно, как и любой другой продукт», являются бесполезными. Критерии должны представлять собой проверяемую спецификацию, чтобы можно было объективно установить выполнение критериев допустимости риска;
- в ситуациях, где трудно определить вероятность причинения вреда, критерии допустимости могут основываться только на тяжести;
- критерии допустимости могут относиться к предварительно выбранным и установленным программным средствам управления процессами (например, к выбору инструментов, перечисленных в таблицах А.1—А.5);
- идентифицируйте критерии допустимости на начальном этапе. Как только будет идентифицирован возможный риск причинения вреда, следует установить дальнейшую задачу или создать спецификацию на риск. Важно установить дальнейшую задачу в свете допустимости риска, прежде чем пытаться управлять рассматриваемым риском.

Восприятие допустимости риска часто смещается к более высоким уровням после предпринятых попыток управлять риском. Предварительное документирование критериев допустимости предотвращает смещение процесса оценивания риска;

- документирование обоснования по установлению допустимости рисков. Такая документация необходима для последующего поддержания процесса, а также для разъяснения принятых решений инспекторам, действующим от имени регулирующих органов.

В.6 Управление рисками

В.6.1 Недопустимый риск

Если в результате оценивания риск был сочтен недопустимым, изготовитель должен идентифицировать и внедрить меры по управлению риском с целью его снижения до допустимого уровня. Такие меры по управлению риском могут повлиять на программное обеспечение или другие части процесса.

В.6.2 Меры по управлению риском, не влияющие на программное обеспечение

Примерами мер по управлению риском, не влияющих на программное обеспечение, являются процедурные изменения, аппаратное резервирование, системы резервного копирования, системы мониторинга, верификация выходных данных (последующая верификация) или проверки поставщиков.

Доступ к встроенному программному обеспечению производственного процесса может быть затруднен, и его изготовитель может предоставлять недостаточно информации. Типичным примером является встроенное в производственное оборудование программное обеспечение, которое используется при изготовлении медицинского изделия. Валидация этого типа программного обеспечения на предмет его предусмотренного применения может быть затруднена, если программное обеспечение валидируется обособленно.

Мерой по управлению риском, которая особенно хорошо работает в таких ситуациях, является последующая верификация выходов программного обеспечения или выходов устройства, управляемого программным обеспечением. Другими словами, можно непосредственно установить пригодность программного обеспечения для его предусмотренного применения посредством мониторинга выходов процесса, управляемого программным обеспечением, на наличие любых потенциально вредных дефектов. Этот подход мог бы заменить вывод о пригодности программного обеспечения для его предусмотренного применения путем применения методологий управления жизненным циклом программного обеспечения в системе менеджмента качества. Такая методология применима только для процессов с небольшим количеством критических операций, которые можно проверить для каждой детали или для статистически установленной выборки. Инженер по валидации должен подробно изложить обоснование необходимости использования верификации выходов вместо непрерывной верификации, а также обосновать все допущения, связанные с используемой заменой. В этом случае последующим этапом является проверка сделанных допущений.

Верификация выходов должна быть задокументирована так же, как и любая другая мера по управлению риском. В частности, важно задокументировать этот процесс верификации как меру по управлению риском, чтобы впоследствии он не был устранен в рамках мероприятий по сокращению расходов на проект. Кроме того, результаты верификации выходов должны быть задокументированы, поскольку определение валидации требует «предоставления объективных свидетельств» валидации, и этот этап верификации заменяет большую часть валидации. По мере развития продукции предусмотренное применение процесса, управляемого программным обеспечением, также может развиваться. В качестве примера можно рассмотреть управляемое программным обеспечением производственное оборудование, которое первоначально выполняло одну критическую операцию на одном компоненте медицинского изделия. Позднее проект медицинского изделия был изменен таким образом, что на этом программируемом оборудовании стали выполняться две критические операции. Предусмотренное применение такого производственного оборудования изменилось (две критически важные для безопасности операции против одной), следовательно, должна быть изменена верификация выходов для проверки двух операций.

Верификация выходов может быть выполнена с помощью операций, выполняемых вручную, или других выполняемых человеком операций. Примерами могут быть визуальный осмотр неровных краев или механическое выравнивание, а также измерения вручную физических допусков или электрической целостности. Независимо от характера теста, если он является верификацией выхода процесса, управляемого программным обеспечением, а также если он используется в качестве меры по управлению риском для этого процесса, то такой верификационный тест должен быть задокументирован. Процедура тестирования для выполняющего этот тест человека должна быть детализированной и содержать четко определенные критерии принятия и отклонения результатов для каждого тестируемого параметра. Проводящие тестирование люди также должны предоставить документированные свидетельства того, что они выполнили процедуры по тестированию выходов процесса.

В.6.3 Меры по управлению риском, влияющие на программное обеспечение

Мерами по управлению риском, влияющими на программное обеспечение, являются либо изменение проекта, либо выбор средств управления процессом.

В контексте настоящего стандарта выбор средств управления процессом также относится к уровню строгости проведения валидации и предполагает выбор инструментов, определенных в таблицах А.1—А.5.

Предпочтительно, чтобы хорошо понятные меры по управлению риском, используемые вне программного обеспечения, например «верификация выходов», а также изменения в проекте программного обеспечения были реализованы раньше, чем средства управления процессом.

Тем не менее следует применить минимальный набор средств управления процессом, особенно для обеспечения уверенности в надлежащем внедрении изменений в проект программного обеспечения в качестве мер по управлению риском.

В.6.4 Верификация мер по управлению риском и оценивание остаточного риска

Выполнение мер по управлению риском должно быть верифицировано. Меры по управлению риском, выходящие за рамки управления процессом, должны быть верифицированы на результативность. В этом случае остаточный риск должен быть оценен по критериям допустимости.

Приложение С
(справочное)

Примеры

Настоящий стандарт применим к программному обеспечению, используемому для автоматизации элементов систем качества и производственных процессов, включая генерацию, измерение, оценку или менеджмент данных, которые предназначены для предоставления в регулирующие органы, систему качества, системы управления производством и обработки данных. Другие предполагаемые виды применения могут включать в себя прямой или косвенный сбор данных от устройств, задействованных в эксплуатации и управлении оборудованием, а также в обработке, протоколировании и хранении данных. Для этих различных видов деятельности программное обеспечение может варьироваться от программного обеспечения, содержащегося в программируемом логическом контроллере (ПЛК) или персональном компьютере (ПК), до программного обеспечения, содержащегося в информационной менеджмент-системе лаборатории (ЛИМС) с несколькими функциями. Ниже приведены некоторые примеры предусмотренного применения:

- программное обеспечение, которое принимает решение по сдаче/приемке продукции;
- программное обеспечение, используемое для ведения записей в системе качества;
- программное обеспечение для обработки и анализа данных, используемое для подачи продукции на оценку (регистрацию);
- программное обеспечение для обработки и анализа данных, используемое для отчетности в регулирующие органы;
- любые средства разработки программного обеспечения или компиляторы, используемые для программного обеспечения, задействованного в управляемом процессе;
- любой программный инструмент или вспомогательный программный инструмент, отвечающий за квалификацию и верификацию жизненно важного программного обеспечения;
- любое программное обеспечение, обеспечивающее прослеживаемость компонентов, продукции или пациентов в рамках системы качества;
- любое «программное обеспечение неизвестного происхождения» (т. е. неизвестного качества и надежности), используемое для вышеуказанных целей.

Примеры, приведенные в данном приложении, представляют собой попытку разработчиков настоящего стандарта предложить практические, реалистичные примеры программного обеспечения, с которыми может столкнуться изготовитель медицинских изделий. Рассмотрение примеров является лучшим способом показать подход на основе критического мышления и продемонстрировать различия в типах программного обеспечения, рисках программного обеспечения и предусмотренном применении.

Следует обратить внимание на следующие уточняющие условия:

- приведенные примеры содержат результаты критического мышления, выполненного разработчиками настоящего стандарта, и предусматривают приемлемый уровень масштаба усилий по валидации, которые обеспечивают уверенность в том, что программное обеспечение будет функционировать как предусмотрено. Пользователям настоящего стандарта настоятельно рекомендуется рассмотреть, какая деятельность и масштаб усилий целесообразны с инженерной точки зрения, а также определить требуемый уровень строгости на основе ключевых факторов для программного обеспечения, применяемого в процессах системы менеджмента качества медицинских изделий;
- всегда существует более чем один способ установить уверенность в адекватности масштаба усилий по валидации. Приведенные в настоящем стандарте примеры демонстрируют методический подход, основанный на современном понимании вопроса и опыте разработчиков настоящего стандарта;
- пользователям настоящего стандарта не следует рассматривать мнение его разработчиков как нечто официальное и предписывающее. Схожесть примеров по формату приведена исключительно в целях представления данных, и в них включены важные соображения для демонстрации использования критического мышления. Приведенные схемы действий не предназначены для использования в качестве шаблонов валидации, они не отражают необходимую глубину и детализацию, которая требуется в фактической документации по валидации;
- приведенные примеры предполагают, что обязательные процессы, указанные в разделе 6, в организации установлены и поддерживаются в рабочем состоянии. Несмотря на то что примеры не содержат развернутых ссылок на обязательные процессы, они должны быть внедрены для обеспечения того, что программное обеспечение и все связанные с ним аспекты, такие как документация и остальная инфраструктура, подлежат управлению изменениями;
- каждый пример начинается с четкого определения управляемого процесса. Таким образом, уже установлено, что процесс и программное обеспечение подпадают под область применения настоящего стандарта. Далее идут идентификация и обобщение деятельности на основе критического мышления;
- примеры предназначены для предоставления информации о решениях и факторах принятия решений, используемых в процессе критического мышления. Они не обязательно отражают детальные шаги по валидации рассматриваемого программного обеспечения;

- любые названия компаний, команд или отдельных лиц, использованные в примерах, являются вымышленными и включены только для облегчения обсуждения.

Используемые примеры в основном показывают приведение конкретной системы в валидированное состояние. Установление валидированного состояния системы имеет большое значение, однако его поддержание на этапе обслуживания системы также очень важно для обеспечения правильной работы программного обеспечения и связанных с ним процессов. Деятельность по обслуживанию требует использования критического мышления и применения того же управления, которые требуются для первоначальной деятельности по валидации.

Пример 1: Программируемый логический контроллер (ПЛК) для производственного оборудования

Вводная информация

Компанией Tubing Supply Company был заключен контракт с крупным изготовителем медицинских изделий на поставку ему комплектующих — полимерных трубок для систем внутривенного вливания. Компания получила спецификации на трубки, среди которых — требования к производству трубок запатентованной формы. Это специальное требование к форме трубок будет выполнено компанией Tubing Supply Company в рамках основного процесса производства сегментов трубок.

Процесс производства трубок специфической формы требует особого внимания поставщика, поскольку является уникальным процессом, для выполнения которого в настоящее время поставщик не имеет оборудования. Было принято решение, что для выполнения этой задачи будет разработано специальное оборудование с программируемым логическим контроллером (ПЛК). Действующие политики компании — изготовителя медицинских изделий требуют проведения валидации процесса применения данного оборудования, включая содержащийся в нем ПЛК, на предмет их предусмотренного применения.

Определение процесса

Поставляющая трубки компания и изготовитель медицинских изделий формируют команду с целью определения процесса, посредством которого будет производиться трубка специфической формы. Установленный командой процесс предусматривает использование определенных температуры и давления для создания необходимой формы полимерной трубки. Процесс состоит из следующих шагов:

a) получить материалы;

b) загрузить материалы в оборудование;

c) сформировать диаметр трубки до нужного с помощью давления и нагрева;

d) охладить трубку;

e) снять трубку с машины;

f) проверить диаметр трубки на соответствие заданным параметрам.

Анализ рисков процесса

Изготовитель медицинских изделий сообщил компании Tubing Supply Company, что в результате анализа риска процесса обнаружены следующие проблемы и связанные с ними опасности:

- отсутствие хорошего соединения с контейнером, содержащим инфузионный раствор, может привести к протеканию. Утечка жидкости не является опасностью, но медработник, осуществляющий уход за пациентом, может поскользнуться. Протекание может также привести к задержке лечения;

- внешние дефекты могут повлиять на приемку заказчиком и привести к задержке лечения;

- существует вероятность получения ожога оператором при термической обработке трубки.

До снижения существует умеренный уровень риска, связанный с отказом продукции из-за опасности падения медработника на скользкой поверхности, задержки лечения и ожогов оператора.

В настоящее время выполняются следующие меры по управлению риском процесса:

- предшествующие операции, такие как входной контроль и уборка производственной линии, проводимые в целях обеспечения пригодности линии для использования;

- последующие проверочные мероприятия, в том числе испытание на герметичность, контроль в рамках процесса и испытания фитингов, которые минимизируют последствия неправильной работы оборудования;

- защитные экраны, автономный датчик температуры и распылитель охлаждающей жидкости, установленные для предотвращения травм оператора.

Используя эту информацию, поставщик и изготовитель медицинского изделия формулируют вывод о том, что в результате рассматриваемого производственного процесса существует низкий остаточный риск несоответствия трубок заданным параметрам.

Определение цели и назначения программного обеспечения

Компания Tubing Supply Company знает, что для валидации применения программного обеспечения сначала нужно определить его предусмотренное применение. Чтобы прийти к общему пониманию о предназначении оборудования, члены команды обсуждают ряд вопросов, способствующих разработке определения цели и назначения системы, которое будет кратким, но приемлемым для использования. Они формулируют цель и назначение системы следующим образом:

- управляемое программным обеспечением оборудование предназначено для автоматизации процесса на этапах 2—6. Система предназначена для использования в подразделении B, производственная линия 3, для создания PN 001. Система будет автоматизировать загрузку, формирование, извлечение и измерение трубок для систем внутривенного вливания IV жидких форм лекарственных средств.

Планирование валидации

Первый шаг в планировании валидации включает установление уровней строгости к документации и анализу результатов. Поскольку остаточный риск процесса определен как низкий, был выбран следующий подход:

- строгость документации:

- документация в этом проекте будет иметь средний уровень строгости, означающий возможность объединения некоторых результатов и отсутствие необходимости перевода проектов документов в подробные проектные спецификации до имплементации;

- уровень анализа:

- анализ и одобрение будут проводиться лицами, ответственными за разработку и внедрение процесса (представитель компании Tubing Supply Company), а также независимым специалистом по качеству (представитель компании — изготовителя медицинских изделий);

- код ПЛК и все спецификации будут являться объектами менеджмента конфигурации, например путем размещения в системе управления документами или системе управления конфигурацией;

- определение системы:

- будут созданы требования к процессу, включающие спецификацию требований к системе, детализирующие функциональные возможности оборудования, в том числе ожидаемые входные и выходные данные оборудования (например, элементы управления проектированием для всей функциональной части оборудования);

- команда создаст руководство по эксплуатации для использования системы оператором. Кроме того, будут созданы требования к программному обеспечению, включающие в себя логический функциональный поток, достаточные для нужд проектирования программного обеспечения.

Установление уверенности и управления программным обеспечением

Компания Tubing Supply Company и изготовитель медицинских изделий ранее не использовали этот программный пакет ПЛК. Компания Tubing Supply Company не располагает информацией, которая могла бы помочь укрепить уверенность в способности программного обеспечения работать в соответствии с требованиями. Тем не менее программирование ПЛК будет находиться под управлением посредством анализа требований, управления конфигурацией и проведения испытаний функциональности системы с помощью протоколов тестирования.

Определение границ взаимодействия программного обеспечения с другими системами

ПЛК содержит только одно программное обеспечение в части оборудования. Это программное обеспечение не связано с какой-либо другой системой.

Анализ риска программного обеспечения

Программное обеспечение может отказать и пропустить далее по производственной линии трубку неправильной формы, что приведет к протеканию жидкости и в дальнейшем к тому, что медработник может поскользнуться. Программное обеспечение также может работать неправильно, что может привести к чрезмерному нагреву и в дальнейшем к ожогам оператора. Само по себе программное обеспечение не создает никаких новых рисков для продукции, которые еще не были зафиксированы при анализе риска процесса. Таким образом, команда устанавливает, что существующие меры по управлению риском и последующие процессы должны оставаться без изменений и являются достаточными для уменьшения рисков, связанных с отказом программного обеспечения.

Завершение планирования валидации

Теперь, когда члены команды знают больше о программном обеспечении и его применении, они должны завершить планирование валидации следующим образом:

- инструменты имплементации:

- программируемые встроенные параметры работы оборудования включают время, температуру и давление. Требуемые настройки и диапазоны для этих параметров в оборудовании отражены в требованиях к программному обеспечению. Поэтому спецификация требований к программному обеспечению является достаточной для нужд и целей проектирования без необходимости установления дополнительной деятельности по проектированию или документированию;

- команда установит матрицу прослеживаемости между требованиями к программному обеспечению и связанными с ними испытаниями, а также проведет анализ устанавливаемой прослеживаемости для обеспечения ее целостности;

- инструменты тестирования:

- тестирование системы программного обеспечения будет основано на требованиях к программному обеспечению и процедурах, изложенных в руководстве оператора;

- регрессионное тестирование будет проводиться при необходимости;

- инструменты развертывания:

- системные операторы и инженеры проведут анализ рабочих инструкций на предмет их четкости и удобства использования;

- использование оборудования потребует аттестации оператора;

- после завершения плана валидации и выполнения установленной в нем деятельности у команды сложилось единое мнение, что система будет последовательно обеспечивать желаемые и предопределенные результаты.

Вопросы обслуживания

Если планируются изменения в какой-либо части этого процесса или возникают изменения в предусмотренном применении программного обеспечения, следует провести анализ и определить, окажут ли эти изменения влияние на какие-либо существующие меры по уменьшению рисков или приведут к появлению новых рисков. Этот анализ включает обзор программных рисков, связанных с оборудованием для производства трубок.

Использование набора инструментов

Из набора инструментов были использованы следующие:

- стадия разработки — определения:
  - определение требований к процессу;
  - анализ риска отказа процесса;
  - предусмотренное применение;
  - планирование валидации;
  - определение требований к программному обеспечению;
  - идентификация мер по управлению рисками в производственном процессе;
- стадия разработки — имплементации:
  - анализ отказов программного обеспечения;
  - анализ прослеживаемости;
- стадия разработки — испытания (тестирования):
  - системное тестирование программного обеспечения;
  - регрессионное тестирование;
- стадия разработки — развертывания:
  - анализ используемых пользователем процедур;
  - аттестация оператора.

Пример 2: Система автоматической сварки

Дэйв является частью команды по валидации всех систем на новой производственной линии. Его работа заключается в валидации процесса сварки составных частей конструкции изделия (трубки с корпусом капельницы). В этом проекте он является менеджером проекта.

Описание процесса

Команда Дэйва проводит много времени в обсуждении: кто разрабатывает и какие участки новой производственной линии валидирует. Дэйв получает детали для сварки, которые уже промаркированы, а все материалы проверены и сертифицированы. Детали испытаны в валидированных системах выше по производственной линии, чем участок Дэйва.

Для настройки сварочной установки необходимо выполнить четыре шага:

- включение установки;
- подтверждение наличия штрих-кода в детали для сварки;
- извлечение программы для операции с деталями из системы управления производством;
- подтверждение правильной версии программы относительно Основных записей изделия (DMR).

Сам процесс термической сварки трубки с корпусом капельницы состоит из 10 этапов:

a) открыть дверь;
b) загрузить детали;
c) закрыть дверь;
d) запустить программу;
e) разместить индексы системы обработки данных визуального контроля в начальной точке;
f) включить лазер;
g) обеспечить перемещение сварных соединений детали системой управления движением;
h) выключить лазер;
i) открыть дверь;
j) убрать деталь.

После завершения этого процесса детали перемещаются в системы, которые не входят в сферу ответственности Дэйва. Он знает, что последующие действия включают разрушающий контроль прочности сварного соединения, проверку геометрических размеров и проверку на герметичность.

Определение предусмотренного применения

С целью определения предусмотренного применения программного обеспечения Дэйв собирает информацию. Он знает, что в процессе важны как точность визуализации, движения, силы и скорости, так и защита оператора и обеспечение прочности термической сварки трубки с корпусом капельницы.

Сначала Дэйв определяет предусмотренное применение, излагая цель и назначение программного обеспечения следующим образом:

Программное обеспечение предназначено для выполнения термической сварки частей изделия и защиты оператора установки от прямого контакта с работающим лазером. Указанная деятельность включает в себя шаги e) — h) в описании процесса.

Анализ риска

Дэйв хотел бы исключить из процесса человеческий фактор. Он знает, что управление лазером, сервомеханизмы и автоматическая следящая система визуализации (машинное зрение) являются ключевыми компонентами этого процесса. Программное обеспечение начинает работу с проверки закрытия двери. В целях безопасности программное обеспечение не запустит процесс без сигнала о закрытии двери. Работа программного обеспечения завершается подтверждением выключения лазера и разрешением на открывание двери. Аварийная остановка или неожиданное открытие двери приводит к отключению лазера. Дэйв использует информацию о процессе и деятельности по менеджменту риска, полученную им из документации по проектированию этого процесса, частью которого является термическая сварка. Дэйв ссылается на проведенный анализ видов и последствий отказов (FMEA) и сосредотачивается на трех областях: критические параметры скрепления деталей, герметичность шва и пользовательские интерфейсы. Дэйв идентифицирует множество опасностей, связанных с рассматриваемым процессом. Во-первых, оператор может получить ожог при контакте с лазером. Процесс термической сварки может работать недостаточно хорошо, что приведет к несоответствию продукции на выходе процесса (нарушение герметичности) и может причинить вред конечному пользователю. Дэйв определяет риск этого процесса как высокий.

Планирование валидации

Для этого проекта Дэйв рассматривает инструменты стадии разработки — определения из набора инструментов и решает, что ему нужно определить требования к программному обеспечению и создать документ по техническому обслуживанию. Требования к программному обеспечению должны включать параметры конфигурации для инструментов, а также настройки времени работы и мощности лазера. Ему также необходимо определить программное обеспечение к аппаратным интерфейсам. В частности, Дэйв включает требования к точности системы

визуализации, диапазонам времени и мощности лазера, требования к точности управления движением и надежности датчика двери, в том числе интерфейс с аппаратным закрыванием двери, если лазер активирован.

Дэйв также устанавливает, что ему необходимо провести формальный анализ требований к программному обеспечению при участии инженера по автоматизации, инженера по производству и инженера по качеству.

Программное обеспечение для этой системы будет закуплено, но Дэйв знает, что его компании нужно будет вносить пользовательские изменения. Ему нужно добавить интерфейс к заводской системе управления производственными процессами (MES).

Меры по управлению риском

Далее Дэйв сосредотачивается на рисках. Он считает, что тяжесть вреда из-за толщины сварного шва и других критических параметров является низкой. Он уверен, что проверка герметичности на выходе процесса и периодический разрушающий контроль прочности соединения являются достаточными. Аналогично, проверка на герметичность подтвердит, что качество термической сварки соответствует критериям приемки. Остается риск в области пользовательских интерфейсов, и в частности риск запуска лазера программным обеспечением при открытой двери. Дэйву известно, что существуют программные проверки плотности закрытия двери, но, учитывая, что при программном отказе тяжесть риска высока, он добавляет избыточную аппаратную блокировку, предотвращающую активацию лазера при открытой двери.

Задачи валидации

Затем Дэйв переходит к задачам валидации. Выбранный им вендор-поставщик предоставил обширные программные инструменты. Поэтому созданные ранее спецификация и анализ требований к программному обеспечению достаточны для реализации проекта без использования дополнительных инструментов проектирования, разработки и конфигурирования из набора инструментов.

Еще одной задачей является предстоящая деятельность по планированию проведения испытаний, которую Дэйв выбрал из раздела «Разработка — испытания (тестирование)» набора инструментов. Планы испытаний должны включать подробную информацию о программных средах и ожидаемые результаты тестирования. Планы испытаний должны быть рассмотрены и утверждены инженером по автоматизации, инженером-технологом и инженером по качеству, а также Дэйвом. Отчет о тестировании будет включать представление фактических результатов тестирования, их сравнение с ожидаемыми результатами, индикацию принятия/отклонения результатов, идентификацию теста и документацию по решению проблем и регрессионному тестированию по любым отказам. В отношении составленного отчета Дэйв считает нужным получить дополнительное одобрение инженера по автоматизации, инженера по производству, инженера по качеству и спонсора проекта (Дэйв — менеджер проекта).

Развертывание

Для развертывания системы термической сварки Дэйв рассматривает инструменты развертывания из набора инструментов и решает, что необходима процедура для оператора на производстве, которая должна быть рассмотрена инженером по автоматизации, инженером по производству и инженером по качеству. Чтобы обеспечить навыки управления сварочным оборудованием у операторов, Дэйв создает процедуру обучения и аттестации оператора, которая включает в себя сдачу теста. Он знает, что система управления производственными процессами (MES) не допустит к работе неаттестованного оператора. Дэйв приходит к выводу, что риск травмирования оператора был успешно уменьшен.

Обслуживание

Дэйв знает, что в его фирме используется инструмент проверки конфигурации. Следовательно, он не проводит конкретного планирования технического обслуживания во время этой валидации.

Пример 3: Система управления процессом автоматической сварки

Таблицы С.1—С.14 в этом примере демонстрируют этапы процесса, показанные на рисунке 2.

Таблица С.1 — Пример 3 — Требования к процессу

Процесс	Итеративный анализ рисков	Планирование валидации и отчеты	Система ПО
Разработка	Определение	Требования к процессу (см. 5.3.2.2) Компания Device Corporation является изготовителем медицинских изделий класса С (см. GHTF/SG1/N77:2012). Компания решила внедрить автоматизированную систему управления процессом сварки. Для обеспечения надлежащей сварки деталей конструкции изделия Device Corporation будет использовать метод принятия решения по выпуску продукции на основе данных о значениях параметров, полученных в ходе технологического процесса (параметрический выпуск). Также Device Corporation решила использовать информацию из этого процесса для поддержания записей истории изделия (DHR). Компания Device Corporation назначила нового менеджера проекта для валидации системы управления процессом автоматической сварки. Менеджер проекта знает, что эта система подпадает под требования ИСО 13485 по валидации применения программного обеспечения, и поэтому считает валидацию рассматриваемой системы необходимой. Чтобы лучше понимать требования и риски, связанные с валидацией системы сварки, менеджер проекта определяет процесс: a) Оператор вводит номер партии в систему для загрузки первой части партии. b) Оператор вставляет сборочные компоненты в крепление машины. c) Оператор нажимает кнопку запуска цикла. Крепление перемещается в сопряженное положение механически с помощью гидравлики. d) Цикл сварки начинается вместе с фиксированной скоростью вращения закрепленных компонентов. e) Инфракрасный термометр контролирует температуру материала во время процесса сварки. Значения температуры записываются в файл вместе с номером партии и порядковым номером для каждой сваренной детали. f) Машина открывает крепление в конце цикла. g) Оператор вынимает скрепленную сварным швом деталь и помещает ее в нужное место лотка для партии в соответствии с порядковым номером. h) Оператор повторяет шаги с b) по g), пока лоток не будет заполнен. i) Оператор нажимает кнопку «конец партии». j) Интерфейс оператора отображает порядковые номера деталей, температура сварного шва которых выходит за пределы параметров технологического процесса. k) Оператор удаляет соответствующие номера деталей из лотка. l) Оператор распечатывает список отклоненных деталей и отправляет лоток с партией и отчет на следующую технологическую операцию. m) Оператор начинает новую партию, повторив шаг a). Менеджер проекта также понимает, что ключевыми функциями автоматизации являются следующие: - сохранение номера партии; - сохранение информации о температуре сварного шва по порядковому номеру детали; - отображение порядковых номеров деталей, по которым превышены пределы температуры процесса при сварке; - печать отчета об отклонении партии		

Таблица С.2 — Пример 3 — Анализ рисков отказа процесса

		Процесс	Итеративный анализ рисков	Планирование валидации и отчеты	Система ПО
Разработка	Определение	Анализ рисков отказа процесса (см. 5.3.2.3) Затем менеджер проекта анализирует, что может пойти не так в существующем процессе. Он понимает, что в случае сбоя процесса неправильно скрепленные детали конструкции могут подвергать пациентов риску использования нестерильных изделий. Случайный выпуск некачественной продукции может произойти из-за ошибки в системе управления сварочным процессом или из-за ошибки оператора. Далее менеджер проекта рассматривает, какие меры по управлению риском уже применяются для его снижения. Он узнает, что в Процессной группе есть процедура по проверке правильности отбраковки скрепленных деталей оператором сварки на следующем этапе процесса. Кроме того, менеджер проекта узнает, что сварочная система является коммерческой системой (Off The Shelf)			

Таблица С.3 — Пример 3 — Цель и назначение программного обеспечения

		Процесс	Итеративный анализ рисков	Планирование валидации и отчеты	Система ПО
Разработка	Определение	Цель и назначение программного обеспечения (см. 5.3.2.5.2) Обладая базовым пониманием процесса, менеджер проекта готов письменно сформулировать цель и назначение для системы управления процессом сварки: - Приложение управления процессом сварки автоматически принимает решения в отношении качества скрепления сварных деталей конструкции. На основании этих решений оператор сварки вручную отбраковывает несоответствующую продукцию. Менеджер проекта анализирует цель и предназначение на предмет надлежащего охвата границ программного обеспечения в рамках процесса и решает пересмотреть ранее составленную им формулировку: - Приложение управления процессом сварки автоматически принимает решения в отношении приемки или отклонения качества скрепления сварных деталей конструкции. На основании этих решений оператор сварки затем вручную отбраковывает несоответствующую по параметрам продукцию. Сварочная станция является единственной контрольной точкой во всем производственном процессе, которая обеспечивает целостность и герметичность соединений конструкции изделия. Затем менеджер проекта рассматривает, какие другие имеющиеся системы будут взаимодействовать со сварочной системой. Он устанавливает, что программное обеспечение представляет собой отдельное приложение, работающее на персональном компьютере, подключенном к инфракрасному температурному датчику, интерфейсу оператора, принтеру и входу/выходу ПЛК установки. Система сварки не подключается к сети			

Таблица С.4 — Пример 3 — Планирование валидации

		Процесс	Итеративный анализ рисков	Планирование валидации и отчеты	Система ПО
Разработка	Определение	Планирование валидации (см. 5.3.2.4) Теперь, когда менеджер проекта понимает процесс и уже установил предусмотренное применение новой системы, он готов разработать первоначальный план валидации. Ранее менеджер проекта установил, что в процессе сварки существует высокий остаточный риск, поскольку процесс должен быть реализован как неверифицируемый. Исходя из этого, он определяет, что необходим тщательный анализ уровня предполагаемых усилий по валидации. Менеджер проекта решает, что ключевые роли по официальному одобрению должны выполнять отделы технологического проектирования и инжиниринга качества, а также инструктор по обучению операционным процессам. Кроме того, менеджер по приемке готовой продукции должен утвердить требования. Менеджер проекта решает начать разработку своего плана валидации, т. к. система качества требует, чтобы для систем с высоким риском план валидации был утвержден до того, как будут одобрены любые другие отчетные документы по валидации или проекту			

Таблица С.5 — Пример 3 — Требования к использованию программного обеспечения и требования к программному обеспечению

		Процесс	Итеративный анализ рисков	Планирование валидации и отчеты	Система ПО
Разработка	Определение	Менеджер проекта считает, что необходимо обеспечить высокий уровень детализации или соблюдения формальностей в процессе валидации. Он знает, что необходимо определить подробные требования к процессу и программному обеспечению. Он письменно формулирует требования к программному обеспечению. Менеджер проекта решает, что программное обеспечение должно включать избыточность в процессы верификации температуры и принятия решения об отклонении некачественных изделий. Также он устанавливает дополнительное требование, чтобы система имела возможность перепечатывать отчет об отклонении в любое время до начала операций по очистке производственной линии. Поскольку эта система функционирует на основе данных о значениях параметров, менеджер проекта также включает требования к ее собственной безопасности наряду с подробным списком значений параметров, которые могут быть изменены в зависимости от уровня доступа к системе			

Таблица С.6 — Пример 3 — Анализ рисков отказа программного обеспечения

		Процесс	Итеративный анализ рисков	Планирование валидации и отчеты	Система ПО
Разработка	Имплементация, тестирование, развертывание	Анализ рисков отказа программного обеспечения (см. 5.3.3.2) Теперь менеджеру проекта необходимо решить, какой подход следует использовать для установления полной уверенности в сварочной системе. Менеджер проекта отмечает, что для исполнения проекта требуется готовый коммерческий продукт (COTS), который обычно используется в промышленности. Он выясняет, что прошлые проблемы с этим продуктом были быстро выявлены его изготовителем и доведены до сведения пользователей. Несмотря на то что менеджер проекта уже установил, что процесс сварки имеет высокий риск, он хочет еще провести формальный анализ риска отказа программного обеспечения. Чтобы подтвердить свои предположения, менеджер проекта анализирует вопросы из базы данных рисков компании. a) Существует ли потенциальный риск для безопасности продукции в случае неправильного функционирования программного обеспечения? *Да* 1) Как? *Система не отклоняет плохо скрепленную деталь, основываясь на температурных пределах по умолчанию. Ограничения сбрасываются к настройкам по умолчанию после перебоев энергоснабжения.* 2) Что нужно сделать для управления этим риском? *Требовать от оператора проверки значений пределов в начале и в конце каждой партии.* b) Существует ли потенциальный риск для качества продукции (кроме риска для безопасности), если пользователь допустил ошибку? *Да* 1) Как? В ручном режиме сварочный лазер может сработать, если оба датчика детали сработают в течение 3 с. 2) Что нужно сделать для управления этим риском? Изменить настройки по умолчанию, чтобы лазер работал только в автоматическом режиме			

Таблица С.7 — Пример 3 — Планирование валидации

		Процесс	Итеративный анализ рисков	Планирование валидации и отчеты	Система ПО
Разработка	Имплементация, тестирование, развертывание	Планирование валидации (см. 5.3.3.3) Понимая требования к программному обеспечению, менеджер проекта обладает достаточной информацией для завершения планирования валидации. Менеджер проекта провел анализ риска и определился с подходом к исполнению программного обеспечения. На этом этапе он делает шаг назад и задает следующий вопрос в свете всего, что он узнал об этой системе: «Какие действия по валидации действительно позволят обрести уверенность в том, что сварочная система соответствует ее предусмотренному применению?» Менеджер проекта обдумывает разработку системы третьей стороной и обеспокоен тем, правильно ли разработчик будет толковать требования для настройки отчета. Поскольку система будет зависеть от данных из различных областей, он предусматривает дополнительные действия по верификации в анализе кода, чтобы подтвердить правильность работы разработчика			

Таблица С.8 — Пример 3 — Имплементация программного обеспечения

		Процесс	Итеративный анализ рисков	Планирование валидации и отчеты	Система ПО
Разработка	Имплементация, тестирование, развертывание	Имплементация программного обеспечения (дизайн, разработка, сборка и тестирование) (см. 5.3.3.4) Решение о том, что программное обеспечение будет закуплено, а не разработано собственными силами компании, было принято на основе доступности готового коммерческого продукта (COTS). Однако менеджеру проекта необходимо доказать отделу качества Device Corporation, что программное обеспечение для управления сваркой было разработано в рамках корректно установленного жизненного цикла разработки (SDLC), поскольку риск предусмотренного применения был определен как высокий. После обсуждения этого вопроса с поставщиком готового коммерческого продукта менеджер проекта узнает, что процессы жизненного цикла разработки программного обеспечения у поставщиков недавно были оценены независимой аудиторской фирмой. Поэтому он связывается с независимой аудиторской фирмой и приобретает копию отчета об аудите поставщика готового коммерческого продукта. Конечным результатом является то, что отдел качества убежден, что поставщик готового коммерческого продукта разработал программное обеспечение в соответствии с корректной моделью жизненного цикла			

Таблица С.9 — Пример 3 — Отчет по валидации

		Процесс	Итеративный анализ рисков	Планирование валидации и отчеты	Система ПО
Разработка	Имплементация, тестирование, развертывание	Отчет по валидации (см. 5.3.3.5) Менеджер проекта завершает отчет по валидации и получает на него официальное одобрение			

Таблица С.10 — Пример 3 — Выпуск программного обеспечения

		Процесс	Итеративный анализ рисков	Планирование валидации и отчеты	Система ПО
Разработка	Имплементация, тестирование, развертывание	Выпуск программного обеспечения (см. 5.3.3.6) Менеджер проекта верифицирует соответствие программного обеспечения, переданного в формальную систему управления конфигурацией, программному обеспечению, указанному в отчете по валидации			

Таблица С.11 — Пример 3 — Анализ изменений

		Процесс	Итеративный анализ рисков	Планирование валидации и отчеты	Система ПО
Обслуживание		Анализ изменений (см. 5.4) Менеджер проекта верифицирует наличие формализованного процесса управления изменениями, который управляет любыми последующими изменениями в валидированной сварочной системе, что соответствует плану валидации, имеющемуся у компании			

Таблица С.12 — Пример 3 — Планирование валидации на стадии обслуживания

		Процесс	Итеративный анализ рисков	Планирование валидации и отчеты	Система ПО
Обслуживание		Планирование обслуживания (см. 5.4.2) Менеджер проекта заранее обдумывает, какая деятельность будет уместна для обеспечения уверенности в том, что система продолжает соответствовать своему предусмотренному применению. Учитывая высокий риск системы, он решает, что следует проводить ежеквартальную калибровку и подтверждение точности и прецизионности фактического значения измеренной температуры и значений температуры, указанных в отчете по партии. Менеджер проекта включает этот раздел в план валидации для документирования заключения и создает запрос на разработку и внедрение процедуры калибровки и подтверждения, чтобы обеспечить ее ежеквартальное проведение			

Таблица С.13 — Пример 3 — Обслуживание программного обеспечения

		Процесс	Итеративный анализ рисков	Планирование валидации и отчеты	Система ПО
Обслуживание		Обслуживание программного обеспечения (см. 5.4.6) Менеджер проекта в соответствии с планом валидации верифицирует наличие в компании процесса проведения периодического анализа, обеспечивающего отсутствие расхождений между сварочной системой и процессом от их предусмотренного применения			

Таблица С.14 — Пример 3 — Снятие с эксплуатации

		Процесс	Итеративный анализ рисков	Планирование валидации и отчеты	Система ПО
Вывод из эксплуатации		Снятие с эксплуатации (см. 5.5) Менеджер проекта в соответствии с планом валидации верифицирует наличие у компании формализованного процесса вывода программного обеспечения из эксплуатации, который регламентирует снятие сварочной системы с эксплуатации			

Выбор инструментов

Инструменты проектирования, разработки и конфигурации:
- определение требований к процессу;
- формализованный анализ требований к программному обеспечению;
- идентификация мер по управлению риском в производственном и бизнес-процессах;
- анализ на стадиях разработки процесса;
- матрица прослеживаемости (характерна для спецификации требований).

Инструменты испытаний (тестирования):
- планирование проведения испытаний;
- системное тестирование программного обеспечения;
- управление конфигурацией программного обеспечения.

Инструменты развертывания:
- анализ используемых пользователем процедур;
- внутреннее обучение по применению;
- установочная квалификация (IQ);
- валидация процесса.

Пример 4: C/C++ языковый компилятор (языка программирования)

Вводная информация

Компании, выпускающей медицинские изделия класса С, необходимо валидировать готовое программное обеспечение — языковой компилятор C/C++ для встроенной системы. Было установлено, что этот компилятор подпадает под регулирующие требования и должен быть валидирован, поскольку он генерирует программное обеспечение для медицинского изделия (исходный код программного обеспечения и исполняемое программное обеспечение), которое помещается в проектные записи медицинского изделия.

Описание процессов системы качества

Для этого примера подходящими являются два процесса системы качества. Первый — это общий процесс системы качества по разработке медицинского изделия, содержащего программное обеспечение класса С (см. рисунок С.1). Вторым является процесс разработки исполняемых программных модулей, которые реализуют программный проект и отвечают всем программным требованиям. Эти программные модули включают в себя языковой компилятор OTSS (off-the-shelf-system/система, готовая к использованию) C/C++ (см. «Имплементация ПО (программного обеспечения)» на рисунке С.1).

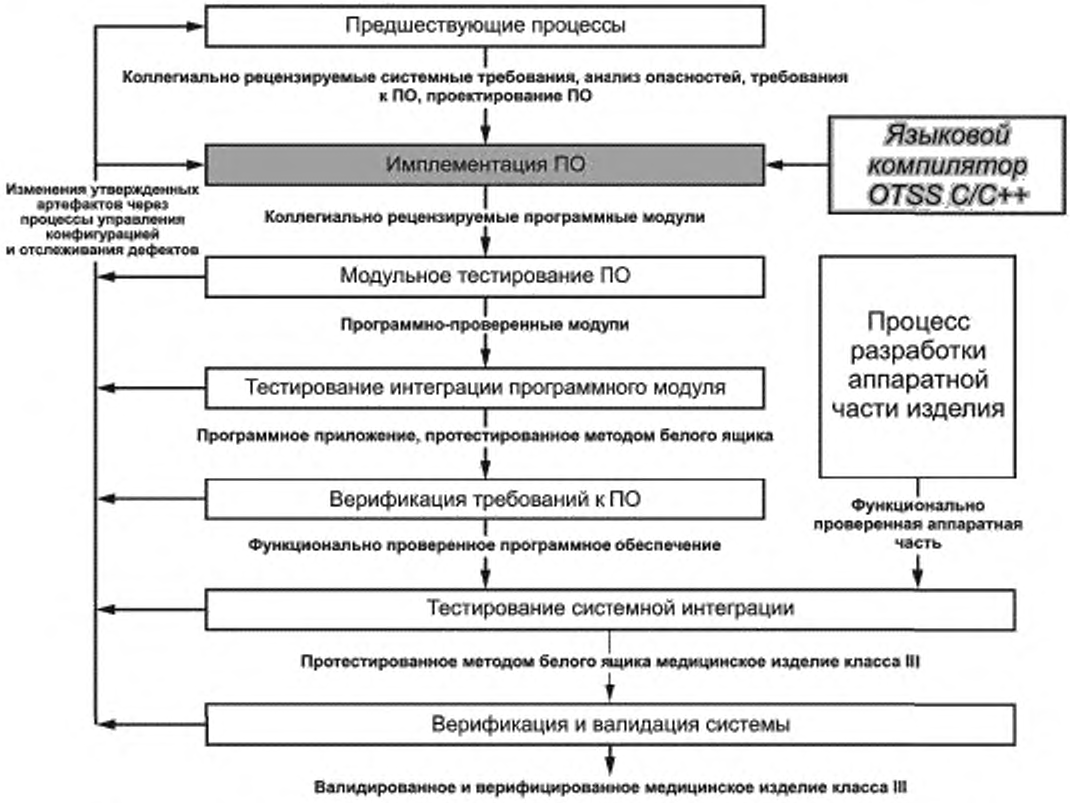


Рисунок С.1 — Имплементация программного обеспечения класса С медицинского изделия

Предшествующие процессы

Процессу имплементации программного обеспечения предшествуют процессы разработки документации системного уровня (например, требований, проекта, анализа опасностей), которая характеризует разрабатываемое медицинское изделие. Часть системы, реализованная в программном обеспечении, затем характеризуется посредством процессов разработки требований к программному обеспечению, разработки проекта программного обеспечения и других программных документов или планов. Параллельно с разработкой программного обеспечения реализуются дополнительные процессы по разработке аппаратной части медицинского изделия.

Процесс имплементации программного обеспечения

Используемый формальный язык программного обеспечения — язык программирования C/C++. Языковой компилятор C/C++ используется для компиляции программных операторов высокого уровня в исполняемый машинный код. Выходными данными процесса имплементации программного обеспечения являются базовые программные модули, которые коллегиально рецензируются на предмет их полноты и правильности. Рецензирование программного модуля проводится в виде экспертной оценки, для которой программный модуль должен быть скомпилирован без ошибок на самом высоком уровне компилятора. Любые предупреждающие сообщения компилятора должны быть объяснены во время экспертной оценки.

Последующие процессы тестирования

Программные модули испытываются или верифицируются в нескольких процессах тестирования:

- модульное тестирование программного обеспечения. Отдельные программные модули проверяются на логическую корректность и предельные условия для каждого модуля. Тестирование может проводиться в системе разработки или целевой системе (аппаратном обеспечении медицинского изделия). Простые программные модули могут не тестироваться, если установлено, что проводимая экспертная оценка кода достаточна для обнаружения логических ошибок модуля;
- тестирование интеграции программных модулей. Программные модули интегрируются и тестируются для обеспечения того, что проект программного обеспечения правильно реализован и что предельные условия верифицированы согласно проекту. Тестирование проводится в целевой системе;
- верификация требований к программному обеспечению. Завершенное программное приложение верифицируется на соответствие всем требованиям к программному обеспечению. Верификация выполняется в целевой системе;
- интеграционное тестирование системы. Программное и аппаратное обеспечение в медицинском изделии тестируется для обеспечения правильной имплементации проекта системы и проверки граничных условий в отношении проекта системы;
- верификация и валидация системы. Медицинское изделие верифицируется на уровне системных требований, а также валидируется на предмет предусмотренного применения.

Анализ риска отказа процесса

Проект осуществлялся в соответствии с процедурой компании по оценке рисков процесса. Общий процесс системы качества по имплементации программного обеспечения для медицинских изделий класса С (который включает в себя все процессы, описанные на рисунке С.1) по своей природе сопряжен с высоким риском, поскольку он генерирует программное обеспечение, которое функционирует в медицинском изделии класса С.

Языковой компилятор C/C++ как часть процесса имплементации программного обеспечения оценивается как малоопасный на основе двух факторов:

- компилятор прямо не причиняет серьезной травмы или смерти пациенту, оператору или людям, находящимся рядом;
- на выходе (исходный код программного обеспечения и исполняемое программное обеспечение) этого инструмента выполняется последующая верификация (например, тестирование программного модуля, тестирование интеграции программных модулей, верификация требований к программному обеспечению, тестирование системной интеграции, верификация и валидация системы).

Определение предусмотренного применения

Цель и назначение языкового компилятора *OTSS C/C++* в описанном выше процессе имплементации программного обеспечения состоит в том, чтобы создать исходный код встроенной системы и выполнить процесс компиляции для генерации исполняемого программного обеспечения для медицинского изделия класса С.

Требования к использованию программного обеспечения

a) Инструмент должен выполнять компиляцию в машинный код целевой платформы C и C++ для работы на процессоре компьютера с сокращенным набором команд (RISC) с использованием операционной системы выбранного поставщика.

b) Компилятор должен иметь отладчик исходного кода.

c) Компилятор должен удовлетворять требованиям стандарта C и C++ Американского национального института стандартов (ANSI).

d) Компилятор должен интегрироваться с различными средами разработки, которые соответствуют отраслевым стандартам.

e) Поставщик должен публиковать список известных ошибок, доступный для поиска. Список следует использовать в качестве справочного материала для консультации по мере необходимости.

f) Поставщик должен иметь большую базу пользователей в регулируемой отрасли.

Анализ риска отказа программного обеспечения

Анализ риска языкового компилятора *OTSS C/C++* показывает, что в случае ошибки могут произойти следующие события.

- Риск 1. Поставщик не может обеспечить соответствующие бизнес-процессы, методы разработки и возможности поддержки.
    - Снижение риска 1. См. раздел «Процесс выбора поставщика» ниже.
- Риск 2. Компилятор выдает некорректные исполняемые операторы.
    - Снижение риска 2. См. раздел «План валидации» ниже.
- Риск 3. Пользователь, который не применяет самый строгий уровень проверки ошибок, неправильно использует компилятор.
    - Снижение риска 3. Улучшить обучение, процедуры и рабочие инструкции.

Процесс выбора поставщика

Проект выполнялся в соответствии с процедурой системы качества компании по выбору и одобрению поставщиков, и это зафиксировано в проектных записях. Процедура включала в себя оценку на месте с рассмотрением политик, процедур, задач и деятельности поставщика по разработке жизненного цикла программных систем. Возможности предлагаемого поставщиком языкового компилятора *OTSS C/C++* были верифицированы на соответствие требованиям к использованию программного обеспечения, определенным выше.

План валидации

Для языкового компилятора *OTSS C/C++* был выбран следующий подход к валидации. Процесс выбора поставщика установил, что поставщик соответствует всем задокументированным требованиям к использованию программного обеспечения. Компилятор достаточное время работал в режиме выполнения у поставщика и будет значительное время работать в режиме выполнения в процессе отладки и тестирования, осуществляемых в проекте. В последующих процессах выходные данные компилятора подвергаются следующим видам динамического тестирования:

- модульное тестирование программного обеспечения;
- тестирование интеграции программных модулей;
- верификация требований к программному обеспечению;
- тестирование системной интеграции;
- верификация и валидация системы.

Отчет по валидации

Содержание отчета по валидации следующее:

- описание OTSS (системы, готовой к использованию) от поставщика;
- требования к использованию программного обеспечения:
    - требования к аппаратному обеспечению;
    - требования к программному обеспечению;
    - патчи;
- оценка риска и анализ опасностей;
- выбор поставщика;
- деятельность по установке;
- валидация:
    - сценарии тестирования использования программного обеспечения и результаты тестирования;
- список известных ошибок;
- управление конфигурацией:
    - обучение;
    - место установки;
    - обслуживание;
    - снятие с эксплуатации.

Выбор инструментов

- Стадия определения (идентификации):
    - предусмотренное применение;
    - планирование валидации;
    - планирование менеджмента риска (оценка рисков).
- Стадия имплементации:
    - меры по управлению риском;
    - аудит поставщиков.
- Стадия имплементации — развертывания:
    - установочная квалификация;
    - внутреннее обучение при необходимости;
    - окончательное приемочное тестирование.
- Стадия обслуживания:
    - планирование обслуживания;
    - анализ известных проблем.

Пример 5: Система автоматизированного тестирования программного обеспечения

Вводная информация

В данном примере изготовителем является производитель медицинского изделия класса С. Медицинские изделия этого изготовителя управляются программным обеспечением. Программное обеспечение архитектурно разделено на два основных компонента: консоль оператора и встроенное программное обеспечение управления в режиме реального времени. Консоль оператора является основным пользовательским интерфейсом системы. Встроенное программное обеспечение управления в реальном времени — это программное обеспечение, которое выполняет электромеханическое управление, сбор данных, синхронизацию и тому подобное. Программное обеспечение консоли оператора (находящееся в ПК под управлением стандартной операционной системы и базы данных) и встроенное программное обеспечение в режиме реального времени (находящееся на процессорной карте встроенной платы) взаимодействуют при помощи стандартного аппаратного обеспечения Transmission Control Protocol/Internet Protocol (TCP/IP) и протокольного интерфейса.

Менеджер по программному обеспечению проекта решил, что было бы полезно улучшить процесс разработки и тестирования программного обеспечения путем внедрения автоматизированного тестирования программного обеспечения. Он решил внедрить автоматизированное тестирование программного обеспечения сначала только для консоли оператора. Автоматизированное тестирование программного обеспечения будет проводиться как в точке тестирования интеграции, так и в точке тестирования системы программного обеспечения.

Определение применимости регулирующих требований к программному обеспечению

Рассматриваемое программное обеспечение для автоматизированного тестирования будет использоваться для проведения испытаний, что является обязательным требованием в процедурах разработки программного обеспечения изготовителя. Также оно предоставит свидетельства проведения требуемого регрессионного тестирования в точках интеграционного и системного тестирования. Исходя из этого, было установлено, что программное обеспечение для автоматизированного тестирования предназначено для автоматизации части процесса разработки и поэтому будет подлежать валидации в соответствии с требованиями ИСО 13485.

Определение процесса

Для лучшего понимания требований и рисков, связанных с внедрением программного обеспечения автоматизированного тестирования консоли оператора, менеджер по программному обеспечению определяет использование программного обеспечения автоматизированного тестирования в ходе процесса разработки программного обеспечения следующим образом.

При разработке программного обеспечения изделия планируется интегрировать в системное программное обеспечение различные модули в разное время. Кроме того, уже интегрированные в систему модули будут подвержены изменениям из-за исправления дефектов и изменения требований. Автоматизированную тестовую систему планируется использовать для регрессионного тестирования программного обеспечения интегрированной системы и для окончательного тестирования конкретного модуля в системе. План проекта программного обеспечения предусматривает интеграцию или обновление модулей два-три раза в неделю. Автоматические тесты будут выполняться в каждой из этих точек интеграции, чтобы убедиться, что новые функциональные возможности работают правильно и что ранее работавшие функциональные возможности не пострадали от нового добавленного кода или изменений в коде в конкретной сборке. Автоматизированные тесты будут выполняться на уровне тестирования программных систем для сборок, которые являются кандидатами на окончательный выпуск для валидации и в конечном счете для потребителей. Автоматизированное тестирование будет также использоваться в случае обнаружения дефектов на заключительных этапах разработки, которые необходимо исправить, чтобы обеспечить уровень регрессионного тестирования, который дополняет запланированное ручное тестирование.

Анализ риска

Менеджер программного обеспечения теперь проводит анализ для определения любых возможных последствий неправильного использования или функционирования программного обеспечения автоматизированного тестирования.

В первую очередь менеджер программного обеспечения должен оценить, сможет ли отказ процесса автоматизированного тестирования, отказ программного обеспечения автоматизированного тестирования или ошибка, сделанная кем-либо из пользователей, в конечном итоге привести к дефекту в медицинском изделии, который потенциально может быть источником вреда пациенту, оператору, стороннему лицу, специалисту по обслуживанию или окружающей среде.

- Самое большое беспокойство у менеджера вызывает то, что система автоматизированного тестирования программного обеспечения выдаст ложный результат о правильном функционировании тестируемого программного обеспечения консоли оператора, хотя в программном обеспечении все еще содержатся дефекты.

- Если необнаруженные дефекты находятся в критической области программного обеспечения, они могут вызвать неисправность в медицинском изделии, что может повлечь за собой причинение вреда.

- Менеджер по программному обеспечению понимает, что такой риск может возникнуть в результате неправильного менеджмента, использования автоматизированного тестового программного обеспечения или дефекта в самом автоматизированном тестовом программном обеспечении.

- Менеджер программного обеспечения решает, что чрезвычайно важно установить граничные условия, при которых можно использовать автоматизированную систему тестирования программного обеспечения, и для чего она может быть использована, чтобы группа разработчиков и группа тестирования программного обеспечения не чрезмерно полагались на систему.

- Лица, которые будут участвовать в настройке, программировании и эксплуатации автоматизированного программного обеспечения для тестирования, должны быть обучены для выполнения своих функций.

- Менеджер программного обеспечения считает, что при наличии средств управления указанными факторами потенциальные связанные риски будут снижены до допустимого уровня.

Определение предусмотренного применения программного обеспечения

Проанализировав возможное применение автоматизированного программного обеспечения для тестирования и связанные с ним риски, менеджер программного обеспечения готов сформулировать цель и предназначение автоматизированной системы тестирования программного обеспечения:

- автоматизированная система тестирования будет использоваться для тестирования сборок программного обеспечения в точках интеграции в процессе разработки;

- автоматизированная система тестирования будет использоваться для валидации тестирования и предварительных сборок выпуска в точках тестирования системы программного обеспечения;

- автоматизированная система тестирования будет выполнять регрессионное тестирование системы для обеспечения того, что рабочие процессы не подверглись негативному влиянию недавно введенного или измененного программного обеспечения;

- общая роль автоматизированной системы тестирования будет заключаться в обеспечении дополнительного регрессионного тестирования для проводимого ручного тестирования;

- для несложных и хорошо понятных рабочих процессов автоматизированная система тестирования может использоваться в качестве окончательного фактора, определяющего правильность программного обеспечения, учитывая, что конкретный протокол был верифицирован как соответствующий эквивалентному ручному тестированию;

- автоматизированная система тестирования будет использовать программное обеспечение, которое обеспечивает защитные меры (снижение риска) для системы с программным обеспечением или медицинского изделия в целом.

Планирование валидации

Менеджер программного обеспечения теперь имеет четкое представление об автоматизируемом процессе, о конкретном предусмотренном применении автоматизированной тестовой системы и о возможных связанных рисках. Менеджер программного обеспечения уже установил, что необходимы конкретные средства управления в отношении использования программного обеспечения. Также он установил, что в случае использования системы автоматического тестирования программного обеспечения в соответствии с предусмотренным применением и в сочетании с соответствующими средствами управления она будет иметь допустимый уровень риска.

- В данном случае менеджер программного обеспечения установил, что при правильном использовании автоматизированной системы тестирования программного обеспечения практически отсутствует риск влияния на возникновение дефектов в медицинском изделии. Менеджер определил, что правильное использование означает, что группа разработки и тестирования программного обеспечения не будет чрезмерно полагаться на использование системы для установления правильности программного обеспечения. Принимая во внимание определение риска как низкого, менеджер по программному обеспечению устанавливает, что требования к валидации системы будут характеризоваться низким уровнем усилий и строгости в отношении испытаний автоматизированной системы тестирования программного обеспечения.

Документация по валидации: подход к отчету по валидации

Подход, выбранный менеджером программного обеспечения, заключается в разработке отчета о валидации программного обеспечения для автоматизированной системы тестирования программного обеспечения, который будет включать в себя сводку всей деятельности, связанной с обеспечением необходимого уровня доверия к системе.

Использование критического мышления

Менеджер программного обеспечения далее устанавливает наилучший способ достижения необходимого уровня уверенности в том, что система будет использоваться надлежащим образом, а также не будет способствовать появлению серьезных дефектов в медицинском изделии.

Среди наиболее важных факторов достижения необходимого уровня доверия к системе он выделяет следующие:

- строгое следование определенному предусмотренному применению:

- обеспечение того, что весь персонал, участвующий в разработке и тестировании программного обеспечения, четко понимает граничные условия и соответствующее предусмотренное применение системы;

- составление документации: включить в отчет о валидации раздел с описанием конкретного предусмотренного применения и способы отражения этой информации в соответствии с планом разработки программного обеспечения;

- комплексная проверка:

- приобретение стандартной автоматизированной системы тестирования программного обеспечения у надежного поставщика, чьи системы тестирования используются для таких же или более критичных приложений;

- анализ предусмотренного применения системы поставщика, чтобы определить, подходит ли оно для использования по назначению;

- получение информации о том, как поставщик валидировал программное обеспечение перед выпуском в обращение. Получить подтверждение валидации коммерческого программного обеспечения у отдела качества поставщика. Это обеспечит уверенность в том, что система автоматического тестирования программного обеспечения была должным образом протестирована поставщиком и создаст базу для дополнительных действий менеджеру программного обеспечения и группам разработки и тестирования программного обеспечения;

- установление взаимоотношений с поставщиком с целью обеспечения уверенности в том, что менеджер программного обеспечения и команда разработчиков программного обеспечения и тестирования знают об известных проблемах и дефектах версии программного обеспечения для тестирования, которое они будут использовать;

- получение представления о планах поставщика по обновлению программного обеспечения, чтобы предусмотреть собственные планы перехода на новые версии программного обеспечения и деятельность по повторной валидации;

- документация: включить в отчет по валидации раздел с описанием результатов деятельности поставщика по проведению комплексной проверки, включая информацию о валидации поставщиком автоматизированной системы тестирования программного обеспечения, о способе доступа к списку дефектов и об ожидаемом переходе на новые версии программного обеспечения;

- установочное тестирование:

- убедиться, что компьютерная среда, в которой будет находиться программное обеспечение, соответствует спецификациям поставщика;

- установить первоначальный протокол высокоуровневого тестирования с целью обеспечения правильной установки программного обеспечения;

- документация: включить в отчет по валидации раздел с описанием результатов деятельности по установке;

- менеджмент риска:

- убедиться, что система будет использоваться только в соответствии с целями и назначением, которые определенны менеджером программного обеспечения;

- включить конкретные допустимые граничные условия в план разработки программного обеспечения для проекта, в котором будет использоваться автоматизированная система тестирования;

- провести анализ с целью определения точных зон охвата системы и убедиться, что ручное тестирование охватывает области, которые не охватывает автоматизированная система тестирования программного обеспечения;

- документация: включить в отчет по валидации раздел с описанием рисков, идентифицированных в ходе первоначального анализа риска, с указанием того, как каждый из этих рисков будет снижен;

- требования к использованию программного обеспечения:

- разработать перечень функциональных возможностей автоматизированной системы тестирования, которые будут использоваться. Разработанный менеджером программного обеспечения и группой разработки и тестирования программного обеспечения перечень будет поименован как «Требования к использованию программного обеспечения»;

- документация: включить «Требования к использованию программного обеспечения» в раздел отчета по валидации с описанием каждого из требований к использованию программного обеспечения;

- валидация автоматизированной системы тестирования:

- использовать «Требования к использованию программного обеспечения» для установления необходимого уровня доверия. Его можно установить, взяв три начальных сценария или протокола автоматизированного тестирования и выполнив параллельный тест для тех же протоколов, запущенных вручную. Три начальных сценария тестирования или протоколы реализуют все функциональные возможности, которые будет использовать команда;

- документация: включить в отчет по валидации раздел с обобщенными результатами параллельного тестирования, а также доказательства того, что результаты тестирования были эквивалентны;

- обучение:

- разработать программу обучения для всех пользователей системы, чтобы убедиться, что они полностью понимают, как использовать систему, и имеют право на ее использование. Менеджер программного

обеспечения считает, что обучение является одним из наиболее важных элементов, необходимых для обеспечения безопасного и результативного использования автоматизированной системы тестирования программного обеспечения;

- документация: включить в отчет по валидации раздел с описанием необходимого обучения для пользователей системы;

- валидация отдельных автоматизированных протоколов тестирования:

- там, где для тестирования программного обеспечения, предназначенного для снижения системных, аппаратных или программных рисков и опасностей, будет использоваться система автоматического тестирования, убедиться, что каждый протокол верифицирован с помощью параллельных автоматических и ручных тестов;

- там, где система автоматического тестирования будет использоваться для окончательного тестирования несложных и хорошо понятных рабочих процессов, убедиться, что каждый протокол верифицирован с помощью параллельных автоматических и ручных тестов;

- документация: убедиться, что записи по валидации программного обеспечения для медицинского изделия содержат доказательства проведения параллельного тестирования тестовых сценариев или протоколов, соответствующих данной категории;

- управление конфигурацией:

- убедиться, что установлена и используется только соответствующая валидированная версия программного обеспечения для автоматического тестирования;

- по мере того как новые версии программного обеспечения автоматизированного тестирования становятся доступными от их поставщика, следует обеспечить своевременное введение новых версий или изменений посредством управления их имплементацией;

- убедиться, что повторная валидация автоматизированной системы тестирования учитывает каждую контрольную точку обновления и что каждая повторная валидация системы проводится и документируется;

- документация: включить в отчет по валидации раздел, описывающий план управления конфигурацией системы.

Отчет по валидации

В результате предпринятой деятельности по укреплению доверия менеджер программного обеспечения составляет отчет по валидации для окончательного анализа и официального одобрения. В отчете содержатся логические обоснования и соображения по установлению дополнительных действий, которые могут понадобиться, чтобы менеджер программного обеспечения смог сделать вывод о том, что использование автоматизированной системы тестирования программного обеспечения не приведет к сценарию, при котором находящееся в разработке соответствующее медицинское изделие могло бы быть непреднамеренно сочтено несоответствующим. Отчет также содержит доказательства того, что все мероприятия, которые были установлены как важные, были проведены в соответствии с планом.

Отчет по валидации содержит следующее:

- определение процесса;
- анализ риска;
- менеджмент риска;
- предусмотренное применение;
- комплексная проверка поставщика программного обеспечения;
- обучение;
- установочное тестирование;
- валидация предусмотренного применения автоматизированной системы тестирования;
- обслуживание, повторная валидация и управление конфигурацией.

Анализ и официальное одобрение отчета по валидации

Менеджер по программному обеспечению направляет отчет по валидации менеджеру проекта, менеджеру по качеству программного обеспечения и менеджеру по тестированию программного обеспечения для анализа и официального одобрения.

Все рецензенты считают, что менеджер программного обеспечения четко продумал предусмотренное применение системы и понимает все риски, связанные с использованием системы. Рецензенты считают, что была выполнена вся деятельность, необходимая для обеспечения уровня доверия к системе, требуемого для использования системы. Рецензенты официально одобряют план. После этого система считается валидированной и вводится в эксплуатацию.

Пример 6: Простая электронная таблица

История вопроса

Лабораторные аналитики компании ZYX устали извлекать различные спецификации из своей системы управления документами для каждого продукта, который они анализируют, а затем вручную вычислять значение угла, которое им нужно сравнить со спецификацией. Для проведения инспекций в лаборатории используется инструмент. Этот инструмент измеряет три координатных положения, которые аналитики используют для вычисления угла, сравниваемого со спецификацией. Лаборатория недавно столкнулась с тремя случаями, когда аналитик неправильно рассчитал угол (из-за «толстых пальцев», как пояснил аналитик), и аналитики хотели бы предотвратить повторение этой ошибки. Они решают создать электронную таблицу для расчета угла и объединить спецификации для всех 50 продуктов, которые они анализируют, в эту электронную таблицу. Они будут вводить три пары координат, которые измеряются инструментом, выбирать наименование продукта из выпадающего меню и получать результат для приемки/отклонения. Аналитики также рассматривают дополнительный интерфейс для инструмента, чтобы передавать координаты непосредственно в электронную таблицу, но из-за стоимости интерфейса это улучшение откладывается до следующего года.

Определение процесса

Существующий процесс включает в себя следующие шаги:

a) замерить инструментом деталь;

b) записать три пары координат;

c) рассчитать угол;

d) извлечь спецификацию для детали из системы управления документов;

e) сравнить значение угла со спецификацией и установить соответствие или несоответствие;

f) положить лист с отметкой о соответствии или несоответствии на детали и отправить их на склад деталей.

Новый процесс будет включать следующие шаги:

a) получить электронную таблицу из системы управления документами;

b) замерить инструментом деталь;

c) ввести три пары координат в электронную таблицу;

d) визуально сравнить пары координат со значениями инструмента;

e) выбрать номер детали в электронной таблице;

f) выбрать «Рассчитать результат» в электронной таблице;

g) визуально проверить правильность выбора номера детали;

h) в зависимости от результата, положить лист с отметкой о соответствии или несоответствии на детали и отправить их на склад деталей.

Определение предусмотренного применения

Аналитики определяют цель и предназначение электронной таблицы следующим образом: электронная таблица будет принимать три внесенные пары координат, рассчитывать угол и затем сравнивать данный угол со спецификацией для выбранного продукта, выдавая результат о соответствии или несоответствии.

Анализ риска

Аналитики проводят мозговой штурм для идентификации возможных опасностей, связанных с электронной таблицей. Они устанавливают, что неправильный результат может означать, что не соответствующие спецификации детали могут быть использованы в производстве. Для того чтобы такие дефектные детали могли попасть в медицинское изделие и причинить вред конечному пользователю изделия, должно произойти по меньшей мере два других последующих отказа. Тем не менее все же существует незначительный, хотя и маловероятный, риск причинения вреда конечному пользователю. Поэтому существует небольшой риск производства продукта, который не соответствует техническим требованиям. Однако существует больший риск увеличения производственных затрат, поскольку, если дефектные детали используются в производстве, они не будут обнаружены до первой инспекции при сборке. В результате сборочный узел придется демонтировать. Кроме того, при получении ложного результата о несоответствии годные детали могут быть забракованы, что снова приведет к увеличению затрат на брак. Поэтому для решения проблем бизнеса в строгом порядке будут осуществлены дополнительные действия в виде разработки электронных таблиц, процедурного контроля, анализа документов и тестирования.

Планирование валидации

В связи с определением низкого уровня риска производства не соответствующего спецификации продукта, уровень затраченных усилий по проведению данной валидации будет также низким. Аналитики решают объединить требования к электронной таблице и план валидации в единый документ. Аналитики также решают объединить документацию по проекту с высокоуровневым планированием тестирования. Аналитики планируют проведение анализа данных документов всеми членами команды аналитиков (состоящей из четырех человек), а также представителем службы обеспечения качества. В дополнение к этому аналитики планируют проконсультироваться с техническими экспертами для разработки репрезентативного набора тестовых данных для обеспечения уверенности в том, что расчеты осуществляются должным образом. Технические эксперты также должны будут официально одобрить разрабатываемый документ.

Меры по управлению риском

Аналитики рассматривают каждый элемент в электронной таблице, который может привести к ошибке или неправильному результату. Для каждого такого элемента аналитики определяют возможные способы уменьшения риска (см. таблицу С.15).

Таблица С.15 — Пример 6 — Риски и способы их уменьшения

Риск	Уменьшение рисков
Неверные величины могут быть введены	Подтвердить каждую пару величин, введенных с помощью инструмента, через процедурный контроль. Для этого в новый процесс был добавлен шаг d)
Расчеты могут быть неверны	Удостовериться, что формула является правильной и что она предоставляет точные результаты, как необходимо
Продукт может быть выбран ошибочно	Подтвердить номер детали через процедурный контроль. Для этого в новый процесс был добавлен шаг g)
Макрос для обозначения результата может быть неверным	Удостовериться, что макрос верен и функционирует как предназначено
Спецификации в электронной таблице могут быть неверны	Удостовериться, что спецификации в электронной таблице соответствуют спецификациям на все 50 продуктов. Дополнить процесс внесения изменений в спецификации, чтобы при изменении спецификации требовалось обновление электронной таблицы. (Это никогда не происходило, но возможно.)
Формула расчета или макрос могут быть изменены после валидации	Валидированная электронная таблица с элементами управления конфигурацией будет помещаться в систему управления документами и извлекаться при необходимости. Элементы управления конфигурацией будут включать защиту паролем и блокировку ячеек для всех ячеек, не связанных с вводом данных

Задачи валидации

Используемая формула понятна, и разработчик имеет опыт разработки макросов электронных таблиц. Валидация будет подтверждать следующие элементы:

- расчеты;
- макрос;
- функцию блокировки ячеек (заблокированные ячейки не могут быть изменены);
- проверку ввода данных (значения в допустимом диапазоне, соответствующий выбор продукта, информативные сообщения об ошибках).

Поскольку электронная таблица одномоментно выдает только один результат, стресс-тестирование или тестирование производительности не требуется. Для всех тестов будет создан единый план тестирования и отчет. Отчет будет содержать готовую для применения электронную таблицу, а также подтвердит управление этой электронной таблицей в системе управления документами компании.

Развертывание

Перед внедрением новой системы процесс тестирования завершается, и операторы производства проходят аттестацию по эксплуатации новой системы обработки данных визуального контроля.

Инструменты из набора инструментов

- Определение требований (документированы в плане валидации).
- Отказ процесса и анализ рисков (документированы в плане валидации).
- Предполагаемое использование (документированы в плане валидации).
- План валидации.
- Планирование испытаний.
- Аттестация оператора.
- План технического обслуживания (что требуется для регрессионного анализа).

Обслуживание

Обслуживание электронной таблицы будет требоваться каждый раз, когда спецификация продукта изменяется или добавляется новый продукт. План эксплуатационных испытаний будет разработан с репрезентативным подмножеством полных проверочных тестов для обеспечения уверенности в том, что новые элементы не нарушают электронную таблицу. План технического обслуживания потребует регрессионного анализа с целью определения необходимости добавления дополнительных (специфичных для вносимых изменений) тестовых сценариев к имеющемуся набору тестов. Этот план также описывает, как обновить электронную таблицу (например, разблокировать ячейки, изменить, заблокировать повторно).

Пример 7: Электронная таблица (более сложная)

Описание программного обеспечения

Команда разработчиков программного обеспечения использовала электронную таблицу Microsoft Excel в качестве средства разработки. (Microsoft Excel упоминается только в качестве примера широко доступного на рынке и подходящего продукта. Эта информация приведена для удобства пользователей настоящего стандарта и не является одобрением ИСО этого продукта.) В эту электронную таблицу будут записывать проекты переводов сообщений для пользователей медицинских изделий класса С или D. Первоначально эти сообщения от изделия пользователям были написаны на американском английском языке. Последующие выпуски программного обеспечения этих изделий будут поддерживать семь языков. Электронная таблица состоит из семи столбцов. Самый левый столбец — это сообщение изделия на английском языке для каждого сообщения от изделия. Каждый из оставшихся столбцов представляет один из поддерживаемых международных языков, и каждая строка в столбце представляет перевод с английского на международный язык для конкретного англоязычного сообщения в крайнем левом столбце этой строки.

Предусмотренное применение

Электронная таблица удовлетворяет временные потребности по:

- визуальной организации сообщений на английском языке и их перевода,
- созданию электронной таблицы, которая может быть отправлена местным представителям для сбора переведенных сообщений либо непосредственно в виде электронной таблицы, либо в рукописном виде на бумажном носителе электронной таблицы, а также
- предоставлению временного средства хранения данных для переведенных сообщений.

После того как переводы собраны и переведены в программное обеспечение медицинского изделия, нет необходимости хранить или поддерживать электронную таблицу.

Вычисляемые ячейки или макросы не являются частью этой электронной таблицы.

Определение необходимости валидации

Excel используется просто для форматирования информации для распространения и сбора иноязычных переводов сообщений изделия. На первый взгляд, электронная таблица кажется простым приложением Excel, и возникает соблазн решить, что она не нуждается в валидации.

Задается следующий вопрос (5.2): «Могут ли отказ или скрытые недостатки программного обеспечения отрицательно повлиять на безопасность медицинских изделий или качество медицинских изделий?»

Ответ на этот вопрос очевиден — «да». Если программное обеспечение или электронная таблица выходит из строя таким образом, что искажаются переводы хранящихся там сообщений, то отказ может повлиять на безопасность изделия. Хотя команда считает, что вероятность сбоя для этого «простого приложения» является низкой, вероятность все еще находится в рамках требований по валидации ИСО13485.

Оценка риска

Если сообщения, содержащиеся в программном обеспечении изделия, переведены не должным образом, то пользователь может запутаться или неправильно их интерпретировать. Следовательно, существует возможность косвенного вреда пациенту, использующему изделие. Отказ программного обеспечения может быть обнаружен, и существует множество возможностей для перекрестных проверок в процессе разработки и валидации изделия для обнаружения и исправления любых отказов программного обеспечения.

Ниже приведены возможные виды отказов, которые могут негативно повлиять на программное обеспечение изделия:

- искажение оригиналов англоязычных сообщений, подлежащих переводу, в результате потери входного файла, потери отдельных сообщений, неправильного порядка сообщений, что приводит к потере контекста или искажению отдельных сообщений в результате случайной потери, замены или перестановки символов;
- искажение отдельных переведенных сообщений, подготовленных и собранных из региональных отделений. Повреждение может быть вызвано потерей входного файла, потерей отдельных сообщений, неправильным порядком сообщений, что приводит к потере контекста или повреждению отдельных сообщений путем случайной потери, замены или переноса символов. Существует дополнительная вероятность для повреждений в любом языке, требующем неанглийских шрифтов, если шрифты неправильно установлены в Excel;
- искажение таблицы собранных результатов, которая показывает аккумулированные результаты для каждого перевода. Повреждение может быть вызвано потерей входного файла, потерей отдельных сообщений, искажением сообщений, что приводит к потере контекста или повреждению отдельных сообщений путем случайной потери, замены или перестановки символов. В дополнение к неправильному порядку строк в электронной таблице может также произойти искажение столбцов. Столбцы, которые не отображают переведенные сообщения в собственных шрифтах и наборах символов, будут неправильно интерпретированы программистами при переводе сообщений в код.

Планирование валидации

Разработчики программного обеспечения признают потенциальный риск для пациента, если сообщения для нового изделия окажутся ошибочными. Тяжесть последствий такого отказа программного обеспечения может быть высокой. Необходимо предпринять усилия для создания уверенности в том, что сообщения, организованные в электронной таблице, представляют собой правильные переводы.

Однако Excel используется только для организации информации. Кажется маловероятным, что любой объем тестирования Excel обнаружит какие-либо дефекты, способные повредить сообщения. Когда инженеры рассматривают эту проблему, то обнаруживают, что к этому виду отказа гораздо чаще приводит человеческий фактор, чем простое приложение Excel.

Размышляя о человеческом факторе, инженеры понимают, что не существует настолько четко определенных процессов для сбора переводов или верификации того, что никакая ошибка человека не проникла в процесс.

Инженеры создают письменную процедуру сбора и верификации переведенных сообщений. Затем они рассматривают возможные риски отказа их процесса, как ошибки программного обеспечения (например, электронной таблицы Excel) могут способствовать этому отказу, и, наконец, что можно сделать для валидации процесса, включая электронную таблицу.

Меры по управлению риском

После более точного определения процесса сбора переводов инженеры определяют меры управления риском для защиты процесса от внесения ошибок в переводы сообщений.

Меры по управлению риском, которые защищают процесс сбора переводов, также защитят программное обеспечение от несоответствия его предусмотренному применению.

- Полученные из региональных отделений переводы должны предоставляться либо в бумажном (печатном) формате, либо в электронном формате с сопроводительной печатной копией. Если региональное отделение предоставляет электронную версию, то данные в этой электронной таблице будут сверяться (и документироваться) с печатной копией при ее передаче в сводную электронную таблицу перевода. Эта верификация защитит от любой неправильной интерпретации результатов, вызванной повреждением электронной таблицы во время передачи или различиями в особенностях шрифта между компьютером, поставляющим перевод, и компьютером, получающим переводы.

- После того как все переводы собраны и помещены в сводную электронную таблицу, электронная таблица в печатном виде должна быть направлена в каждое региональное отделение для рассмотрения и официального одобрения. Это официальное одобрение региональными отделениями защитит от любого неправильного толкования результатов, вызванного повреждением электронной таблицы во время передачи или различиями в особенностях шрифта между компьютером, поставляющим перевод, и компьютером, получающим переводы.

- После того как сводная электронная таблица будет принята всеми региональными отделениями, официально разработчиками и ответственными за обеспечение качества, печатная копия сводной электронной таблицы должна стать исходным материалом для процесса разработки программного обеспечения для изделия. Кроме того, печатная копия основной электронной таблицы должна быть источником любых ожидаемых результатов верификации переводов в программном обеспечении изделия.

Задачи по валидации

В дополнение к этим мерам по управлению риском необходимо выполнить другие задачи верификации и валидации для обеспечения того, чтобы программное обеспечение надлежащим образом соответствовало своему временному предусмотренному применению. Эти задачи заключаются в следующем:

- по каждому письменному переводу, полученному от региональных отделений, печатная копия обновленной сводной электронной таблицы должна быть сверена строка за строкой с печатной копией отдельной электронной таблицы перевода. Чтобы исключить любые ошибки перевода, вызванные различиями шрифтов между компьютерными платформами или принтерами, необходимо сверить печатные копии между собой;

- процесс управления версиями должен быть подробно задокументирован. Этот процесс должен конкретно учитывать следующее:

  - изменения в требованиях к сообщениям (например, на английском языке) по мере развития функциональных возможностей изделия;

  - изменения в сводном документе по мере того, как региональные офисы предоставляют переводы, а также анализируют и модифицируют обновленную сводную электронную таблицу;

- хотя электронная таблица очень проста, с ее использованием связаны некоторые вполне реальные риски управления версиями;

- конфигурация электронной таблицы должна включать номер версии самой электронной таблицы, версию используемого Excel, конфигурацию компьютерной платформы и конфигурацию принтера, используемого для создания печатной копии электронной таблицы. Полная конфигурация важна, поскольку различия шрифтов могут существовать в разных установках Windows или Office и в разных версиях встроенного ПО принтера. Единственный способ убедиться, что переводы не изменяются непреднамеренно, — использовать ту же конфигурацию при использовании электронной таблицы;

- конфигурация электронной таблицы (т. е. операционная среда и управление версиями) должна также управляться, чтобы предотвратить хаотичные, несогласованные изменения. Ответственность за принятие решения о внесении изменений в конфигурацию и документирование истории изменений возлагается на одного человека;

- версия каждой электронной таблицы должна быть отражена в ее печатной версии;

- таблицы перевода в программном обеспечении изделия должны указывать, какая версия основной электронной таблицы на бумажном носителе использовалась в качестве входных данных для программного обеспечения по переводу сообщений;

- индивидуальная задача по верификации перевода должна включать следующее:

- сообщение на английском языке должно быть проверено строка за строкой путем сравнения сводной версии и версии перевода электронной таблицы. Это сравнение защищает от любого повреждения (например, поврежденных или отсутствующих сообщений) файла электронной таблицы, которое могло произойти, когда файл был передан региональным отделениям и когда файл был возвращен региональными отделениями;

- после того как переводы вставлены в основную электронную таблицу (вручную или с помощью функции вырезания/вставки Excel), следует провести верификацию путем построчного сравнения полученного результата на бумажном носителе пересмотренной сводной электронной таблицы с печатной копией электронной таблицы перевода;

- когда программное обеспечение устройства тестируется для осуществления сообщений, процедуры тестирования должны использовать печатную копию последней версии сводной электронной таблицы (и должны ссылаться на номер версии) для сравнения осуществленных сообщений с предусмотренными сообщениями.

Все эти валидационные задачи должны быть задокументированы и собраны в качестве объективных свидетельств валидации процесса и таблицы Excel.

Этот подход к валидации приводит к 100%-ной верификации полученных результатов в сравнении с исходными данными программного обеспечения. Дальнейшее тестирование электронной таблицы не планируется. Несмотря на отсутствие традиционного тестирования, инженеры чувствуют себя уверенно в процессе своей деятельности и считают, что их обоснование валидации было ценной практикой. Инженеры считают, что любой отказ программного обеспечения будет обнаружен, и у них есть путь восстановления через печатные копии, которые собираются и записываются в соответствующих точках процесса. Печатные копии и документированные построчные проверки предоставляют документальные свидетельства деятельности.

Обслуживание

Электронная таблица предназначена для удовлетворения временных потребностей. Она должна быть удалена после того, как переведенные сообщения были встроены в код. Планы по обслуживанию не создаются.

Обсуждение

Предусмотренное применение и первоначальный анализ рисков, связанных с использованием электронной таблицы, имеют решающее значение для определения того, что электронная таблица требует дополнительного внимания с точки зрения валидации. При другом предусмотренном применении та же самая электронная таблица вполне могла бы привести к выводу, что использование электронной таблицы имеет низкий риск и низкую сложность. Если определение предназначенного применения заключалось бы просто в отслеживании хода сбора переводов (т. е. переводы в электронной таблице не использовались бы в деятельности по проектированию и имплементации), то можно было бы заключить, что практически не существует риска для целостности изделия, электронная таблица является лишь инструментом управления бизнесом и не подпадает под регулирующие требования по валидации.

«Процесс», который это программное обеспечение «автоматизировало», был частью процесса сбора данных, форматирования и хранения переводов сообщений для изделия. Пример интересен с нескольких точек зрения:

- валидация требовала крайне малого, почти незначительного тестирования программного обеспечения для валидации его применения. Важно отметить, что программное обеспечение (Excel) и электронная таблица были валидированы для этого конкретного использования, но не были валидированы в целом для любого использования. Команда сочла, что тестирование вряд ли выявит какие-либо дефекты в программном обеспечении, но существует уязвимость изделия, если в программном обеспечении каким-то непредсказуемым образом произойдет отказ;

- валидация состояла из 100%-ной верификации выходных данных электронной таблицы. Печатные версии использовались как «золотой стандарт». После того как копии были одобрены и учтены в файле проектирования и разработки, любой последующий отказ программного обеспечения был несущественным. Любой отказ до официального одобрения программного обеспечения будет идентифицирован в процессе анализа и одобрения;

- «процесс» был модифицирован, чтобы сделать его невосприимчивым к любому отказу программного обеспечения электронных таблиц;

- инженеры полагали, что вероятность ошибки со стороны человека намного выше вероятности отказа программного обеспечения в этом приложении. Пользователи могут делать опечатки, могут использовать неправильную версию электронной таблицы или совершать другие подобные ошибки. В этом случае «валидация программного обеспечения» также сделала процесс более невосприимчивым к ошибкам со стороны человека;

- этот пример подтверждает важность управления конфигурацией даже для обычных офисных средств повышения производительности.

Примечание — Этот пример был основан на реальном случае, который не был так тщательно обработан. В реальной ситуации возникли пользовательские ошибки, связанные с версиями электронных таблиц. Неожиданно проблемы, относящиеся к версиям шрифтов различных установок Excel на разных компьютерах, дали разные результаты в печатном виде (отличающиеся шрифты принтеров также оказались проблемным моментом). Таким образом, казалось бы, простая электронная таблица, валидация которой чуть было не была отклонена за ненадобностью, на самом деле стала проблематичной в своем искажении переводов сообщений.

Пример 8: Параметрический стерилизатор

Сотруднице Always-Safe Medical Device Company Мэри было поручено возглавить работу по валидации новой автоматизированной стерилизационной системы, которая будет специально разработана для использования в ее компании.

Определение процесса

Мэри начинает первое определение и документирование с обобщения того, что она знает о внедряемом на ее предприятии процессе стерилизации 100%-ным этиленоксидом (ЕТО):

- медицинские изделия помещаются в стерилизатор вручную. Этот процесс включает оценивание параметров цикла стерилизации для поддержания выпуска продукции по параметрам (параметрический выпуск);
- программное обеспечение автоматизированной системы стерилизации управляет циклом стерилизации;
- медицинские изделия вручную извлекаются после завершения цикла и переносятся в камеру для дегазации.

Анализ риска процесса

Мэри серьезно обеспокоена связанным с этим процессом риском, так как отказ процесса может иметь серьезные последствия, в том числе следующие:

a) неправильная стерилизация медицинских изделий. Отказ процесса может привести к серьезному вреду или смерти вследствие инфицирования при использовании нестерильной продукции;

b) потеря информации по истории изделия и прослеживаемости продукции;

c) выброс токсичных химических веществ в производственные помещения или окружающую среду. Отказ процесса может привести к серьезным травмам или смерти операторов стерилизаторов, а также людей, находящихся поблизости.

Поэтому Мэри рассматривает вопрос о том, какие меры по управлению риском следует принять и верифицировать для снижения этих рисков. Мэри считает, что риском можно управлять с помощью точного соблюдения техники выпуска продукции по параметрам процесса стерилизации с целью обеспечения того, что нужное количество газа используется в течение надлежащего периода времени при правильной температуре и правильной относительной влажности. Кроме того, визуальный контроль правильности параметрических значений от встроенных датчиков стерилизатора независимо подтвердит адекватность хода процесса стерилизации. Наконец, она посчитала, что для борьбы с утечкой химических веществ на объекте необходимы отказоустойчивые аварийные выключатели и защитные герметичные сооружения.

При выполнении данных мер по управлению риском только множественные одновременные отказы системы могут привести к появлению на выходе процесса нестерильного изделия. Однако, в связи с высокой тяжестью последствий отказов, Мэри определяет остаточный риск процесса как высокий. Именно поэтому необходим высокий уровень строгости при проведении валидации.

Определение цели и назначения программного обеспечения

Мэри хочет иметь детальное представление о том, как в данной системе будет использоваться программное обеспечение. В первую очередь она рассматривает, что именно должно делать программное обеспечение. В рассматриваемой системе программное обеспечение управляет процессом стерилизации медицинских изделий 100%-ным этиленоксидом (ЕТО), а также записывает информацию для включения в файл истории изделия и проводит анализ значений параметров стерилизации для поддержки выпуска продукции по параметрам. Для удовлетворения текущего спроса на продукцию был закуплен новый стерилизатор, камера которого вмещает партии большего размера, чем существующая система. Операторы процесса стерилизации совместно со службой обеспечения качества будут исследовать новую систему для установления критериев приемки выпуска медицинских изделий. Мэри понимает, что основные усилия будут направлены на контроль и мониторинг функционирования стерилизационной камеры в режиме реального времени во время циклов стерилизации, а также на сохранение информации в базе данных. Мэри с удовлетворением узнает о планируемом физическом размещении системы в стерилизационном подразделении и что система, как правило, будет отключаться один раз в неделю для обеспечения любого необходимого обслуживания.

Мэри устанавливает, что программное обеспечение будет автоматизировать все аспекты цикла стерилизации, от укладки вручную изделий в камеру до выгрузки вручную из камеры.

Мэри документирует цель и назначение программного обеспечения следующим образом:

- Программное обеспечение системы стерилизации будет осуществлять управление и мониторинг процесса стерилизации, а также будет оценивать параметры цикла стерилизации для поддержки выпуска продукции по параметрам (параметрический выпуск).

Планирование валидации

Теперь, когда Мэри понимает, для чего предназначено программное обеспечение, она готова разработать план валидации с высоким уровнем строгости. Она знает о необходимости последующей подробной детализации, но хочет начать планирование валидации сейчас, чтобы она могла заранее обнаружить риски отказа программного обеспечения и рассмотреть идентифицированные риски при завершении своего планирования.

Из-за определенного ранее высокого остаточного риска процесса Мэри считает, что она должна обеспечить высокий уровень детализации и соблюдения формальности в процессе валидации. Она рассчитывает использовать высокий уровень строгости и детализации в документации и рассматривает создание большинства документов в качестве самостоятельных, а не объединенных, как это часто делают при меньшем требуемом уровне усилий. Из-за высокого риска рассматриваемой системы она решает относиться к разработке с тем же уровнем строгости, что она будет использовать для разработки программного обеспечения медицинских изделий. Поэтому она решает следовать требованиям МЭК 62304:2006 (включая Изм. 1:2015) в качестве методологии управления жизненным циклом. В качестве руководства по менеджменту риска программного обеспечения она ссылается на МЭК/ТО 80002-1. Кроме того, для обеспечения рассмотрения всех потенциальных источников вреда Мэри решает применить в процессе разработки анализ древа неисправностей программного обеспечения. Она также решает формально определить и документировать требования к бизнес-процессам пользователей и требования к программному обеспечению. Любой функционал, представляющий особое беспокойство, будет специфически идентифицирован. Мэри также планирует проведение официального анализа требований к программному обеспечению. Официальное одобрение должно быть проведено отделом обеспечения качества, инженером по стерилизации и менеджером по стерилизации. Из-за критичности и высокого риска данной системы окончательное одобрение отчета по валидации будет проведено при участии членов высшего руководства.

Определение требований к программному обеспечению

Теперь Мэри составляет в письменном виде определение требований к программному обеспечению. Она решает, что требования к программному обеспечению должны касаться аварийных сигналов, обработки ошибок и сообщений, подтверждения параметрических настроек, интерфейса к системе записей истории изделия, управления и мониторинга датчиков, управления и мониторинга перемещений.

Установление уверенности и управления программным обеспечением

Мэри помещает всю разработку жизненного цикла под управление внутренних процедур управления разработкой Always-Safe. Поскольку все делается внутри организации, никакой деятельности в отношении поставщика осуществлять не требуется.

Определение границ программного обеспечения с другими системами

Затем Мэри рассматривает возможное взаимодействие нового стерилизатора с другими системами. Она устанавливает, что единственное взаимодействие будет происходить с существующей системой баз данных Always-Safe, которая будет хранить данные, сформированные во время циклов стерилизации.

Анализ рисков отказов программного обеспечения

Несмотря на уже установленный Мэри высокий риск автоматизируемого бизнес-процесса, ей все еще необходимо провести анализ риска отказа программного обеспечения. Используя настоящий стандарт в качестве эталонного референтного материала, Мэри выбирает для этой деятельности количественную модель определения риска. Распределение (ранжирование) новой системы по уровням тяжести и вероятности она проводит следующим образом:

- риск по уровню «тяжести» высок (10), потому что отказ системы может привести к смерти или серьезной травме;

- риск по уровню «вероятности» также высок (10), так как отказ самого программного обеспечения может привести к причинению вреда, поскольку именно программное обеспечение принимает решение о приемке в отношении стерилизации.

Она определяет значение риска как 20, что по установленным критериям относит его к недопустимому риску. Такие результаты оценивания риска означают, что должна применяться самая строгая методология валидации. Применяемая методология должна быть столь же строгой и всеобъемлющей, как если бы стерилизатор сам был медицинским изделием.

В результате снижения остаточный риск данной автоматизированной системы должен быть настолько низок, насколько это возможно и практически осуществимо. Из-за высокого уровня тяжести вреда, исходящего от системы, стерилизация по своей сути является процессом с высоким уровнем риска. Следовательно, также будет выполняться дополнительная, связанная с риском деятельность, указанная в МЭК/ТО 80002-1.

Завершение плана валидации

Поскольку Мэри уже завершила определение требований к программному обеспечению, оценила риск и определилась с подходом к имплементации программного обеспечения, то теперь она имеет достаточно информации для завершения детального плана валидации.

Составляя первый проект своего плана валидации, Мэри решила применять строгий подход к менеджменту риска. Она уже запланировала обеспечение высокого уровня детализации и соблюдения формальности в процессе валидации.

Соответственно, она подбирает инструменты менеджмента риска (установленные в МЭК/ТО 80002-1), которые она планирует использовать следующим образом.

- Инструменты менеджмента риска:
  - анализ древа неисправностей в программном обеспечении;
  - план менеджмента риска;
  - идентификация мер по управлению риском в рамках производственного или бизнес-процесса;
  - анализ отказа программного обеспечения (анализ риска).

Затем Мэри размышляет о том, как она обеспечит уверенность в программном обеспечении на этапах проектирования, разработки и компоновки (настройки конфигурации) программного обеспечения. Она уже решила следовать МЭК 62304:2006 (включая Изм. 1:2015) для управления жизненным циклом. Теперь она идентифицирует другие конкретные инструменты, которые будут использованы для обеспечения корректной разработки программного обеспечения на этапах его проектирования, разработки и компоновки (настройки конфигурации).

- Инструменты проектирования, разработки и конфигурации:
  - МЭК 62304:2006 (включая Изм. 1:2015) Анализ документации и архитектуры;
  - проектные спецификации программного обеспечения;
  - детальное проектирование и анализ программного обеспечения;
  - стандарты написания кода программного обеспечения;
  - матрица прослеживаемости;
  - идентификация мер по управлению риском в рамках разработки программного обеспечения системы;
  - анализ и верификация кода;
  - анализ проектирования и разработки.

Мэри не сомневается, что ей придется тщательно протестировать новую разрабатываемую систему. Во-первых, она решает наряду с обычным модульным тестированием, интеграционным тестированием и тестированием интерфейсов предусмотреть еще и формальную деятельность по планированию проведения испытаний. Однако, поскольку эта система будет выпускать готовые изделия в режиме реального времени, она решает расширить границы тестирования системы посредством добавления тестирования устойчивости, тестирования производительности и более обширное тестирование набором входных данных с целью моделирования как можно больших вариантов реальных условий работы.

- Инструменты испытаний (тестирования):
  - планирование проведения испытаний (тестирования);
  - модульное тестирование (юнит-тесты);
  - интеграционное тестирование;
  - тестирование программных интерфейсов;
  - регрессионное тестирование (при необходимости);
  - системное тестирование;
  - тестирование устойчивости (стресс-тесты);
  - тестирование набором входных данных;
  - тестирование производительности.

Мэри знает, что ее работа с системой не будет завершена до полной реализации системы в реальной производственной среде. Поэтому она обращает свое внимание на планирование деятельности по валидации на этапе развертывания. Она хочет быть уверенной в том, что система надлежащим образом документирована и что пользователи хорошо обучены ее правильному использованию. Также она хочет быть уверенной в правильной установке системы. Таким образом, план валидации Мэри на этапе развертывания дополняется следующими пунктами:

- инструменты развертывания:
  - анализ пользовательских процедур;
  - внутреннее обучение;
  - квалификация правильности установки (IQ);
  - операционная квалификация (QQ);
  - эксплуатационная квалификация (PQ);
  - аттестация оператора.

Планирование обслуживания

Мэри внимательно относится к обслуживанию программного обеспечения из-за высокого остаточного риска. Она планирует несколько видов деятельности по обслуживанию для обеспечения качества программного обеспечения после развертывания системы, включая оценку результативности обучения пользователей, методы мониторинга системы, проверку правильности выходных данных системы и отчетность о несоответствиях. Она также проверяет и подтверждает, что калибровка и другая деятельность по обслуживанию оборудования происходят в дополнение к деятельности по обслуживанию программного обеспечения.

Деятельность по снятию с эксплуатации

Мэри испытывала трудности с выводом предыдущей системы из эксплуатации, так как сгенерированные этой системой данные должны были быть архивированы для сохранения записей истории изделия, а старый формат не был совместим с новым форматом данных. Новая система использует универсальный формат данных для обеспечения гибкости при переносе оставшихся от предыдущей системы данных в новую систему.

Пример 9: Система отчетности о несоответствующих материалах — полное обновление системы

Корпорация Advanced Medical Specialities модернизирует программное обеспечение системы отчетности о несоответствующих материалах (NCMRS), которая является коммерческим программным пакетом. Advanced Medical решила не обновлять систему в прошлые разы, поэтому теперь она отстает на две версии. (Advanced Medical в настоящее время работает под управлением версии 2, а последней версией является версия 4.) Advanced Medical требуется обновление для поддержания текущего соглашения о техническом обслуживании программного обеспечения. Версия 4 программного обеспечения NCMRS-Pro значительно изменилась по сравнению с предыдущими версиями. Помимо прочего, продукт был перенастроен с типичного клиент-серверного приложения на веб-приложение. Новое программное обеспечение также включает в себя значительные новые возможности и функции. Фрэнк, владелец бизнес-процессов и менеджер проектов в Advanced Medical, не имеет новых требований к существующему программному обеспечению и процессу, но он хочет воспользоваться новыми функциями программного обеспечения.

Фрэнк консультируется с группой по регулированию и решает, что нынешний интерфейс между системой ERP и NCMRS может оставаться как есть, без каких-либо модификаций. Фрэнк признает, что новая версия способна записывать данные обратно в систему ERP, а также что ее расширенный интерфейс должен быть тщательно проверен при валидации. Фрэнк и его коллеги, инженер по качеству производства и группа по регулированию, начинают работу по определению масштаба предполагаемых усилий по валидации. Эти сотрудники упоминаются как «команда» в остальной части этого примера.

Определение процесса

Фрэнк начинает с анализа существующего руководства по процессу с целью определения того, какие элементы рабочего процесса будут автоматизированы новым программным обеспечением. Новое программное обеспечение изменит следующие функции:

a) идентификацию потенциальных несоответствующих материалов или продукции (вне области валидации);

b) ввод информации, относящейся к материалу и обстоятельствам, связанным с его обнаружением (входит в область рассмотрения при валидации);

c) направление информации для обеспечения надлежащей идентификации, оценивания, исследования и размещения материала (входит в область рассмотрения при валидации);

d) распространение информации среди заинтересованных сторон и других компьютерных систем, которая необходима для надлежащей обработки финансовых, закупочных, плановых и календарных операций (входит в область рассмотрения при валидации);

e) физическое расположение материалов, хотя соответствующие данные о расположении будут записаны в системе (вне области валидации).

Анализ риска процесса

Фрэнк осознает, что этот процесс и поддерживающее программное обеспечение содержат в себе некоторый риск. Отказ процесса может иметь серьезные последствия, в том числе следующие:

- непреднамеренная поставка в производственную зону несоответствующих материалов;
- непреднамеренный выпуск в обращение несоответствующей продукции;
- увеличение стоимости производства из-за возросших отходов, переработки и тому подобного.

Фрэнк и группа по регулированию рассматривают, какие меры по управлению риском применяются для их снижения, включая следующие:

- средства управления в существующих процедурах по обнаружению, отделению, контролю и исправлению несоответствующих материалов;
- анализ менеджмента и качества данных по статистическому управлению процессами, а также других мер по идентификации развивающихся тенденций, которые могут сигнализировать о том, что процессы не находятся под надлежащим управлением;
- постоянное обучение операторов для обеспечения соблюдения процедур;
- финансовые отчеты, помогающие выявить использование материалов, которые приводят к нехарактерным проблемам с функционированием производственных процессов.

При наличии таких мер по управлению риском только множественные одновременные отказы системы могут привести к неспособности надлежащим образом управлять несоответствующей продукцией или материалами. Однако, учитывая потенциальное влияние последствий таких отказов на качество, соответствие регулирующим требованиям и финансовые показатели, Фрэнк приходит к заключению, что остаточный риск процесса потребует осуществления достаточно строгой деятельности по укреплению доверия, направленной на обеспечение правильного функционирования программного обеспечения и его соответствия предусмотренному применению.

Определение цели и назначения программного обеспечения

Фрэнк хочет иметь подробное представление о том, как обновление программного обеспечения повлияет на его пользователей и его организацию. Фрэнк приходит к выводу, что программное обеспечение по существу является автоматизированным инструментом отслеживания и управления проблемами. Производственный

персонал, работающий со стандартными инструментами, оборудованием и прочими инструментами, несет ответственность за идентификацию и изоляцию потенциально несоответствующей продукции и материалов. Как только проблема исследована, сведения о ситуации вводятся в программное обеспечение. Затем программное обеспечение управляет рабочим процессом, назначениями и уведомлениями для решения проблемы и документирует ту или иную деятельность, необходимую для обработки и размещения продукции и материалов. Обновление программного обеспечения должно упростить процесс, сделав его более эффективным, и предоставить группе по обеспечению качества более мощные инструменты для анализа данных и тенденций, а также обеспечить большую наглядность проблем в области качества. Фрэнк понимает, что необходимые для обновления изменения процесса в первую очередь коснутся самого рабочего процесса и процесса распространения информации. Программное обеспечение само по себе не принимает окончательных решений и не определяет какие-либо результаты независимо, но хранит и документирует решения, принятые людьми, взаимодействующими с системой.

Фрэнк устанавливает, что программное обеспечение автоматизирует аспекты рабочего процесса управления несоответствиями. Группой по регулированию было документировано следующее заявление о цели и назначении программного обеспечения.

- Программное обеспечение NCMRS предназначено для поддержки процесса управления несоответствующей продукцией и материалами. Система используется для документирования шагов процесса, которые установлены в соответствующих стандартных операционных процедурах, и для ведения записей, фиксирующих выполненные шаги процесса, время их выполнения и выполняющий их персонал, а также результат каждого шага. Система предоставляет данные для мониторинга деятельности в области качества и улучшения.

Определение границ взаимодействия программного обеспечения с другими системами

Программное обеспечение NCMRS имеет два интерфейса, включая один основной интерфейс с ERP-системой и один вторичный интерфейс с системой управления персоналом (HR) компании. Основной интерфейс представляет собой пакетный процесс и разработан для планомерного обновления системы два раза в день данными о готовой продукции, продукции в процессе производства, ведомости материалов и ведомости операций. Интерфейс также будет поставлять данные для отчета о несоответствующих материалах (NCMR) обратно в ERP с данными о качестве хранения, расположении материалов и другой транзакционной информацией. Вторичный интерфейс однонаправлен от системы HR и предназначен для обновления данных о сотрудниках NCMR для целей планирования и назначения.

Первоначальное планирование валидации

Фрэнк приобрел дополнительную уверенность в том, что функционирование процесса и программного обеспечения NCMR адекватно понято и должным образом документировано. Теперь он готов разработать первоначальный план валидации. Дополнительные детали будут разработаны в процессе планирования.

Группа по регулированию определяет документы, которые будут иметь наибольшую ценность и надлежащим образом конкретизируют, что должно делать программное обеспечение. На эти документы будут ссылаться как на «требования», однако они не будут являться типичным набором требований пользователя как таковых. Вместо этого документы будут представлять собой серию подробных описаний того, как программное обеспечение должно функционировать. Таким образом, анализ автоматизированного и ручного тестирования будет более качественным и удобным для проверки и оценки результатов. Вместо рассмотрения отдельных тестов для определения выполнения конкретных требований пользователей, проводимый анализ будет совокупно рассматривать как предусмотренное функционирование, так и сами результаты функционирования системы.

Набор документов будет включать следующее:

a) документирование рабочих процессов и бизнес-правил. Эту область программного обеспечения можно конфигурировать, поэтому команда запланирует подготовку необходимого набора конфигураций и разработает подробные технологические потоки и логические диаграммы, описывающие операцию;

b) документирование интерфейса. Данные документы будут описывать, какие элементы данных перемещать из систем ERP и HR в NCMRS, какие элементы данных перемещать из NCMRS в систему ERP и когда именно перемещать данные элементы;

c) документация по миграции данных. Этот набор документов будет описывать, какие архивные данные будут перемещены в обновленную систему.

План валидации будет включать результаты анализа и официального одобрения каждого из этих документов. Официальное одобрение потребуется от группы по обеспечению качества, от инженерно-производственного отдела и группы информационных систем. Из-за критической значимости и риска системы все члены высшего руководства должны провести окончательное официальное одобрение отчета по валидации.

Определение требований к использованию программного обеспечения

Фрэнк и члены команды приступают к формированию набора указанной выше документации, ссылаясь при этом на документацию по системе, предоставленную поставщиком, и предыдущую документацию по существующим интерфейсам.

Установление деятельности по укреплению доверия и управлению программным обеспечением

У Фрэнка уже был положительный опыт работы с этим программным обеспечением и этим поставщиком. Теперь Фрэнк определяет следующие пять основных направлений, по которым команда будет прилагать усилия с целью укрепления доверия к программному обеспечению:

a) в соответствии с внутренними политиками и процедурами Advanced Medical поставщик имеет в компании статус официально одобренного поставщика. Предыдущие аудиты показали, что поставщик имеет адекватные систему качества и жизненный цикл продукции (SDLC). Поставщик выпускает коммерчески доступное программное обеспечение, имеющее устоявшуюся историю в регулируемой отрасли, которое имеет предназначенное применение, аналогичное предусмотренному применению Advanced Medical. Поставщик будет периодически подвергаться аудиту для поддержания статуса официально одобренного поставщика;

b) Advanced Medical будет использовать поставляемый поставщиком инструмент автоматизированного тестирования для верификации правильности установки программного обеспечения и его функционирования в рамках комплекта тестов. Этот инструмент может обрабатывать более 8000 различных транзакций за несколько часов. Однако инструмент не проверяет некоторые параметры конфигурации, которые компания планирует включить;

c) команда подготовит дополнительный план испытаний, который включает параллельную обработку статистически значимой выборки фактических отчетов о несоответствиях на бумажном носителе. Результаты будут проанализированы для обеспечения точности, целостности данных и соблюдения процедур;

d) команда будет верифицировать преобразование данных и перенос существующих системных записей с помощью метода выборки, обеспечивающего сохранение целостности архивных записей. Количество записей будет использоваться для верификации 100 % преобразований данных;

e) интерфейсы данных будут верифицированы с использованием метода выборки для измерения полноты и точности передачи данных.

Анализ рисков отказа программного обеспечения

С целью установления необходимой степени строгости валидации Фрэнк использует настоящий документ как ссылочный. Отказы программного обеспечения могут привести к неправильной обработке электронных записей, а также к их потере или повреждению. Снижение рисков находится под управлением внутренней системы качества поставщика, установочной квалификации программного обеспечения (инструмент автоматизированного тестирования), а также дополнительными проверками и тестированием вариантов использования. Вследствие наличия управления процессами на последующих этапах, остаточный риск этой системы считается настолько низким, насколько это практически возможно.

Завершение планирования валидации

Проведенный анализ предписывает применение довольно строгой методологии валидации. Применяемая методология в разумной степени обеспечивает, что программное обеспечение будет функционировать в соответствии с предусмотренным применением. Члены команды приходят к выводу, что они надлежащим образом определили требования системы, приняли решение о подходе к имплементации, проанализировали риск программного обеспечения и получили достаточно информации для создания детального плана валидации.

Основная часть тестирования будет выполнена с использованием набора автоматизированных тестов, который команда уже рассмотрела и установила его пригодность в отношении цели и назначения программного обеспечения. Дополнительное вспомогательное тестирование будет проводиться на основе реальных вариантов использования, составленных в производственном подразделении. Целью этих тестов является: a) проверка того, что процесс функционирует так, как предназначено, b) снижение временных затрат на принятие системы пользователями и их обучение и в) верификация того, что изменения конфигурации не оказали отрицательного влияния на программное обеспечение. Дополнительное тестирование не предназначено для замещения внутренней системы тестирования поставщика, которое ранее было верифицировано посредством проведения аудита. Успешное завершение автоматизированного тестирования позволит установить, что программное обеспечение установлено правильно и является функционально приемлемым.

Команда выбирает следующие инструменты из настоящего документа, чтобы выполнить оставшиеся действия по установке, конфигурации, тестированию, верификации и валидации.

- Инструменты проектирования, разработки и конфигурации:
  - анализ документированной архитектуры;
  - идентификация мер по управлению риском при проектировании программных систем;
  - анализ предлагаемой конфигурации;
  - анализ списка «известных проблем» поставщика;
  - анализ документации по валидации базовой системы поставщика;
  - анализ схем рабочих процессов стандартного программного обеспечения «в установочной конфигурации»;
  - анализ стандартной библиотеки отчетов «в установочной конфигурации»;
  - анализ пробелов в изменениях конфигурации стандартных рабочих процессов и рабочих правил.

- Инструменты тестирования:
  - планирование тестирования;
  - описание и результаты поставляемого поставщиком средства автоматизированного тестирования для верификации и квалификации установки;
  - тестирование установки и производительности (часть набора автоматизированных тестов);
  - тестирование сценариев использования на основе реальных записей несоответствий в качестве входных данных, а не искусственно созданных тестовых сценариев;
  - план выборки для верификации перенесенных данных;
  - проверки системы для верификации рабочих интерфейсов.
- Средства развертывания:
  - анализ пользовательских процедур;
  - внутреннее обучение;
  - аттестация оператора.

Планирование обслуживания

Фрэнк планирует использовать несколько мероприятий по техническому обслуживанию для обеспечения постоянного качества программного обеспечения после развертывания системы, включая оценку результативности обучения пользователей, методы мониторинга системы, периодический аудит результатов работы системы и отчетности о дефектах как внутри компании, так и у поставщика. Фрэнк договорился с поставщиком о том, чтобы уведомления об ошибках, выпуске обновлений и другие сообщения доводились до сведения соответствующих сотрудников, ответственных за обслуживание программного обеспечения в Advanced Medical.

Деятельность по снятию с эксплуатации

Фрэнк планирует сохранить текущую систему доступной после завершения эксплуатации, чтобы иметь возможность сравнить пропускную способность и результаты, из которых могут быть скомпилированы показатели производительности. После того как модернизированная система будет успешно функционировать в течение 6 месяцев, Фрэнк полностью снимет с эксплуатации предыдущую систему.

Пример 10: Программное обеспечение для планирования заседаний комиссии по рассмотрению отчетности о несоответствующих материалах (NCMR)

Компания со штатом в 1000 сотрудников решает опробовать новое программное решение, которое поможет компании в электронном виде планировать совещания для проведения обязательного анализа NCMR. Проектная группа, назначенная для реализации автоматизации, изучает недавно выпущенное на рынок коммерческое программное обеспечение. Поставщик утверждает, что программное обеспечение может планировать совещания посредством использования данных, полученных через другие компьютерные системные интерфейсы. Проектная группа решает, что это программное обеспечение может быть полезным для планирования заседаний комиссии по рассмотрению NCMR, если программное обеспечение будет способно собирать данные NCMR из валидированной NCMR-системы базы данных компании.

Определение процесса

Команда собирается для обсуждения процесса планирования совещаний комиссии по анализу NCMR и рассматривает процедуры обработки NCMR компании. Обсуждение приводит к следующему определенному процессу:

a) как только несоответствие идентифицировано, соответствующий материал маркируется, изолируется и регистрируется в валидированной базе данных NCMR;

b) проводятся еженедельные совещания по анализу результатов всех расследований, связанных с несоответствиями и рекомендуемыми действиями по дальнейшему обращению;

c) для каждого совещания определяется перечень подготовленных к анализу несоответствий NCMR, а также лица, которые должны присутствовать, представлять результаты и участвовать в действиях по одобрению и дальнейшему обращению;

d) за день до совещания комиссии по анализу NCMR ее участникам направляется уведомление о заседании. Данное уведомление включает перечень NCMR для обсуждения.

Анализ риска процесса

Посредством мозгового штурма члены команды оценивают потенциальный вред, который может быть причинен вследствие возможных отказов процесса:

- уведомление о совещании не отправлено;
- уведомление о совещании отправлено не вовремя;
- к участию приглашены не те лица;
- перечень NCMR для анализа составлен некорректно.

Группа отмечает, что выпущенная процедура обработки NCMR требует назначения конкретного лица в качестве менеджера по обработке NCMR. Это лицо отвечает за своевременную обработку всех NCMR и публикацию показателей процесса NCMR на основе данных, содержащихся в валидированной базе данных NCMR. Во всех выявленных группой случаях негативным результатом использования программного обеспечения для планирования совещаний является нарушение эффективности проведения совещаний комиссии по NCMR. Это нарушение накладывает дополнительную временную нагрузку на менеджера по обработке NCMR. Таким образом, риск отказа процесса анализа определяется как низкий с точки зрения риска нарушения регулирующих требований, риска для окружающей среды и риска причинения вреда людям.

Определение предусмотренного применения

Команда определяет цель и назначение применения программного обеспечения, применимые регулирующие требования и разграничения следующим образом.

- Использование программного обеспечения:
  - «кто?» — программное обеспечение в первую очередь будет использоваться менеджером по процессу обработки NCMR;
  - «что?» — программное обеспечение автоматически разошлет электронные приглашения на совещание лицам, которые идентифицированы как его участники на этой неделе;
  - «когда?» — программное обеспечение будет использоваться, когда необходимо запланировать совещания по анализу NCMR;
  - «куда?» — поскольку все участники являются локальными, программное обеспечение должно использоваться только в локальной сети (LAN);
  - «каким образом?» — программное обеспечение извлекает перечень NCMR, которые открыты и нуждаются в рассмотрении комиссии по анализу NCMR. Менеджер по обработке NCMR идентифицирует те NCMR, которые будут рассмотрены на следующем совещании. Затем программное обеспечение использует таблицу, настроенную менеджером по обработке NCMR, с целью идентификации круга лиц, которые должны присутствовать на данном совещании. Дата предстоящего совещания устанавливается менеджером по обработке NCMR. За один день до совещания программное обеспечение отправляет электронное приглашение соответствующим участникам;
  - «для чего?» — программное обеспечение будет использоваться для улучшения своевременного уведомления соответствующих лиц для участия в еженедельных совещаниях комиссии по анализу NCMR.

- Границы взаимодействия:
    - данное программное обеспечение взаимодействует с базой данных NCMR и графическим интерфейсом пользователя.
- Применимые регулирующие требования:
    - программное обеспечение не хранит информацию, используемую для подтверждения соответствия каким-либо регулирующим требованиям. Вся информация по записям истории изделия, связанная с NCMR или обработкой NCMR, записывается либо на бумажном носителе, либо в валидированной базе данных NCMR.

После создания и анализа данного определения о целях и назначении применения команда устанавливает, что предлагаемое программное обеспечение не автоматизирует деятельность, которая является обязательной для выполнения регулирующих требований и не создает записи по качеству, требуемые существующим регулированием. Хотя оно используется для обеспечения большего удобства проведения совещаний, которые являются частью деятельности, требуемой регулирующими органами (процесс NCMR), само программное обеспечение не автоматизирует требуемую деятельность. В результате группа документирует вышеуказанное предусмотренное применение и четко указывает, что формальная валидация не требуется. Однако команда также признает, что даже небольшое изменение в использовании на стадии обслуживания может существенно повлиять на первоначальное решение команды о необходимости валидации. Например, если программное обеспечение используется для хранения протоколов совещаний или для составления списка присутствовавших на нем лиц для их рассмотрения регулирующим органом, это повлияет на первоначальное решение о необходимости валидации. Поэтому команда обновляет свои процедуры системы качества, чтобы установить требование по оцениванию предусмотренного применения на периодической основе или в результате изменений связанных процессов.

Использование инструментов

Были использованы следующие инструменты.

- Стадия разработки — определения:
    - определение требований к процессу;
    - анализ риска отказа процесса;
    - определение предусмотренного применения.
- Стадия обслуживания:
    - планирование обслуживания.

Обсуждение

В результате определения конкретного использования и границ деятельности, автоматизированной данным программным обеспечением, команда может обоснованно заявить, что программное обеспечение не соответствует определению программного обеспечения, используемого в рамках систем качества медицинских изделий и, следовательно, программное обеспечение не нуждается в валидации. При идентификации такого программного обеспечения следует проявлять большую осторожность для обеспечения того, что фактическое использование программного обеспечения полностью охватывается определением предусмотренного применения. Также важно понимать, что предусмотренное применение может легко измениться на стадии обслуживания даже без изменений в самом программном обеспечении. Поэтому планирование технического обслуживания является важной частью обеспечения надлежащего управления программным обеспечением, используемым компанией.

Пример 11: Система управления перечнем одобренных поставщиков

Корпорация Acme является изготовителем медицинских изделий класса B. Корпорация использует процедуру для ведения перечня официально одобренных поставщиков (AVL), которая выполняется вручную. Корпорация Acme хочет разработать систему AVL для автоматизации процесса проверки, что поставщик официально одобрен для поставки конкретной детали.

- Джек, менеджер проекта компании Acme для новой системы AVL, определяет рассматриваемый процесс AVL как процесс системы качества медицинских изделий, связанный с управлением закупками в ИСО 13485.

Таким образом, предлагаемая система AVL подпадает под требования по валидации применения программного обеспечения.

Определение процесса

Для лучшего понимания требований и рисков, обусловленных разработкой системы AVL, Джек определяет связанные бизнес-процессы следующим образом:

a) когда инженерная группа хочет одобрить нового поставщика, образцы деталей поставщика направляются в группу качества для квалификационной проверки;

b) после квалификации деталей поставщика группа качества направляет группе закупок электронное письмо, разрешающее ввод наименования поставщика, а также утвержденных номеров деталей и описаний в перечень одобренных вендор-поставщиков (AVL). Этот перечень ведется и поддерживается в бумажном виде в группе закупок. Инспекционная группа по приемке имеет доступ к AVL;

c) группа закупок вручную выполняет проверку, направленную на обеспечение правильного ввода наименования поставщика в AVL;

d) когда группа закупок заказывает детали, то она сверяется с AVL для обеспечения того, что поставщик одобрен и уполномочен поставлять запрошенные детали;

e) если поставщик одобрен, группа закупок подписывает заявку с указанием проведенной проверки в системе AVL.

Анализ риска процесса

Джек рассуждает, что может пойти не так в существующем в настоящее время процессе. Если процесс нарушится, то детали могут быть заказаны у неодобренного поставщика. Такое может произойти либо потому, что неодобренный поставщик был добавлен в AVL, либо потому, что группа закупок не проверяет AVL перед заказом деталей.

Затем Джек рассматривает существующие меры по управлению риском для смягчения таких рисков. Джек видит, что группа закупок имеет процедуру проверки вручную того, что наименования поставщиков были правильно добавлены в AVL, а доступ к перечню разрешен только для авторизованных сотрудников. Кроме того, Джек видит, что текущая процедура закупки требует подписи сотрудника группы закупок о проведении проверки нахождения поставщика в AVL до выдачи заказа на покупку. Контроль, обеспечивающий размещение заказов у одобренных поставщиков, осуществляется в инспекционной группе по приемке, где AVL снова проверяется при получении деталей.

На основании анализа этих мер по управлению риском Джек устанавливает, что остаточный риск процесса является низким. Поэтому он предполагает, что новая система AVL, вероятней всего, будет системой с низким риском.

Определение предусмотренного применения

Теперь, когда Джек понимает, что бизнес-процесс должен быть автоматизирован, он составляет в письменном виде документ, содержащий формулировку цели и назначения для предлагаемой новой системы AVL:

- Система AVL автоматизирует проверку поставщиков и деталей по электронному перечню AVL с целью обеспечения того, что детали заказываются только у официально одобренных поставщиков. В новой системе будет использоваться база данных AVL, которая будет связана с существующей системой заказов на закупку. Новая система будет использоваться группой качества корпорации Acme в штаб-квартире при осуществлении процесса квалификационной проверки поставщиков, а также агентами по закупкам в процессе формирования заказов на закупку.

Джек рассматривает другие системы и процессы, с которыми система AVL будет взаимодействовать, а также добавляет некоторые пояснения к составленному им документу для уточнения границ функционирования новой системы:

- Процесс закупок будет находиться во взаимодействии с процессом, автоматизируемым системой AVL. Интерфейс будет состоять из запроса к базе данных AVL о статусе поставщика, указанного в заказе на закупку. Процесс закупок не подтверждает точность данных в AVL и не взаимодействует с процессом оценки поставщика.

Планирование валидации

Теперь, когда Джек осознает, что бизнес-процесс должен быть автоматизирован, цель и назначение новой системы установлены, то можно приступать к разработке первоначального плана валидации. Более подробно он решает изложить этот план несколько позже, однако, с целью определения масштаба необходимой деятельности и усилий по валидации, он хочет начать планирование валидации сейчас.

Ранее Джек определил остаточный риск в существующем процессе AVL как низкий. Поэтому он считает, что в целях валидации ему не нужно поддерживать высокий уровень формализации и детализации масштаба прилагаемых усилий. Он знает, что важно определить требования к бизнес-процессам пользователей и требования к программному обеспечению для новой системы. Но поскольку система имеет низкий риск, Джеку не нужно иметь отдельные документы с отдельными подписями на каждом документе. Он решает объединить требования к бизнес-процессам пользователей, требования к программному обеспечению и план проведения испытаний в один документ, используя формат таблицы.

Поскольку рассматриваемая система имеет весьма низкий риск, Джек обоснованно устанавливает, что отсутствует необходимость в проведении тщательного управленческого анализа планируемого масштаба усилий по валидации. Он решает, что официального одобрения менеджером по работе с поставщиками и представителем руководства по обеспечению качества будет вполне достаточно. Но Джек также считает, что для обеспечения уверенности в правильности определения требований пользователя он должен заручиться отзывом представителя группы закупок.

Джек начинает составлять проект своего плана валидации в соответствии с принятыми решениями. Корпорация Acme использует стандартный формат, которому должны соответствовать все планы валидации. Некоторые разделы плана валидации еще не определены, однако Джек дополнит план после одобрения первоначального проекта системы.

Определение требований к программному обеспечению

Далее Джек в письменном виде составляет требования к программному обеспечению. Он решает, что требования к программному обеспечению должны включать «что» (действия, требуемые процессом или системой AVL), спецификацию интерфейса, устанавливающего взаимодействие системы AVL с системой закупок, словарь данных и примеры допустимых запросов, которые новая система должна быть в состоянии обработать.

Определение границ взаимодействия программного обеспечения с другими системами

Затем Джек рассматривает другие системы, с которыми новая система AVL должна будет взаимодействовать. Он устанавливает, что единственное взаимодействие будет происходить с существующей системой закупок Acme, которая может запрашивать базу данных AVL с помощью простых запросов на языке SQL.

Установление деятельности по обеспечению уверенности и управлению программным обеспечением

Далее Джек должен решить, какой подход и какие технологии он должен использовать для создания новой системы. Поскольку бизнес-требования достаточно просты, то и объем транзакций будет низким. Так как система AVL является системой с низким риском, Джек решает разработать ее с помощью широко доступной и простой в использовании системы баз данных Microsoft Access. (Microsoft Access упоминается только в качестве примера подходящего и широко доступного продукта. Эта информация приведена для удобства пользователей настоящего документа и не является одобрением ИСО этого продукта.)

Поскольку Microsoft является сторонним разработчиком программного обеспечения, Джек должен решить, какие виды деятельности по установлению доверия к Microsoft Access он должен выполнить. Джек отмечает, что Microsoft Access является широко используемым инструментом и что в прошлом любые проблемы или нюансы, связанные с данным продуктом, были быстро выявлены и опубликованы на форумах в Интернете. В сочетании с тем, что система AVL является системой с низким уровнем риска, Джек решает не предусматривать проведение аудита поставщика в отношении Microsoft как разработчика базы данных.

Поскольку новая система содержит электронные записи, Джек решает внедрить стороннее программное обеспечение как «обертку» для Microsoft Access в качестве необходимого средства управления в отношении обеспечения достоверности записей.

Анализ риска отказа программного обеспечения

Хотя Джек уже установил, что автоматизируемый бизнес-процесс имеет низкий риск, ему все еще нужно проанализировать риск отказа программного обеспечения. Для этого он решает использовать количественную модель определения риска (со шкалой от 1 до 10). Ранжирование новой системы по уровням тяжести и вероятности он проводит следующим образом:

- Джек ранжирует «тяжесть» как среднюю (6), поскольку отказ программного обеспечения способен причинить вред лишь косвенно. Он обосновывает присвоенный ранг наличием средств управления на последующих этапах процесса;

- Джек ранжирует «вероятность» как низкую (1), поскольку конструкция базы данных довольно проста, что делает маловероятным необнаружение критических ошибок в процессе тестирования;

- сочетание результатов ранжирования определяет риск как низкий.

Поэтому Джек будет выполнять задачи валидации, исходя из низкого уровня риска.

Завершение плана валидации

Когда Джек определил требования к программному обеспечению, оценил риск и определился с подходом к имплементации программного обеспечения, то он теперь имеет достаточно информации для завершения плана валидации. В этот момент он делает шаг назад и в свете доступной информации о системе, подходе к имплементации

и программном риске задает себе следующий вопрос: «Какая деятельность по валидации действительно дала бы мне уверенность в том, что эта система подходит для ее предусмотренного применения?»

Поскольку система является закупленным инструментом базы данных и имеет относительно низкий риск, Джек считает, что запланированная им деятельность по валидации достаточна, однако ему необходимо учесть требования окружающей среды с целью обеспечения должного управления изменениями в операционной системе и версии Access. Он обновляет план валидации, дополняя его требованием официального управления конфигурацией программного обеспечения.

Поскольку система разрабатывается третьей стороной, Джек должен быть уверен, что разработчик правильно интерпретирует требования к настройке, входным данным, интерфейсам, хранению данных и выходным данным. Так как данная система будет зависеть от входных данных от существующих систем, то для подтверждения правильности работы разработчика Джек добавляет интерфейсный и интеграционный тесты системы в качестве важной деятельности в план валидации.

Наконец, Джек хочет убедиться, что разработчик во время разработки поддерживает надлежащее управление версиями, поэтому он добавляет управление версиями программного обеспечения в качестве необходимой деятельности в свой план валидации.

Обобщая и используя принцип критического мышления, Джек добавляет следующие инструменты к остальной части деятельности по разработке и масштабу усилий по валидации.

- Средства проектирования, разработки и конфигурации:
  - анализ документирования архитектуры программного обеспечения;
  - матрица прослеживаемости (интегрированная в спецификацию требований);
  - меры по управлению риском (документировано в пользовательской спецификации).
- Средства тестирования:
  - интеграционное тестирование (документировано в спецификации требований);
  - тестирование интерфейса (документировано в спецификации требований);
  - системное тестирование (документировано в спецификации требований).
- Средства развертывания:
  - анализ пользовательских процедур;
  - внутреннее обучение работы с приложением;
  - установочная квалификация.

Планирование поддержки

Джек заранее продумывает, какая деятельность может быть уместна для обеспечения качества программного обеспечения после развертывания системы. Учитывая низкий остаточный риск системы, Джек решает, что следует проводить ежеквартальный обзор точности данных AVL в базе данных. Джек включает в свой план валидации раздел для документирования ежеквартального обзора и составляет запрос на разработку и внедрение процедуры, обеспечивающей проведение ежеквартального обзора после запуска системы.

Пример 12: Программное обеспечение управления калибровкой

Медицинская компания XYZ быстро растет. XYZ приобретает компании в Европе и Азии. Рост компании означает, что потребности в управлении калибровкой в XYZ также растут. В настоящее время менеджер по калибровке XYZ ведет журнал учета всей информации по калибровке и еженедельно проводит анализ реестра откалиброванного оборудования с целью определения потребности в повторной калибровке какого-либо оборудования. По мере роста компании этот реестр становится слишком большим и территориально рассредоточенным, чтобы один человек мог управлять им с помощью системы отображения данных на бумажном носителе. Настало время внедрить компьютеризированную систему.

Определение процесса

XYZ имеет стандартную операционную процедуру (SOP), требующую, чтобы компьютеризированные системы, автоматизирующие часть системы качества, были валидированы для их предусмотренного применения. XYZ сначала определяет процесс управления калибровкой для выявления связанных с процессом рисков и установления того, будет ли программное решение автоматизировать весь текущий процесс или только его часть. Менеджеры XYZ анализируют стандартную операционную процедуру (SOP) управления калибровкой, которая содержит подробные сведения о следующих этапах процесса:

a) закупается новое оборудование;

b) новому оборудованию присваивается уникальный идентификационный номер (ID);

c) устанавливается процедура калибровки;

d) новое оборудование калибруется;

e) статус калибровки указывается на оборудовании посредством маркировки;

f) поддерживаются записи калибровки, включающие требования к калибровке, статус и дату истечения срока действия;

g) записи по калибровке рассматриваются с точки зрения адекватности отчетности и деятельности по управлению калибровкой.

Анализ риска процесса

Процесс управления калибровкой содержит некоторые присущие ему риски независимо от того, является ли он системой с отображением данных на бумажном или электронном носителе. Связанные с этим процессом риски заключаются в следующем:

- при использовании оборудования после истечения срока его калибровки, регистрируются некорректные измерения. Эта проблема может иметь ряд последствий в зависимости от вида оборудования и стадии процесса, в котором оно используется;

- размещенная на оборудовании некорректная маркировка может указывать на то, что оборудование откалибровано, хотя уже истек срок его калибровки. Эта ошибка также имеет ряд последствий в зависимости от вида оборудования и стадии процесса;

- записи калибровки могут быть утеряны, что может повлечь появление некоторого оборудования с истекшим сроком калибровки. Эта проблема может задержать работу;

- если статус калибровки записан некорректно, то может возникнуть ситуация использования оборудования с истекшим сроком калибровки;

- если две единицы оборудования получают один и тот же идентификационный номер, записи не будут уникальными.

Установлено, что данный процесс сопряжен с высоким риском из-за возможных последствий неправильной калибровки. В худшем случае для измерений при окончательной приемке медицинского изделия может быть использовано оборудование, не прошедшее калибровку, в результате чего ему может быть неправомерно присвоен статус соответствующего изделия.

Чтобы устранить эту проблему, менеджеры XYZ должны обновить стандартную операционную процедуру (SOP), чтобы включить инструкции для каждого пользователя оборудования. Каждый пользователь должен перед использованием проверить маркировку окончания срока калибровки на оборудовании. При составлении протоколов, в которых задействовано калиброванное оборудование, каждый пользователь должен записать в него идентификационный номер оборудования и дату его калибровки. Также пользователи должны быть обучены тому, как идентифицировать элементы, которые должны быть откалиброваны, и научены не использовать оборудование с истекшим сроком или без маркировки.

Реализация таких мер позволяет снизить остаточный риск системы до умеренного уровня. Менеджеры XYZ считают, что инструкция пользователя является подходящей, но недостаточно результативной мерой по снижению остаточного риска до минимального.

Определение предусмотренного применения программного обеспечения

Программная система не будет выполнять деятельность по калибровке; она будет представлять собой базу данных, содержащую информацию о калибровке и данные об оборудовании, истории его калибровки и текущем статусе. Программная система будет управлять этапами b), f) и g) процесса калибровки.

Менеджеры XYZ соглашаются, что XYZ будет валидировать систему для следующих целей и назначений.

- Система управления калибровкой используется для предоставления идентификационных номеров требующего калибровки оборудования, печати маркировочных ярлыков для калибруемого оборудования, хранения данных о результатах калибровки и составления отчета о статусе калибровки оборудования. Система управления калибровкой автоматизирует выполнение части требований ИСО 13485, касающихся оборудования для контроля, измерения и испытаний.

Планирование валидации

Чтобы основательно подготовиться к деятельности по валидации, руководители XYZ начинают ее планирование с установления ожидаемых результатов работы процесса калибровки и привлекают к процессу планирования межфункциональную группу. Они документируют следующие шаги:

- определение уровня строгости создаваемой документации для выбранных инструментов;
- строгость документации для системы будет умеренной. Таким образом, ключевые результаты будут формироваться и одобряться по отдельности;
- определение уровня наблюдения и одобрения (вовлечение в работу и анализ результатов команды менеджеров и межфункциональной группы) для выбранных инструментов;
- учитывая, что эта система управления калибровкой будет использоваться в глобальном масштабе, целесообразно выделить, в форме официального одобрения плана валидации и отчета о валидации системы, деятельность по менеджменту информационных технологий и управление операциями на глобальном уровне, которая должна проводиться в процессе валидации системы. Кроме того, к анализу и официальному одобрению всех документов будут привлечены менеджеры по оборудованию новых производственных участков;
- выбор «предопределенных» инструментов из набора инструментов:
  - требования к пользовательским и бизнес-процессам;
  - требования к программному обеспечению;
  - формальный анализ требований к программному обеспечению.

Определение требований к программному обеспечению

Требования к программному обеспечению будут содержать следующие элементы:

- функциональные рабочие диаграммы процесса;
- требования к электронной записи и электронной подписи;
- требования к логической схеме данных;
- требования к отчетности;
- требования к печати маркировочных этикеток для оборудования;
- требования безопасности пользователей и их профилей;
- эксплуатационные требования;
- требования к пропускной способности.

Установление деятельности по обеспечению уверенности и управлению программным обеспечением

Менеджеры XYZ проводят опрос трех поставщиков данного типа продукции и устанавливают, что продукция одного из них наилучшим образом соответствует предусмотренному применению программного обеспечения в XYZ. Выпускаемая этим поставщиком система широко используется в медицинской промышленности, хотя рассматриваемая версия продукта является относительно новой. Некоторая уверенность может быть получена из записей, связанных с предыдущей версией программного обеспечения. Анализ уже известных дефектов будет проводиться на основе новых сообщений о проблемах. Тестированию новых функций, не входивших в ранее выпущенную версию, будет уделено особое внимание со стороны группы по разработке тестов.

Определение границ взаимодействия программного обеспечения с другими системами

Данное программное обеспечение не имеет интерфейсов с другими программными системами.

Анализ риска программного обеспечения

Группа по валидации встречается с глобальными менеджерами по калибровке и совместно использует вопросник, приведенный в таблице С.16, для определения риска программного обеспечения. Сначала они идентифицируют риски, а затем определяют для них меры по управлению риском. В завершение они оценивают допустимость остаточного риска (см. таблицу С.17).

Таблица С.16 — Пример 12 — Анализ рисков

	Вопрос по идентификации рисков	Отметьте «да» или «нет». В случае отметки «да» выберите идентификатор риска (риск 1, риск 2, ... риск *n*)
1.1 Безопасность продукции (вред)	Существует ли потенциальный риск для безопасности продукции в случае отказа программного обеспечения? Да, во всех случаях. Неоткалиброванное оборудование может быть ошибочно идентифицировано как калиброванное. - Вред для пациента — Да. Несоответствующая спецификации продукция может быть применена к пациенту, если для измерений было использовано неоткалиброванное оборудование. - Вред для оператора — Да. Если измерения температуры или мощности неверны, то оператор может пострадать или получить травму. - Вред для случайного лица, находящегося рядом — Да. Конкретный вред зависит от оборудования. - Вред для обслуживающего персонала — Да. Если измерения температуры или мощности неверны, то обслуживающий персонал может пострадать или получить травму. - Вред для окружающей среды — Да. Если давление измерено неправильно и резервуар содержит экологически вредные материалы, то возможна утечка	Риск 1 — используется неоткалиброванное оборудование
1.2 Безопасность продукции (вред)	Существует ли потенциальный риск для безопасности продукции, если пользователь делает ошибку? Да, во всех случаях, если пользователь вводит неверные данные калибровки для единицы оборудования (см. 1.1). - Вред для пациента — Да. - Вред для оператора — Да. - Вред для случайного лица — Да. - Вред для обслуживающего персонала — Да. - Вред для окружающей среды — Да	См. риск 1
2.1 Качество продукции	Существует ли потенциальный риск для качества продукции (кроме риска безопасности), если программное обеспечение неисправно? Да. Продукция может не соответствовать спецификации, поскольку неоткалиброванное оборудование может быть ошибочно идентифицировано программным обеспечением как откалиброванное. Хотя неправильная идентификация не является проблемой безопасности, она может вызвать неудовлетворенность потребителя	См. риск 1
2.2 Качество продукции	Существует ли потенциальный риск для качества продукции (кроме риска безопасности), если пользователь делает ошибку? Да. Если пользователь вводит неверные данные калибровки для единицы оборудования и оборудование используется для измерения продукции, то продукт может не соответствовать спецификации. Хотя неправильная спецификация не является проблемой безопасности, она может вызвать неудовлетворенность потребителя	
3.1 Целостность записей	Существует ли потенциальный риск для целостности записи в системе, которая является хранилищем записей? Потеря записей — Да. Записи о калибровке могут быть утеряны. Повреждение записей — Да. Записи о калибровке могут быть повреждены	Риск 2 — записи о калибровке утеряны и вызывают проблему оценки соответствия. Риск 3 — записи о калибровке повреждены и вызывают проблему оценки соответствия
4.1 Демонстрация соответствия стандарту ИСО	Существует ли потенциальный риск в отношении способности продемонстрировать соответствие регулирующим требованиям? Потеря записей — Да. Данные о калибровке могут быть утеряны. Повреждение записей — Да. Данные о калибровке могут быть повреждены	См. риски 2 и 3

Таблица С.17 — Пример 12 — Оценивание и управление риском

Идентификатор риска	Описание	Тяжесть	Управление	Остаточный риск
Риск 1	Неоткалиброванное оборудование используется для измерения продукции или используется для измерения давления или мощности (риск возникает, так как программное обеспечение ошибочно идентифицирует оборудование или так как пользователь вводит неверные калибровочные данные для оборудования)	Высокая	Система предназначена для печати маркировочных этикеток, содержащих идентификационный номер оборудования, серийный номер, а также статус калибровки и срок действия. По процедуре, сотрудники обучаются проверять эту информацию перед использованием оборудования. Другой процесс требует, чтобы калибровочные данные были верифицированы вторым лицом перед их занесением в отчет по калибровке	Допустимый
Риск 2	Записи утеряны, и деятельность по управлению калибровкой не приводит к должному результату	Средняя	Группой по калибровке на бумажном носителе поддерживаются все записи в отношении калибровки	Допустимый
Риск 3	Записи повреждены, и деятельность по управлению калибровкой не приводит к должному результату	Средняя	Группой по калибровке на бумажном носителе поддерживаются все записи в отношении калибровки	Допустимый

По завершении анализа риска менеджеры XYZ удовлетворены тем, что остаточный риск после снижения является допустимым.

Завершение плана валидации

С целью завершения планирования валидации менеджеры изменяют план таким образом, чтобы он содержал следующий набор инструментов.

- Инструменты имплементации:
  - матрица прослеживаемости;
  - анализ конфигурации системы.
- Инструменты тестирования:
  - анализ величин измерения;
  - планирование испытаний;
  - поставляемый поставщиком комплект тестов с дополнительным тестированием для запланированной конфигурации и нового функционала, не входившего в предыдущую версию программного обеспечения.
- Средства развертывания (внедрения):
  - внутреннее обучение по применению программного обеспечения;
  - установочная квалификация (для сервера и рабочих станций).

Планирование обслуживания

В дополнение к планированию валидации системы менеджеры XYZ решают, что планирование обслуживания системы будет полезно, поскольку оно обязательно потребуется в будущем.

Будут внедрены методы мониторинга системы для анализа всех дефектов, анализа проблем использования системой и анализа изменений в предусмотренном применении.

С целью более эффективной реализации изменений будет разработан план классификации изменений в системе (т. е. аппаратное обеспечение, обновления, исправления, проблемы безопасности).

Пример 13: Система автоматизированного зрения

Инженеры в компании Гэри очень хороши в своей работе. Они знают, что продукция, которая производится в зоне автоматизации Гэри — металлический стержень, длина которого варьируется от 1,27 см до 3,81 см. Эти стержни имеют два применения: одно для размеров длиной 2,54 см или меньше, а другое для длины в 3,18 см (± 0,64 см). Все стержни имеют ширину 0,32 см. Оба применения предназначены для медицинских изделий и требуют, чтобы стержень был определенной длины. Гэри поручил инженеру по автоматизации проверить автоматизированную систему зрения, которая сортирует детали. Эта система заменяет ручной процесс измерения/сортировки. Других изменений в процессе нет, поэтому это полная область применения предстоящей валидации системы.

Описание процесса

Спецификация толщины стержня одинакова для обоих применений и отслеживается в сырье, используемом автоматом для резки стержней. Все критерии приемки подтверждаются на предыдущих производственных процессах, за исключением длины стержня, измеряемой автоматизированной системой зрения, за которую и отвечает Гэри.

Машинный процесс довольно прост. Стержни загружаются в приемник, который использует воронку для размещения стержней на конвейерной ленте, по одному за раз. Каждый стержень транспортируется к месту остановки, где камера смотрит на стержень и измеряет его длину. В зависимости от результата, стержни перемещаются в два контейнера: один для стержней длиной 2,54 см или меньше и другой для более длинных стержней.

Далее в процессе нет никаких дополнительных проверок длины стержней. Существует повышенный риск причинения вреда пациенту, если используется стержень неправильного размера, потому что такой стержень может привести к утечкам в изготавливаемых изделиях. Далее в процессе не предусмотрено никакого метода проверки для снижения этого повышенного риска. Если стержень правильной длины и в пределах установленных размеров, то утечки в изделии не произойдет. Изделия изготавливаются в течение многих лет, и рассматриваемый риск хорошо понятен. Автоматизированная система зрения заменяет ручной процесс измерения.

Определение предусмотренного применения

Поскольку Гэри хорошо разбирается в подлежащем автоматизации процессе, он начинает с определения его цели и назначения.

- Программное обеспечение предназначено для подтверждения факта нахождения на конвейере только одного металлического прута, а также для измерения его длины.

Анализ риска

Гэри использует внутренний процесс анализа риска и устанавливает, что риск последствий отказа для системы высокий, потому что нет никакого способа обнаружить несоответствующий размер стержня, кроме как через последующий отказ продукции или посредством разрушающих испытаний. Последствия отказа могут причинить вред пациенту. Критическим параметром для процесса является точный размер длины стержней. Автоматизация не увеличивает и не уменьшает этот риск.

Планирование валидации

При разработке первой итерации плана валидации Гэри предполагает использовать строгий подход к процессу ее проведения (анализ риска определил риск на высоком уровне). После изучения потенциально подходящих инструментов валидации из набора инструментов он планирует формальный документ по определению требований и проведение анализа требований к программному обеспечению. В этот анализ требований к программному обеспечению будут вовлечены инженер-технолог, инженер по автоматизации и инженер по качеству.

Программное обеспечение для этой системы будет разработано собственными силами, но, учитывая предыдущую автоматизацию системы, предстоящая разработка будет относительно простой.

Меры по управлению риском

Идентифицированы две зоны рисков:

a) требуется подтверждение, что только один стержень находится в месте замера. Машина пропускает стержни через узкое отверстие, измеряя 0,64 см в ширину и 0,48 см в высоту. Исходя из этого, стержни могут попасть на конвейер только располагаясь вдоль и не пройдут, если один из стержней находится поверх другого, из-за размера отверстия. Однако две детали могут располагаться на конвейере рядом.

Чтобы уменьшить этот риск, программное обеспечение будет проверять ширину каждого стержня перед проверкой длины. Если ширина бруска превышает 0,32 см (± 0,08 см в соответствии с ранее проверенной спецификацией), то стержень отклоняется, поскольку на конвейере находятся два стержня. Для этой цели к конструкции машины добавляется третий контейнер (для брака);

b) стержни могут располагаться слишком близко друг к другу, затрудняя определение того, где именно заканчивается один стержень и начинается другой. Программное обеспечение будет направлять стержни в контейнер для брака, если их длина более 3,81 см.

Задачи валидации

Затем Гэри переходит к задачам валидации. Он определяет необходимость в составлении официальной проектной документации и планирует официальную инспекцию каждого раздела проекта с участием тех же членов команды, которые анализировали требования. Кроме того, как только код будет сгенерирован, он будет рассмотрен с точки зрения соответствия проектной документации другим инженером по автоматизации и инженером-технологом, которые имеют опыт разработки программного обеспечения. Деятельность по управлению поставщиками не выбирается, поскольку программное обеспечение разрабатывается внутри компании. Инженеру по автоматизации, инженеру-технологу и инженеру по качеству будет поручено провести анализ прослеживаемости программного обеспечения и проектной документации до требований. С целью обеспечения охвата всех требований они будут анализировать прослеживаемость только после проведения тестирования.

Выбор Гэри касательно набора инструментов раздела «Испытания (тестирование)» включает планирование проведения испытаний. Планы испытаний должны содержать сведения о программной среде применения и ожидаемых результатах тестирования. Гэри планирует несколько типов тестирования на различных этапах разработки, включая модульное тестирование, интеграционное тестирование и системное тестирование. Будут использованы позитивные и негативные тестовые сценарии, а также эксплуатационное тестирование, связанное со скоростью конвейерной ленты. Помимо Гэри, планы испытаний должны быть проанализированы и одобрены другим инженером по автоматизации, инженером-технологом, инженером по качеству. Отчет по результатам тестирования включает в себя: фактические результаты тестирования; их сравнение с ожидаемыми результатами тестирования; критерии соответствия или несоответствия результатов теста; идентификатор теста (его уникальный номер); документацию по разрешению проблем и регрессионному тестированию для любых отказов. Гэри необходимо официальное одобрение отчета по результатам тестирования от той же группы сотрудников.

Имплементация, тестирование и развертывание

Для развертывания системы автоматизированного зрения Гэри анализирует инструменты развертывания из набора инструментов и решает, что потребуется установочная квалификация. Кроме того, он устанавливает необходимость создания пользовательской процедуры и что для пользователей системы потребуется аттестация.

Обслуживание

Отдел Гэри коллективно планирует обслуживание всех систем в производственном подразделении. Никакого специального планирования действий в этой области не требуется.

Пример 14: Система захвата и размещения

Корпорация Hi-Quality Medical (далее — Hi-Quality) является изготовителем медицинских изделий класса В. Hi-Quality хочет автоматизировать размещение частично готовых деталей с одной станции в картриджи, которые являются частью медицинского изделия, производимого компанией.

Джилл, которая является менеджером проекта для новой системы захвата и размещения (ЗР), устанавливает, что процесс ЗР является процессом системы качества медицинского изделия в соответствии с ИСО 13485, поскольку он является частью производства медицинского изделия. Поэтому предлагаемая система ЗР будет подпадать под требования к валидации программного обеспечения.

Определение существующего процесса

Чтобы лучше понять требования и риски, связанные с разработкой системы ЗР, Джилл определяет соответствующий бизнес-процесс следующим образом:

a) детали, поступающие с участка сборки 11 в процессе производства, помещаются в картриджи для участка сборки 12 (из расчета 20 деталей на картридж). В настоящее время эта операция выполняется вручную оператором;

b) затем оператор вручную помещает картриджи на входную линию участка сборки 12;

c) оператор собственноручно проверяет правильность укладки деталей в картридж. [На выполнение шагов b) и c) для каждого картриджа требуется около 3 мин];

d) картридж переходит на другие участки сборки, которые включают визуальный осмотр, подтверждающий отсутствие деформаций на всех предыдущих этапах процесса.

Анализ риска процесса

Затем Джилл размышляет, что может пойти не так в текущем процессе. Ее анализ показывает, что могут произойти следующие события:

a) оператор может деформировать частично законченную деталь. Деформация будет обнаружена на дальнейших этапах инспекционной станцией;

b) оператор может неправильно поместить деталь в картридж или пропустить слот в картридже. Неправильное размещение или отсутствие гнезда в настоящее время обнаруживается на станции 12 во время ручного осмотра.

Учитывая эти меры по управлению риском, Джилл устанавливает, что остаточный риск процесса является низким. Она предполагает, что новая система ЗР также будет системой с низким уровнем риска.

Определение нового процесса

Оценив риск процесса и используя полученные знания о системе ЗР, Джилл определяет новый процесс следующим образом:

a) в систему ЗР будут загружаться картриджи;

b) система ЗР будет брать детали с участка сборки 11 и помещать их в картриджи (из расчета 20 деталей на картридж);

c) система ЗР будет визуально контролировать картридж для обеспечения правильности укладки и заполнения всех слотов картриджа. Любые несоответствующие картриджи будут автоматически отклонены;

d) система ЗР поместит прошедший приемку картридж на участок сборки 12. [Шаги b) — d) теперь займут 1 мин];

e) картридж направится на другие участки сборки, которые включают визуальный осмотр, подтверждающий отсутствие деформаций на всех предыдущих этапах процесса.

Определение предусмотренного применения программного обеспечения

Теперь Джилл лучше понимает причины автоматизации процесса и готова сформулировать в письменном виде цель и назначение предлагаемой новой системы ЗР:

- Система ЗР будет брать детали с участка сборки 11 и помещать их в картриджи. Она будет подтверждать правильность укладки и заполнения всех слотов картриджа, отклонять все несоответствующие картриджи и затем перемещать картридж на входную линию участка сборки 12 со скоростью один картридж в минуту.

Затем Джилл рассматривает возможное взаимодействие системы ЗР с другими системами. Она приходит к выводу, что других взаимодействий нет. Она устанавливает наличие взаимодействия с пользователями и отсутствие программных интерфейсов.

Планирование валидации

Проанализировав автоматизируемый бизнес-процесс и установив цели и назначение новой системы, Джилл готова разработать первый проект плана валидации. Позже ей нужно будет добавить больше деталей, однако, начав планирование валидации сейчас, она сможет определить уровень необходимых усилий по валидации.

Ранее Джилл установила низкий остаточный риск существующего процесса. Исходя из этого, она считает нецелесообразным установление высокого уровня детализации и формализации в процессе валидации. Джилл знает, что важно определить требования к бизнес-процессам пользователей и требования к программному обеспечению для новой системы. Однако она отмечает, что это система низкого риска и нет необходимости создавать

отдельные документы с отдельными подписями на каждом документе. Поэтому Джилл решает объединить требования к бизнес-процессам пользователей, требования к программному обеспечению и план проведения испытаний в один документ.

Поскольку в новой системе такой низкий риск, Джилл решает, что тщательный анализ со стороны руководства по валидации не требуется и что будет достаточно одобрения со стороны руководителя производства и представителя службы по обеспечению качества. Однако с целью обеспечения правильности определения требований пользователя она добавляет проведение анализа оператором процесса.

Джилл начинает составление своего проекта плана валидации на основе стандартного формата плана валидации, установленного в корпорации Hi-Quality. Некоторые разделы плана валидации все еще не заполнены; Джилл заполнит их после официального одобрения первоначального проекта системы.

Определение требований к системе и программному обеспечению

Джилл обращается к рассмотрению требований к системе и требований к программному обеспечению. Она решает, что требования к программному обеспечению будут включать этапы процесса ЗР (или системные шаги) вместе с документированием интерфейса взаимодействия системы ЗР с участками сборки 11 и 12. Требования к системе включают скорость и точность движений системы ЗР. Для уменьшения риска причинения вреда Джилл добавляет требование безопасности по обеспечению физического барьера между оператором и захватывающим манипулятором ЗР.

Установление деятельности по обеспечению уверенности и управлению программным обеспечением

Теперь Джилл должна определиться с подходом и технологией, которые она должна использовать для закупки новой системы. Учитывая простоту бизнес-требований, цена будет низкой. Поскольку новая система имеет низкий риск, Джилл решает закупить систему ЗР у внешнего поставщика. Исходя из соотношения цены и качества, она решает закупить систему ЗР у Controlsys Inc., компании-лидера в области систем ЗР.

Controlsys является внешним поставщиком системы. Следовательно, теперь Джилл должна решить, какие виды деятельности она должна предпринять для достижения уверенности в Controlsys. Она оценивает имеющуюся у нее информацию об этой компании. Джилл отмечает, что Controlsys предлагает широко используемый продукт ЗР с отличной репутацией. Имевшиеся ранее проблемы и нюансы данного продукта были быстро выявлены самой компанией и опубликованы на форумах в Интернете. Обзор этой информации показывает, что существует только несколько известных незначительных проблем. Джилл проводит анализ и подтверждает, что эти проблемы не связаны со спецификой сформулированного ей предусмотренного применения программного обеспечения. Кроме того, Controlsys предлагает автоматизированный набор тестов для установочной, операционной и эксплуатационной квалификации (IQ/OQ/PQ). Учитывая историю компании и низкий уровень риска системы ЗР, Джилл решает не проводить аудит в компании Controlsys. Она официально одобряет компанию Controlsys как вендор-поставщика.

Анализ рисков отказа программного обеспечения

Несмотря на уже установленный Джилл низкий риск автоматизируемого бизнес-процесса, ей все еще необходимо провести анализ риска отказа программного обеспечения. Она решает использовать количественную модель определения риска и проводит ранжирование новой системы по уровням тяжести и вероятности следующим образом:

- Джилл ранжирует «тяжесть» как низкую (3) по шкале от 1 до 10, поскольку последствия отказа программного обеспечения будут обнаружены в последующей деятельности;

- она ранжирует «вероятность» как низкую (1), поскольку конструкция системы довольно проста, что делает маловероятным необнаружение критических ошибок в процессе тестирования;

- она определяет значение риска. Полученное значение равно 4, что классифицирует риск как низкий.

Поэтому Джилл будет выполнять задачи валидации, исходя из низкого уровня риска.

Завершение плана валидации

Поскольку Джилл уже завершила определение требований к программному обеспечению, оценила риск и определилась с подходом к исполнению программного обеспечения, то она теперь имеет достаточно информации для завершения плана валидации.

Так как предлагаемая система имеет низкий остаточный риск, Джилл выбирает следующие инструменты для оставшейся части усилий по разработке и валидации.

- Инструменты проектирования, разработки и компоновки (настройки конфигурации):
  - анализ документирования архитектуры программного обеспечения;
  - матрица прослеживаемости (интегрированная в спецификацию требований);
  - меры по управлению риском будут документированы в пользовательской спецификации.
- Инструменты испытаний (тестирования):
  - интеграционное тестирование (документировано в спецификации требований);
  - тестирование интерфейса (документировано в спецификации требований);
  - системное тестирование программного обеспечения (документировано в спецификации требований).

- Инструменты развертывания:
  - анализ пользовательских процедур;
  - внутреннее обучение операторов;
  - автоматизированный набор тестов, поставляемый поставщиком (от Controlsys).

Планирование обслуживания

Теперь Джилл рассматривает уместную деятельность по обеспечению качества функционирования системы после ее развертывания. Из-за низкого остаточного риска системы она следует рекомендациям изготовителя и добавляет калибровку механизма движения в график калибровки. Джилл устанавливает для этой системы самый продолжительный валидационный цикл, принятый в компании (3 года).

Анализ выполненных действий на основе критического мышления

Наконец, Джилл спрашивает себя, рассмотрела ли она все элементы, необходимые для достижения уверенности в своем подходе к валидации. Она приходит к выводу, что выбранная и завершенная деятельность по валидации обеспечивает допустимый уровень уверенности в том, что программное обеспечение будет работать должным образом.

Библиография

[1] ISO 9000 Quality management systems — Fundamentals and vocabulary (Системы менеджмента качества. Основные положения и словарь)

[2] ISO 12207 Systems and software engineering — Software life cycle processes (Информационная технология. Системная и программная инженерия. Процессы жизненного цикла программных средств)

[3] ISO 13485:2016 Medical devices — Quality management system — Requirements for regulatory purposes (Изделия медицинские. Системы менеджмента качества. Требования для целей регулирования)

[4] ISO 14971 Medical devices — Application of risk management to medical devices (Изделия медицинские. Применение менеджмента риска к медицинским изделиям)

[5] ISO/IEC Guide 51 Safety aspects — Guidelines for their inclusion in standards (Аспекты безопасности. Руководящие указания по включению их в стандарты)

[6] IEC 62304:2006/AMD1:2015 Medical device software — Software life cycle processes — Amendment 1 (Изделия медицинские. Программное обеспечение. Процессы жизненного цикла)

[7] IEC/TR 80002-1 Medical device software — Part 1: Guidance on the application of ISO 14971 to medical device software (Программное обеспечение медицинских изделий. Часть 1. Руководство по применению ИСО 14971 к программному обеспечению медицинских изделий)

[8] National Institute of Standards and Technology (NIST) Special Publication 500-234, *Reference Information for the Software Verification and Validation Process*, Dolores R. Wallace, Laura M. Ippolito, Barbara Cuthill, March 19. 1996

[9] Software Engineering Institute. Capability Maturity Model Integration (CMMI)

[10] Pressman R., 1992. Software Engineering, A Practitioner's Approach. McGraw-Hill, Inc, Third Edition

[11] GHTF/SG1/N77, 2012 Principles of I Devices Classification (Принципы классификации медицинских изделий)

УДК 006.85:006.354	ОКС 11.040.01 35.240.80	Р20	IDT

Ключевые слова: программное обеспечение, менеджмент риска, система менеджмента качества, изготовитель, медицинское изделие, валидация

БЗ 10—2020

Редактор *Е.А. Моисеева*
Технические редакторы *В.Н. Прусакова, И.Е. Черепкова*
Корректор *Е.М. Поляченко*
Компьютерная верстка *Д.В. Кардановской*

Сдано в набор 28.08.2020. Подписано в печать 25.09.2020. Формат 60 × 84 $^1/_8$. Гарнитура Ариал.
Усл. печ. л. 9,30. Уч.-изд. л. 8,75.

Подготовлено на основе электронной версии, предоставленной разработчиком стандарта

ИД «Юриспруденция», 115419, Москва, ул. Орджоникидзе, 11
www.junsizdat.ru gost.e-ecolog.ru

Создано в единичном исполнении во ФГУП «СТАНДАРТИНФОРМ»
для комплектования Федерального информационного фонда стандартов,
117418 Москва, Нахимовский пр-т, д. 31, к. 2.
gost.e-ecolog.ru gost.e-ecolog.ru

gost.e-ecolog.ru