



П Р И К А З

об утверждении технического регламента «Процессы жизненного цикла программного обеспечения» RT 38370656 - 002:2006

№ 78 от 01.06.2006

Мониторул Официал N 95-97/335 от 23.06.2006

* * *

Во исполнение Постановления Правительства № 873 от 30.07.2004 г. «Об утверждении Национальной программы по разработке технических регламентов» ПРИКАЗЫВАЮ:

1. Утвердить технический регламент «Процессы жизненного цикла программного обеспечения» RT 38370656 - 002:2006 (прилагается).

2. Настоящий технический регламент вступает в силу по истечении 30 дней со дня опубликования в «Официальном мониторе Республики Молдова».

Министр информационного развития

Владимир МОЛОЖЕН

Кишинэу, 1 июня 2006 г.
№ 78.

Приложение
к приказу министра
информационного развития
№ 78 от 1 июня 2006 г.

ТЕХНИЧЕСКИЙ РЕГЛАМЕНТ
«Процессы жизненного цикла программного обеспечения»
RT 38370656 - 002:2006

1. ОБЪЕКТ И ОБЛАСТЬ ПРИМЕНЕНИЯ

Настоящий технический регламент распространяется на процессы жизненного цикла программного обеспечения и обязателен при создании всех автоматизированных информационных систем государственного значения.

Технический регламент ориентирован на применение объектно-ориентированных технологий и современных средств управления проектами, разработки и поддержки версииности.

Экономические агенты, поставляющие/приобретающие программное обеспечение для негосударственных автоматизированных информационных систем могут применять настоящий технический регламент в добровольном порядке.

2. ТЕРМИНОЛОГИЯ И УСЛОВНЫЕ ОБОЗНАЧЕНИЯ

2.1. Используемые в настоящем техническом регламенте понятия имеют следующие значения:

Инкрементный подход (в контексте разработки программной системы): Процесс непрерывного развития архитектуры системы, когда каждая новая версия содержит улучшения по сравнению с предыдущей.

Приобретатель: Организация, которая приобретает или получает программный продукт или услуги по программному обеспечению от поставщика.

Деятельность: Протяженное во времени выполнение какого-либо действия некоторой ролью.

ПРИМЕЧАНИЕ - Деятельность выполняется одной или несколькими функциями системы, описанных с точки зрения взаимодействия бизнес-работников, согласно логической структуре предприятия.

Бизнес-актер: Объект, работающий за пределами системы, но взаимодействующий с ней.

Архитектура: Совокупность существенных решений по организации программной системы, а также набор структурных элементов и интерфейсов, из которых она состоит, вместе с поведением, описываемым в терминах коопераций этих элементов.

ПРИМЕЧАНИЕ - К архитектуре программной системы относятся также использование, функциональность, производительность, гибкость, повторное применение, ясность, экономические и технологические ограничения и компромиссы, а также эстетические аспекты.

Артефакт: Элемент информации, используемый или порождаемый в процессе разработки программной системы или эксплуатации программного продукта.

База данных: Совокупность связанных данных, организованных по определенным правилам, предусматривающим общие принципы описания, хранения и обработки данных.

Требование: Желаемые функциональность, свойство или поведение объекта (системы).

Комментарий: Аннотация, присоединенная к элементу или множеству элементов.

Поведение: Наблюдаемый эффект события, в том числе его результаты.

Предметная область: Специфическая область знаний или деятельности, характеризующаяся специфическими концепциями и терминами.

Разработчик: Организация, которая осуществляет действия по разработке (включая анализ требований, проектирование, реализацию программного продукта вплоть до квалификационного тестирования) в течение процесса жизненного цикла программного обеспечения.

Этап: Перечень процессов, которые необходимо выполнить за промежуток времени между двумя опорными точками в жизненном цикле программной системы, в течение которого должны быть достигнуты заранее поставленные четко определенные цели, артефакты доведены до готовности и принято решение об окончании данного этапа или переходе к следующему.

Бизнес-функция: Набор действий при выполнении бизнес-процесса, которые дают полезный результат конкретному бизнес-актеру.

ПРИМЕЧАНИЕ - Бизнес-функция содержит все основные и альтернативные сценарии работы для получения конечного результата.

Поставщик: Организация, которая поставяет приобретателю программный продукт или услуги по программному обеспечению в соответствии с условиями контракта или распорядительного документа.

Идентификатор объекта: Атрибут данных, значение которого однозначно определяет информационный объект.

Инфраструктура процесса сопровождения: Организационная структура, обязанности, процедуры, процессы и ресурсы, используемые при выполнении сопровождения.

Итерация: Четко очерченный перечень работ, для которых определены конечная цель и критерий оценки. В результате нескольких итераций должна быть выпущена версия программной системы.

Адаптивное сопровождение: Модификация программного продукта, обеспечивающая его работоспособность в измененных или изменяющихся условиях (среде).

Предупреждающее сопровождение: Модификация программного продукта после поставки в целях обнаружения и корректировки имеющихся в нем скрытых ошибок для предотвращения явного проявления этих ошибок при эксплуатации данного продукта.

Корректирующее сопровождение: Реактивное изменение программного продукта, выполняемое после его поставки для корректировки обнаруженных проблем (несоответствий, ошибок).

ПРИМЕЧАНИЕ - Подобные изменения исправляют программный продукт для того, чтобы он соответствовал установленным требованиям.

Полное сопровождение: Модификация программного продукта после поставки для повышения его рабочих характеристик, добавления функциональности или улучшения сопровождаемости.

ПРИМЕЧАНИЕ - Полное сопровождение обеспечивает модернизацию (усовершенствование) программного продукта в интересах пользователя, уточнение соответствующих программных документов и его перепрограммирование для улучшения рабочих характеристик и других атрибутов программного продукта.

Сообщение о проблеме (СП): Термин, используемый для определения и описания проблем, обнаруженных в программном продукте на этапе сопровождения.

Итерационный метод (в контексте разработки программной системы): Метод разработки программной системы, при котором осуществляется процесс управления потоком исполняемых версий.

Модель жизненного цикла: Структура, включающая процессы, действия и задачи, вовлеченные в разработку, эксплуатацию и сопровождение программного продукта, и охватывающая всю продолжительность жизни программной системы от определения ее технических требований и до окончания использования.

Объект: Виртуальное отражение реально существующих сущностей, как материальных, так и нематериальных, в которых инкапсулированы состояние и поведение.

Сопроводитель: Организация, осуществляющая деятельность по поддержке (сопровождению).

Бизнес-процесс: Заданная последовательность событий, осуществляемая группой логически связанных деятельностей, использующих ресурсы организации для получения результата при выполнении целей организации.

ПРИМЕЧАНИЕ - Бизнес-процесс описывается с помощью бизнес-функций, которые показывают ожидаемое поведение системы.

Программный продукт: Программная система, имеющая коммерческую ценность, готовая к использованию или уже эксплуатируемая.

Проект: Уникальный процесс, состоящий из набора координируемых и управляемых видов деятельности с датами начала и окончания, предпринимаемых для достижения цели в соответствии с оговоренными требованиями, включая ограничения по срокам, затратам и ресурсам.

Предложение о модификации (ПМ): Общий термин, используемый для определения предполагаемых изменений в сопровождаемом продукте.

ПРИМЕЧАНИЕ - Конкретное предложение о модификации может быть далее классифицировано как коррекция или модернизация и определено как корректирующий, предупреждающий, адаптивный или полный тип сопровождения.

Бизнес-правило: Описание некоторого правила или условия, которое должно быть выполнено в пределах системы.

Реорганизация: Методы, которые позволяют уменьшить краткосрочные трудности, связанные с перепроектированием, при этом не изменяется функциональность программного обеспечения, а меняется его структура, которая становится более понятной и модифицируемой.

Риски: Внутреннее или внешнее воздействие на систему, которое может неблагоприятно повлиять на область проекта или привести к его провалу.

Роль (в контексте жизненного цикла программной системы): Определенное поведение и обязанности личности или индивидуумов, работающих в команде (рабочей группе).

Бизнес-роль (в контексте описания бизнес-процессов): Поименованное специфическое поведение сущности в конкретной ситуации.

ПРИМЕЧАНИЕ - Бизнес-роль может быть статической (например, как один из участников ассоциативной связи) или динамической (например, роль, сотрудничающая с кем-либо).

Задача: Готовый для интеграции в общую систему программный элемент, реализуемый в минимальные сроки.

Сценарий: Представление знаний, использующее заданную последовательность событий для определения результатов взаимодействия между известными элементами.

Системная услуга: Услуга, предоставляемая пользователю программной системой или программным продуктом.

Услуга: Выполнение действий, направленных на удовлетворение нужд клиента или предусмотренных нормативными документами, результат которых отражается в информационной системе и может выражаться в материальной или нематериальной форме.

Автоматизированная информационная система (АИС): Совокупность программных и аппаратных средств, предназначенных для обработки информации, информационных ресурсов и инфраструктуры пользователя.

Автоматизированные информационные системы государственного значения: Автоматизированные информационные системы, функционирующие в органах публичной власти и управления.

Программная система: Множество элементов программного обеспечения, организованных для достижения конкретной цели, иногда разложенных на несколько подсистем и описываемых набором моделей, возможно, с различных точек зрения.

Программное обеспечение: Совокупность программ системы обработки информации и программных документов, необходимых для эксплуатации этих программ.

Состояние: Ситуация в жизненном цикле объекта, во время которой он удовлетворяет некоторому условию, выполняет определенную деятельность или ожидает какого-либо события.

Стереотип: Расширение языка UML (Unified Modeling Language), позволяющее создавать новые виды строительных блоков, производные от существующих, но специфичные для описания конкретной задачи.

ПРИМЕЧАНИЕ - Стереотипы бывают текстовые и графические.

Квалификационное тестирование: Тестирование, проводимое разработчиком и засвидетельствованное заказчиком в целях демонстрации соответствия программного продукта утвержденному техническому заданию и готовности его использования в рабочих условиях.

Пользователь: Отдельное лицо или организация, которое эксплуатирует программный продукт, использует услуги программного обеспечения.

Версия: Относительно полный и самосогласованный набор артефактов, предназначенный для внутреннего или внешнего использования.

2.2. Условные обозначения

В настоящем техническом регламенте применены следующие условные обозначения:



 - роль (в контексте жизненного цикла программной системы);



 - деятельность, выполняемая ролью при реализации конкретных процессов.

3. СТОРОНЫ, УЧАСТВУЮЩИЕ В ЖИЗНЕННОМ ЦИКЛЕ ПРОГРАММНОЙ СИСТЕМЫ

В данном техническом регламенте термины «организация» и «сторона» синонимичны. Когда организация или ее часть участвует в каком-либо процессе жизненного цикла программной системы, то она становится «стороной». Стороны могут быть из одной и той же организации или из разных организаций.

Организация или сторона получает свое наименование в зависимости от процесса, который она осуществляет: например, она называется «разработчик», когда реализует процесс разработки.

Основными сторонами, которые принимают участие в процессах жизненного цикла программной системы, являются те, которые иницируют или исполняют развитие, использование или сопровождение программных продуктов.

Основными сторонами являются:

- a) приобретатель (заказчик);
- b) поставщик;
- c) разработчик;
- d) сопроводитель;
- e) пользователь.

ПРИМЕЧАНИЕ - Допускается объединение в одном лице нескольких сторон, например: заказчик и пользователь, поставщик и разработчик.

В случае, если пользователь не является заказчиком, то он может участвовать в равной степени с заказчиком на всех этапах жизненного цикла программной системы, в частности, в рамках, оговоренных в договоре (контракте) с заказчиком.

За действия и задачи в рамках основного процесса жизненного цикла программной системы отвечает сторона (или стороны), начавшая и осуществляющая данный этап.

4. ЭЛЕМЕНТЫ ЖИЗНЕННОГО ЦИКЛА ПРОГРАММНОЙ СИСТЕМЫ

4.1. Проекты

Для того чтобы выполнить над программной системой в течение ее жизненного цикла необходимые действия/преобразования, организация создает и контролирует проекты.

Проекты имеют определенные границы, сроки и цель. Границы могут включать в себя действия как на всех этапах жизненного цикла программной системы или подмножестве этапов, так в одном или нескольких определенных процессах или некоторых деятельности в рамках процесса. Временной диапазон может меняться по продолжительности.

Цель проекта связана с целью программной системы в целом или ее составных частей.

Любая программная система может разделяться на более мелкие системы и одновременно быть частью другой, более крупной программной системы.

Таким образом, взаимодействие программных систем приобретает иерархическую структуру, как показано на рисунке 1.

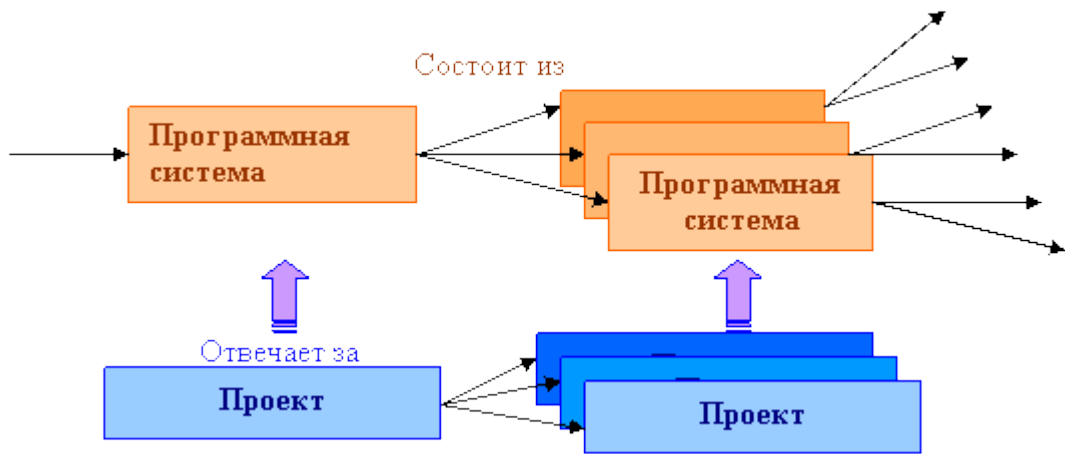


Рисунок 1. Иерархическая структура программных систем и проектов

Настоящий технический регламент применяется на всех уровнях иерархии, относящихся к программной системе. Поскольку программная система может рекурсивно разделяться на системные элементы, то процессы настоящего технического регламента могут быть повторно применены к каждому системному элементу, то есть каждый системный элемент имеет свой собственный жизненный цикл.

Каждая программная система в иерархической структуре может быть предметом ответственности отдельного проекта. Таким образом, существует тесная связь между уровнями детализации в архитектуре программных систем и уровнями ответственности в иерархии проектов. Каждый проект отвечает за получение и использование уровней в составе программной системы ниже него и за создание и поставку на уровень выше него.

4.2. Этапы

Программные системы, рассматриваемые в настоящем техническом регламенте, являются искусственными, созданными и используемыми для предоставления услуг в определенных условиях для выгоды пользователей и других заинтересованных лиц.

Жизненный цикл программной системы состоит из ряда этапов, на которых программная система планируется, создается, внедряется, эксплуатируется, сопровождается и списывается. Это иллюстрируется типовой моделью жизненного цикла программной системы, приведенной на рисунке 2.

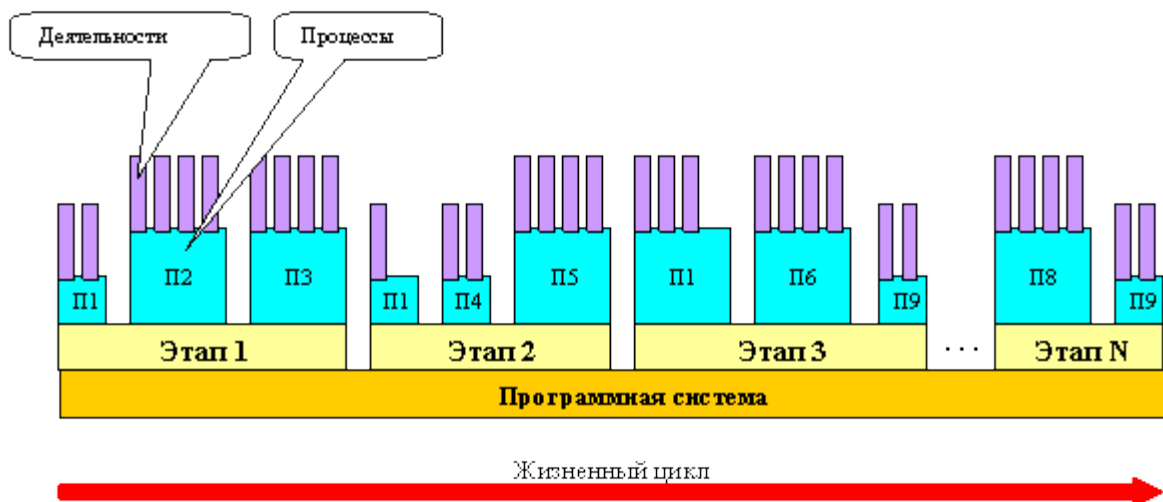


Рисунок 2. Модель жизненного цикла программной системы

Этапы жизненного цикла объединяются в последовательности, которые могут накладываться друг на друга и/или повторяться в соответствии с границами, величиной, сложностью, меняющимися потребностями и возможностями воплощения программной системы. Они являются итерационными и реализуются пошаговыми действиями.

Типичная модель жизненного цикла программной системы состоит из шести этапов:

- a) предпроектные работы;
- b) разработка программной системы;
- c) внедрение программной системы;
- d) эксплуатация программной системы;
- e) сопровождение программной системы;
- f) утилизация программной системы.

Используя указанные этапы и процессы, поставщик совместно с заказчиком должны создать модель жизненного цикла конкретной программной системы.

Каждым этапом достигается отчетливая цель и вносится вклад в жизненный цикл программной системы в целом. Каждый этап учитывается при планировании и выполнении жизненного цикла программной системы.

Для реализации каждого этапа организация должна располагать соответствующей инфраструктурой, бюджетом, программно-техническим комплексом, инструментами, утвержденными процедурами и компетентными людскими ресурсами. Все указанные компоненты планируются до начала реализации этапа, а создаются и/или приобретаются по мере необходимости.

Начало и конец каждого этапа и процесса должны быть документально оформлены в установленном в организации порядке.

4.2.1. Этап предпроектных работ

Этап предпроектных работ начинается с первоначального осознания потребности создания новой или модификации существующей автоматизируемой информационной системы (АИС) (группы АИС) или путей автоматизации деятельности и процессов. Данный этап является периодом начального изучения, сбора фактов и анализа, изучения действующего законодательства и существующей организационной структуры, обзора аналогичных существующих АИС. Для разработки выходных документов данного этапа разработчику требуется обратная связь с пользователями.

Посредством анализа, определения осуществимости, оценки затрат, маркетинга, интеллектуальных способностей и материально-технического снабжения, изучения альтернатив, а также экспериментальной разработки, вырабатываются одно или несколько альтернативных решений для удовлетворения потребности заказчика или реализации концепции. Также на этом этапе определяется потребность в одной или нескольких подсистемах для реализации программной системы.

Выходными элементами этапа предпроектных работ являются следующие основные артефакты:

- концепция системы;
- бизнес-предложение.

На этом этапе принимается решение: продолжать реализацию программной системы или прекратить дальнейшую работу.

4.2.2. Этап разработки

Этап разработки начинается с достаточно подробного уточнения требований заказчика и проектного решения и преобразует их в один или несколько жизнеспособных программных продуктов, которые обеспечивают работу в течение этапа эксплуатации. На этом этапе программная система может быть прототипом.

Оговариваются, анализируются, разрабатываются, создаются, испытываются и, при необходимости, оцениваются механизмы взаимодействия программных элементов и организационных структур, а также определяются требования к аппаратным средствам, инфраструктуре пользователя, обучению и сопровождению.

Результатом выполнения этапа являются разработка программного продукта, работоспособность и функциональность которого подтверждена квалификационными испытаниями, необходимой документацией, а также проведение других необходимых разработок.

На этом этапе посредством привлечения всех заинтересованных сторон обеспечивается включение в проект аспектов будущих этапов, а также требований и возможностей соответствующих систем обеспечения. При этом требуется обратная связь с заинтересованными лицами, которые отвечают за реализацию последующих этапов.

Выходными элементами этапа разработки являются следующие основные артефакты:

- техническое задание;
- технический проект;
- версия программной системы;
- техническая документация;
- протокол квалификационного тестирования;
- акт передачи в опытную эксплуатацию.

В случае объявления тендера на разработку программной системы техническое задание должно являться составной частью тендерного предложения.

Планирование этапа разработки начинается на предыдущем этапе, чтобы обеспечить наличие или возможность создания в организации инфраструктуры систем обеспечения разработки, состоящей из методов, технических средств, инструментов и компетентных людских ресурсов для осуществления анализа процессов, моделирования, создания прототипов, проектирования, интегрирования, испытания и документирования. Эти элементы разрабатываются или приобретаются по мере необходимости для поддержки этапа разработки.

4.2.3. Этап внедрения

Этап внедрения начинается с разрешения на установку программного продукта в местах его функционирования. Программная система в определенных случаях может производиться индивидуально, собираться, интегрироваться и испытываться или может производиться массово. На этом этапе осуществляются подготовка инфраструктуры пользователя и оборудование помещений, в которых будет осуществляться функционирование программного продукта, проверка работоспособности программного продукта в реальных условиях эксплуатации, а также организуется обучение персонала работе с программным продуктом или его компонентами.

Планирование этапа внедрения начинается на предыдущем этапе (этапе разработки). Тиражирование программного продукта может продолжаться в течение всего остатка жизненного цикла программной системы.

Выходными элементами этапа внедрения являются следующие основные артефакты:

- акт о завершении разработки;
- акт передачи в эксплуатацию.

Основная часть этапа заканчивается вводом программного продукта в эксплуатацию.

4.2.4. Этап эксплуатации

Этап эксплуатации начинается с установки и документального подтверждения начала использования программного продукта. Использование программного продукта производится в целях получения требуемых программных услуг при непрерывной функциональной и стоимостной эффективности. Этот этап продолжается до начала этапа утилизации.

Планирование этапа эксплуатации начинается на предыдущих этапах. Данный этап включает процессы, которые связаны с использованием программных продуктов для предоставления услуг, а также контроль функционирования, выявления и документирования возникающих проблем.

Действия (либо отсутствие каких-либо предпринимаемых действий) по выявленным проблемам могут быть следующими:

- утверждение акта ввода в эксплуатацию;
- обслуживание программного продукта;
- инициирование действий сопровождения при незначительных модификациях;
- дополнительное инициирование этапа разработки в случае крупных модификаций;
- инициирование этапа списания.

Во время этапа эксплуатации программный продукт или его отдельные части могут модифицироваться и развиваться, приводя к образованию различных конфигураций. Разработчик модификаций должен обеспечивать управление версионностью программных элементов.

Выходными элементами этапа эксплуатации являются следующие основные артефакты:

- акт ввода в эксплуатацию;
- предложение о модификации;
- сообщения о проблемах.

4.2.5. Этап сопровождения

Этап сопровождения может быть активизирован процессом эксплуатации путем представления предложения о модификации или сообщения о проблеме. Процесс сопровождения применяется для модификации программного продукта при сохранении его работоспособности и целостности данных. Этим процессом контролируется работа программного продукта, регистрируются проблемы для анализа, предпринимаются предупреждающие и корректирующие действия, а также действия по адаптации и усовершенствованию программного продукта.

Применяются следующие основные типы сопровождения:

- корректирующее сопровождение;
- предупреждающее сопровождение;
- адаптивное сопровождение;
- полное сопровождение.

Планирование этапа сопровождения начинается на предыдущих этапах. Выходными элементами этапа сопровождения являются следующие основные артефакты:

- концепция сопровождения;
- план сопровождения;
- согласованный вариант изменений;
- измененный программный продукт и документация;
- акт передачи в эксплуатацию модифицированного программного продукта.

Данный этап завершается одновременно с завершением этапа эксплуатации. При необходимости на этапе сопровождения проводятся обучение и переподготовка персонала.

4.2.6. Этап утилизации

Этап утилизации предусматривает изъятие из эксплуатации программного продукта и связанных с ним программных подсистем и вспомогательных программных процессов. Все связанные с прежним программным продуктом подсистемы, документы и данные должны быть помещены в архивы. Данные должны быть защищенными и доступными для аудиторской проверки.

Планирование этапа утилизации начинается на предыдущих этапах.

Выходными элементами этапа утилизации являются следующие основные артефакты:

- изъятый из эксплуатации программный продукт;
- архив изъятых артефактов.

Этап утилизации начинается, когда программный продукт снимается с эксплуатации и продолжается в течение активного использования остальных элементов программной системы.

4.3. Роли жизненного цикла программной системы

При выполнении процессов жизненного цикла программной системы определяются следующие основные роли:

- менеджер проекта;
- представитель заказчика;
- юридический консультант;
- аналитик бизнес-процессов;
- менеджер разработки программной системы;
- бизнес-проектировщик;
- системный аналитик;
- архитектор программной системы;
- проектировщик программной системы;
- проектировщик пользовательского интерфейса;
- программист;
- интегратор;
- тестировщик;
- разработчик тестов;
- технический писатель;
- менеджер внедрения программного продукта;
- технолог бизнес-процессов;
- системный администратор;
- администратор баз данных;
- менеджер эксплуатации;
- пользователь программного продукта;
- менеджер сопровождения программного продукта;
- специалист управления конфигурацией;
- аналитик службы сопровождения;
- тестировщик службы сопровождения.

При реализации конкретных проектов допускаются дополнение состава ролей, предопределенных настоящим техническим регламентом, а также выполнение нескольких ролей одним исполнителем.

4.4. Процессы жизненного цикла программной системы

Определенные в настоящем техническом регламенте процессы жизненного цикла программной системы могут использоваться любой организацией (подразделением), приобретающей и использующей, и/или создающей и поставляющей программную систему. Процессы могут применяться на любом уровне в иерархии программной системы и на любом этапе жизненного цикла программной системы.

Процессы жизненного цикла программной системы базируются на принципах модульности (максимальная связанность функций процесса и минимальная связь между процессами) и ответственности (процесс ассоциируется с ответственностью за его реализацию). Функции, выполняемые этими процессами, определяются специфическими целями, требуемыми результатами и наборами деятельности, которые составляют этот процесс.

Любой процесс, применяемый в жизненном цикле программной системы, продолжителен во времени. Структурно он может состоять из трех фаз, как показано на рисунке 3.

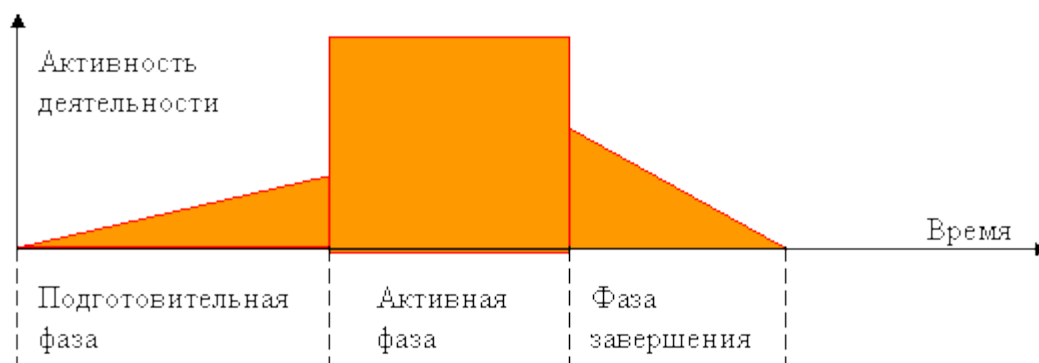


Рисунок 3. Структура выполнения процессов

Во время подготовительной фазы может происходить ознакомление с предметной областью, изучение результатов ранее выполнявшихся процессов, производиться предварительные планирование и расчеты.

Во время активной фазы происходит непосредственное выполнение процесса. В настоящем техническом регламенте при описании процессов освещаются действия только активной фазы.

Во время фазы завершения могут выполняться действия по анализу результатов, завершаться деятельности, начатые во время активной фазы, продолжаться работы, не ограниченные сроком окончания.

Организация (сторона) осуществляет процессы жизненного цикла программной системы избирательно для достижения цели и результатов этапов жизненного цикла системы. Описываемые в настоящем техническом регламенте процессы могут быть дополнены теми процессами, которые организация в зависимости от проекта считает необходимыми для достижения максимальной эффективности системы.

Для каждого процесса определяются цели и результаты, а также деятельности, необходимые для их достижения.

Каждый процесс должен быть задокументирован.

4.5. Перечень процессов

Процессы жизненного цикла программной системы объединяются в четыре группы, согласно таблице 1.

Таблица 1. Процессы жизненного цикла программной системы

Группа	Наименование процесса
Процессы договора	Процесс приобретения
	Процесс поставки
Процессы предприятия	Процесс управления средой предприятия
	Процесс управления инвестициями
	Процесс управления процессами жизненного цикла программной системы
	Процесс управления ресурсами
	Процесс управления качеством
Процессы проекта	Процесс планирования проекта
	Процесс оценки проекта
	Процесс контроля проекта
	Процесс принятия решений
	Процесс управления рисками
	Процесс управления конфигурацией
	Процесс управления информацией
Технические процессы	Процесс концептуального моделирования

Процесс прикладного моделирования
Процесс определения требований заинтересованного лица
Процесс анализа требований
Процесс структурного проектирования
Процесс воплощения
Процесс интеграции
Процесс проверки
Процесс утверждения
Процесс перехода
Процесс эксплуатации
Процесс сопровождения
Процесс изъятия

В настоящем техническом регламенте рассмотрены общие положения процессов договора, предприятия, проекта и подробно технические процессы.

4.6. Использование процессов

Типовое использование процессов показано на рисунке 4.

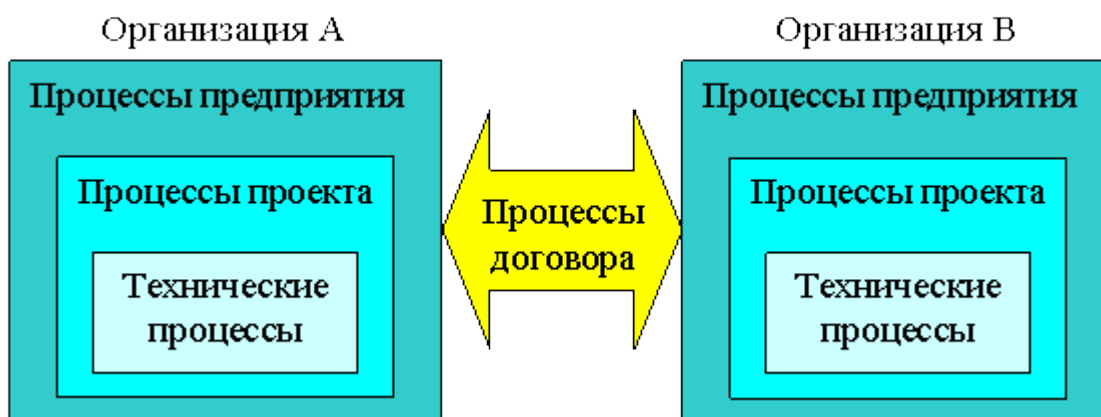


Рисунок 4. Типовое использование процессов

В рамках проекта может осуществляться параллельное использование процессов, например, когда проектные действия и подготовительные действия для создания программной системы выполняются в одно и то же время, а также между проектами, например, когда программные элементы разрабатываются в одно и то же время с различной ответственностью проекта.

Повторяющееся использование процессов важно для поступательного уточнения результатов процесса. Например, взаимодействие между последовательными проверочными действиями и интеграционными действиями может постепенно укреплять уверенность в соответствии программного продукта требованиям заказчика.

Рекурсивное использование процессов на каждом уровне детализации в жизненном цикле программной системы является ключевым аспектом применения настоящего технического регламента. Результаты процессов на любом уровне, будь то структура, программные элементы или программные услуги, могут являться входными элементами для тех же процессов, используемых на более высоком или низком уровне.

4.7. Процессы договора

Процессы договора определяют требования для заключения договоров с организационными единицами, внешними и внутренними для организации.

ПРИМЕЧАНИЕ. Под договором следует понимать документ, определяющий взаимоотношения сторон и сам предмет договора. Таким документом может быть контракт, приказ, распоряжение и др.

Процессами договора являются:

а) процесс приобретения;

б) процесс поставки.

Процессы договора определяют деятельности между двумя сторонами - приобретателем и поставщиком. Поставщик применяет процесс поставки для организации проекта, результатом которого будет поставка приобретателю программного продукта или услуги. Приобретатель использует процесс приобретения для организации работ с поставщиком в целях получения программной системы или ее элементов, поддержки системы или иных программных услуг.

4.7.1. Процесс приобретения

4.7.1.1. Цель процесса приобретения

Цель процесса приобретения заключается в получении программного продукта или программной услуги в соответствии с требованиями приобретателя (заказчика).

4.7.1.2. Результаты процесса приобретения

В результате успешной реализации процесса приобретения:

а) определяется стратегия приобретения;

б) выбирается (назначается) поставщик;

с) определяется взаимосвязь с поставщиком;

д) заключается договор или принимается иной документ для приобретения программного продукта или услуги;

е) принимается программный продукт в соответствии с договором или требованиям к системе;

ф) предоставляется оплата или выполняется иное действие для завершения процесса.

4.7.2. Процесс поставки

4.7.2.1. Цель процесса поставки

Цель процесса поставки заключается в обеспечении приобретателя (заказчика) программным продуктом или услугой согласно утвержденным требованиям.

4.7.2.2. Результаты процесса поставки

а) определяется конкретный приобретатель (заказчик программного продукта или услуги);

б) составляется ответ на запрос приобретателя;

с) заключается договор или принимается иной документ на поставку программного продукта или услуги согласно критериям приемки;

д) в соответствии с согласованными условиями и процедурами поставки поставляется программный продукт, соответствующий договору или утвержденным требованиям;

е) передается приобретателю ответственность за приобретенный программный продукт или услугу;

ф) принимается оплата или выполняется иное действие для завершения процесса.

4.8. Процессы предприятия

Процессы предприятия управляют способностью организации приобретать или поставлять программные продукты или услуги посредством инициирования, поддержки и управления проектами. Эти процессы обеспечивают ресурсы и инфраструктуру, необходимые для поддержки проектов и выполнения заключенных договоров.

Процессами предприятия являются:

а) процесс управления средой предприятия;

б) процесс управления инвестициями;

с) процесс управления процессами жизненного цикла программной системы;

д) процесс управления ресурсами;

е) процесс управления качеством.

4.8.1. Процесс управления средой предприятия

4.8.1.1. Цель процесса управления средой предприятия

Цель процесса управления средой предприятия заключается в определении и поддержании политики и процедур, необходимых для деятельности организации согласно настоящему техническому регламенту.

4.8.1.2. Результаты процесса управления средой предприятия

В результате успешной реализации процесса управления средой предприятия:

- a) формируются политика и процедуры для управления жизненным циклом программной системы согласно целям предприятия;
- b) определяется политика приспособления процессов жизненного цикла программной системы для удовлетворения нужд и потребностей этой системы;
- c) определяются правила применения процессов жизненного цикла программной системы, а также методология разработки программных систем;
- d) определяются ответственность и полномочия при управлении жизненным циклом программной системы;
- e) определяются политика и процедуры управления качеством;
- f) формируется политика совершенствования процессов жизненного цикла программной системы.

4.8.2. Процесс управления инвестициями

4.8.2.1. Цель процесса управления инвестициями

Цель процесса управления инвестициями заключается в инициировании и поддержке инвестиционных проектов, обеспечивающих достижение целей программной системы. Этим процессом осуществляется вложение соответствующих фондов и ресурсов организации и санкционируются органы, необходимые для учреждения выбранных инвестиционных проектов. Этим процессом непрерывно осуществляется контроль над проектами для обоснования инвестиций.

4.8.2.2. Результаты процесса управления инвестициями

В результате успешной реализации процесса управления инвестициями:

- a) определяются и выбираются инвестиционные возможности или потребности;
- b) определяются и распределяются ресурсы и бюджеты;
- c) определяются ответственность и полномочия для управления инвестиционными проектами;
- d) составляются планы реализации проектов;
- e) поддерживаются проекты, удовлетворяющие требованиям договора заинтересованного лица или организации;
- f) проекты, не удовлетворяющие требованиям договора заинтересованного лица или организации, прекращаются или изменяются.

4.8.3. Процесс управления процессами жизненного цикла программной системы

4.8.3.1. Цель процесса управления процессами жизненного цикла программной системы

Цель процесса управления процессами жизненного цикла программной системы заключается в обеспечении наличия эффективных процессов жизненного цикла программной системы, которые использует организация. Этот процесс обеспечивает реализацию процессов жизненного цикла программной системы, которые должны отвечать целям и политике организации. Процессы определяются, адаптируются и поддерживаются необходимым образом, используя эффективные, проверенные методы и инструменты.

4.8.3.2. Результаты процесса управления процессами жизненного цикла системы

В результате успешной реализации процесса управления процессами жизненного цикла системы:

- a) определяются процессы жизненного цикла системы, которые должны использоваться организацией;
- b) определяется методология разработки программных систем;

с) определяется политика по применению процессов жизненного цикла программной системы, их адаптации к особенностям индивидуальных проектов;

д) определяются критерии оценки применения процессов жизненного цикла программной системы.

4.8.4. Процесс управления ресурсами

4.8.4.1. Цель процесса управления ресурсами

Цель процесса управления ресурсами заключается в обеспечении проектов ресурсами. Этот процесс обеспечивает конкретные, надежные и возобновляемые ресурсы, материалы и услуги для поддержки проектов и процессов в течение всего жизненного цикла программной системы. Процесс также включает предоставление обученного, квалифицированного и опытного персонала для выполнения процессов жизненного цикла.

Этим процессом обеспечиваются эффективная координация и совместное использование ресурсов, информации и технологий.

4.8.4.2. Результаты процесса управления ресурсами

В результате успешной реализации процесса управления ресурсами:

- а) проекты обеспечиваются необходимыми средствами, материалами и услугами;
- б) разрешаются конфликты в точках сопряжения нескольких проектов.

4.8.5. Процесс управления качеством

4.8.5.1. Цель процесса управления качеством

Цель процесса управления качеством заключается в обеспечении соответствия программного продукта, услуги и применения процессов жизненного цикла программной системы договорам, утвержденным проектным планам и требованиям приобретателя.

4.8.5.2. Результаты процесса управления качеством

В результате успешной реализации процесса управления качеством:

- а) в рамках организации создается система качества, которая обеспечивает достижение требуемых проектных уровней качества;
- б) предоставляется информация о состоянии качества каждого проекта;
- с) организацией (подразделениями) для каждого проекта планируются и выполняются деятельности по управлению системой качества согласно международным стандартам ISO серии 9000.

4.9. Процессы проекта

Процессы проекта используются для разработки и развития планирования проекта, оценки фактических достижений, контроля над выполнением проекта вплоть до его завершения. Отдельные процессы проекта могут использоваться в любое время в течение жизненного цикла программной системы и на любом уровне иерархии проектов в соответствии с требованиями проектных планов или непредвиденными событиями. Применение процессов проекта может зависеть от рисков и сложности проекта.

Процессами проекта являются:

- а) процесс планирования проекта;
- б) процесс оценки проекта;
- с) процесс контроля проекта;
- д) процесс принятия решений;
- е) процесс управления рисками;
- ф) процесс управления конфигурацией;
- г) процесс управления информацией.

ПРИМЕЧАНИЕ. Планирование, оценка и контроль являются ключевыми процессами любой практической деятельности по управлению проектом. Они предусматриваются в управлении любым предприятием, начиная от организации в целом и заканчивая отдельным процессом жизненного цикла программной системы и его деятельностями.

4.9.1. Процесс планирования проекта

4.9.1.1. Цель процесса планирования проекта

Цель процесса планирования проекта заключается в создании эффективного и работоспособного плана проекта. В этом процессе определяются рамки деятельностей и

комплексов работ, устанавливаются графики проведения этих работ, включая календарные планы, а также распределяются ресурсы для выполнения комплекса работ.

Процесс планирования проекта осуществляется на всех этапах жизненного цикла программной системы и применяется для всех процессов. Первоначально разработанные планы уточняются при каждом переходе к следующему этапу или процессу вплоть до полного их исполнения или прекращения проекта.

Согласно договорным обязательствам ответственная сторона (организация) должна разработать план руководства проектом согласно приложению 1 и календарный план реализации проекта согласно приложению 2. Ей необходимо рассмотреть и проанализировать требования технического задания на разработку программной системы для определения рамок, в которых осуществляется руководство и обеспечение проекта, а также степень вовлеченности заказчика. При необходимости допускается разработка других проектных планов, применяемых к отдельным процессам жизненного цикла программной системы.

В случае необходимости на основании плана руководства проектом поставщик может заключать контракт (договор) с внешними организациями (субподрядчиками).

4.9.1.2. Результаты процесса планирования проекта

В результате успешной реализации процесса планирования проекта:

- a) определяются роли, ответственность и полномочия сторон;
- b) разрабатывается и утверждается план руководством проекта;
- c) разрабатываются и утверждаются календарный план реализации проекта и другие необходимые проектные планы;
- d) формально запрашиваются ресурсы и услуги, необходимые для выполнения проекта;
- e) определяются критерии для показателей реализации проекта;
- f) кадры проекта проходят обучение (инструктаж) в соответствии с планом проекта.

4.9.2. Процесс оценки проекта

4.9.2.1. Цель процесса оценки проекта

Цель процесса оценки проекта заключается в определении состояния проекта. Этим процессом периодически оцениваются прогресс выполнения плана управления проектом, технических планов и общих предпринимательских целей, определенных в других планах проекта. При выявлении существенных отклонений информация о них доводится до сведения заинтересованных сторон, и предпринимаются управленческие действия.

4.9.2.2. Результаты процесса оценки проекта

В результате успешной реализации процесса оценки проекта:

- a) предоставляются данные об исполнении проекта или результаты его оценки;
- b) оценивается адекватность ролей, прав и обязанностей участников проекта;
- c) оценивается адекватность ресурсов и услуг, необходимых для выполнения проекта;
- d) анализируются отклонения в показателях исполнения проекта;
- e) вовлеченные стороны информируются о состоянии проекта.

4.9.3. Процесс контроля проекта

4.9.3.1. Цель процесса контроля проекта

Цель процесса контроля проекта заключается в руководстве реализацией проекта в рамках бюджетных средств проекта и в удовлетворении техническим требованиям проекта. В некоторых случаях этот процесс может изменять деятельности проекта для исправления выявленных отклонений в управлении проектом или технических процессов. При необходимости изменения могут включать повторное планирование.

4.9.3.2. Результаты процесса контроля проекта

В результате успешной реализации процесса контроля проекта:

- a) определяется и направляется корректирующее действие, если достижения проекта не удовлетворяют запланированным целям;

- b) инициируется повторное планирование проекта, если изменились цели и ограничения проекта или плановые предположения оказались недействительными;
- c) санкционируется (или нет) действие по переходу проекта с одного запланированного этапа на другой;
- d) достигаются цели проекта.

4.9.4. Процесс принятия решений

4.9.4.1. Цель процесса принятия решений

Цель процесса принятия решений заключается в выборе наиболее выгодного пути развития проекта, если существуют альтернативы. Этот процесс требует принятия решения для достижения оговоренных, желательных или оптимизированных результатов во время жизненного цикла программной системы. Анализируются альтернативные действия, а также выбирается и направляется ход этих действий. Решения и их обоснование должны регистрироваться.

4.9.4.2. Результаты процесса принятия решений

В результате успешной реализации процесса принятия решений:

- a) определяются обстоятельства и потребность в решении;
- b) определяются альтернативные пути осуществления действий;
- c) выбирается предпочтительный ход действий;
- d) накапливаются и отражаются в отчетах решения, их обоснования и предпосылки.

4.9.5. Процесс управления рисками

4.9.5.1. Цель процесса управления рисками

Цель процесса управления рисками заключается в минимизации воздействия возможных неопределенных событий. Такие события могут привести к увеличению стоимости программной системы, срыву проектных планов и/или ухудшению технических характеристик.

Процесс определяет, оценивает и управляет всеми рисками в течение всего жизненного цикла программной системы, включая риски, влияющие на успех проекта, а также риски, возникающие во время использования программного продукта.

Поставщик обязан определить и проанализировать риски, присущие данному проекту, свести их воздействие к минимуму и учесть возможные их воздействия в процессе планирования проекта.

Риски, воздействующие на проект, делятся на четыре категории:

- a) риски, связанные с требованиями к программной системе, являются самыми опасными и заключаются в том, что система может не удовлетворять требованиям заказчика;
- b) технологические риски - связанные с работоспособностью элементов системы, архитектурным решением и сборкой разнородных компонентов в единую систему;
- c) риски, связанные с квалификацией персонала - определяются опытом и профессионализмом сотрудников, задействованных в проекте;
- d) политические риски - связаны с корпоративной политикой взаимодействующих сторон, а также особенностями организаций и деятельностью составляющих подразделений.

4.9.5.2. Результаты процесса управления рисками

В результате успешной реализации процесса управления рисками:

- a) оценка рисков осуществляется в течение всего жизненного цикла программной системы;
- b) определяются и классифицируются риски;
- c) количественно определяется неблагоприятное воздействие рисков;
- d) определяются меры по предотвращению рисков;
- e) определяется стратегия управления каждым выявленным риском;
- f) предпринимаются меры в отношении рисков, составляющих проблему;
- g) ведется отчетность о рисках и их состоянии.

4.9.6. Процесс управления конфигурацией

4.9.6.1. Цель процесса управления конфигурацией

Цель процесса управления конфигурацией заключается в установлении и поддержании актуальности и целостности всех продуктов и элементов проекта или процесса.

4.9.6.2. Результаты процесса управления конфигурацией

В результате успешной реализации процесса управления конфигурацией:

- a) определяется стратегия управления конфигурацией;
- b) определяются элементы, требующие управления конфигурацией;
- c) устанавливаются основы конфигурации;
- d) отслеживаются изменения в элементах конфигурации;
- e) контролируется выпуск элементов конфигурации;
- f) информация о статусе управляемых элементов конфигурации обеспечивается в течение жизненного цикла программной системы.

4.9.7. Процесс управления информацией

4.9.7.1. Цель процесса управления информацией

Цель процесса управления информацией заключается в обеспечении вовлеченных сторон своевременной, полной, действительной, конфиденциальной (если требуется) информацией в течение жизненного цикла программной системы и, в соответствующих случаях, после него. Этим процессом создается, собирается, обрабатывается, сохраняется, извлекается, распространяется и используется информация, относящаяся к программному продукту. Этот процесс управляет всей информацией, включая техническую и проектную информацию, информацию предприятия, договоров и пользователя.

4.9.7.2. Результаты процесса управления информацией

В результате успешной реализации процесса управления информацией:

- a) определяется вся подлежащая управлению информация;
- b) собираются, сохраняются и отслеживаются артефакты и данные, получаемые в результате работ;
- c) определяются формы представления информации;
- d) регистрируется статус информации;
- e) поддерживаются актуальность, полнота и действительность информации;
- f) информация предоставляется вовлеченным сторонам.

4.10. Технические процессы

Технические процессы используются для определения необходимости в программной системе, преобразования этой потребности в эффективный программный продукт, обеспечения устойчивого воспроизведения программного продукта, когда это необходимо, использования программного продукта с целью предоставления необходимых услуг, поддержки предоставления этих услуг и, когда программный продукт снимается со службы, утилизации этого продукта.

Технические процессы определяют деятельности, которые позволяют функциям предприятия и проекта оптимизировать преимущества программной системы и уменьшить риски, возникающие в результате технических решений и действий. Эти деятельности обеспечивают придание программным продуктам и услугам таких качеств, как наличие, своевременность, затратная эффективность, функциональность, надежность, ремонтпригодность, масштабируемость, полезность и ряд других качеств, которые требуют приобретающие и поставляющие организации. Они также обеспечивают соответствие программных продуктов и услуг законодательным требованиям общества, включая здравоохранение, безопасность, защищенность персонала и факторы окружающей среды.

Технические процессы и основные деятельности, заканчивающиеся созданием артефактов, распределяются по этапам жизненного цикла программной системы, как указано на рисунке 5.

Этапы	Процессы или основные деятельности	Временные диаграммы выполнения процессов и основных деятельностей	Основные артефакты
Предпроектные работы	Концептуальное моделирование		Концепция
	Прикладное моделирование		Бизнес-предложение
Разработка	Определение требований		Техническое задание
	Анализ требований		Техническое задание
	Структурное проектирование		Технический проект
	Воплощение		Программный продукт Документация
	Интеграция		Версия программной системы
	Проверка (тестирование)		Протокол. Акт передачи в опытную эксплуатацию
Внедрение	Утверждение (опытная эксплуатация)		Акт о завершении разработки
	Переход (перенос)		Акт передачи в эксплуатацию
Эксплуатация	Использование		Акт ввода в эксплуатацию. Программные услуги
	Контроль и анализ эксплуатации		Предложение о модификации. Сообщение о проблеме
Сопровождение	Подготовка		Концепция сопровождения. План сопровождения
	Анализ проблем		Согласованный вариант изменения
	Внесение изменений		Измененный программный продукт и документация
	Проверка и приемка		Акт передачи в эксплуатацию
Утилизация	Изъятие системы из эксплуатации		Изъятый из эксплуатации программный продукт

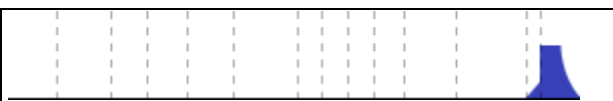
	Хранение элементов и данных		Архив изъятых артефактов
--	-----------------------------	--	--------------------------

Рисунок 5. Технические процессы жизненного цикла программной системы

4.10.1. Процесс концептуального моделирования

4.10.1.1. Цель процесса концептуального моделирования

Цель процесса концептуального моделирования заключается в предоставлении заинтересованным сторонам модели (общего видения) программной системы (или группы взаимосвязанных систем), выполняемых ею функций, описания информационного пространства и взаимодействия с другими внешними АИС.

Этим процессом проводится обследование области деятельности, в которой будет функционировать программная система, анализ уровня информатизации данной области, выясняются необходимость и цели создания или модификации системы. Также проводится изучение нормативно-правового пространства и организационной структуры, осуществляющей управление данной областью. При необходимости разрабатываются предложения по их актуализации. На основе полученных знаний строится модель функционирования АИС.

Данный процесс может быть инициирован указаниями президента и правительства, изданием нормативно-правовых документов, или требованием заказчика АИС.

За действия в рамках этого процесса отвечает поставщик. Ему необходимо оформить проект концепции и подготовить документы, необходимые для ее согласования. Структура концепции системы - согласно приложению 3.

Оформление, согласование и утверждение концепции осуществляются в соответствии с:

а) законом Республики Молдова о нормативных актах Правительства и других органов центрального и местного публичного управления № 317-XV от 18.07.2003 г. („Monitorul Oficial al Republicii Moldova” № 208-210/783 от 03.10.2003 г.);

б) Постановлением Правительства о порядке проведения юридической экспертизы и государственной регистрации ведомственных нормативных актов №1104 от 28.11.1997 г. („Monitorul Oficial al Republicii Moldova” № 6-7/10 от 21.01.1998 г.).

4.10.1.2. Результаты процесса концептуального моделирования

В результате успешной реализации процесса концептуального моделирования:

- а) определены цель и назначение моделируемой АИС;
- б) изучены нормативно-правовая база, организационная структура управления и разработаны предложения по их актуализации;
- с) определены функциональность, информационные объекты и потоки, пути интеграции с другими системами;
- д) концепция оформлена, согласована и утверждена заказчиком.

4.10.1.3. Деятельности процесса концептуального моделирования

Деятельности данного процесса показаны на рисунке 6.

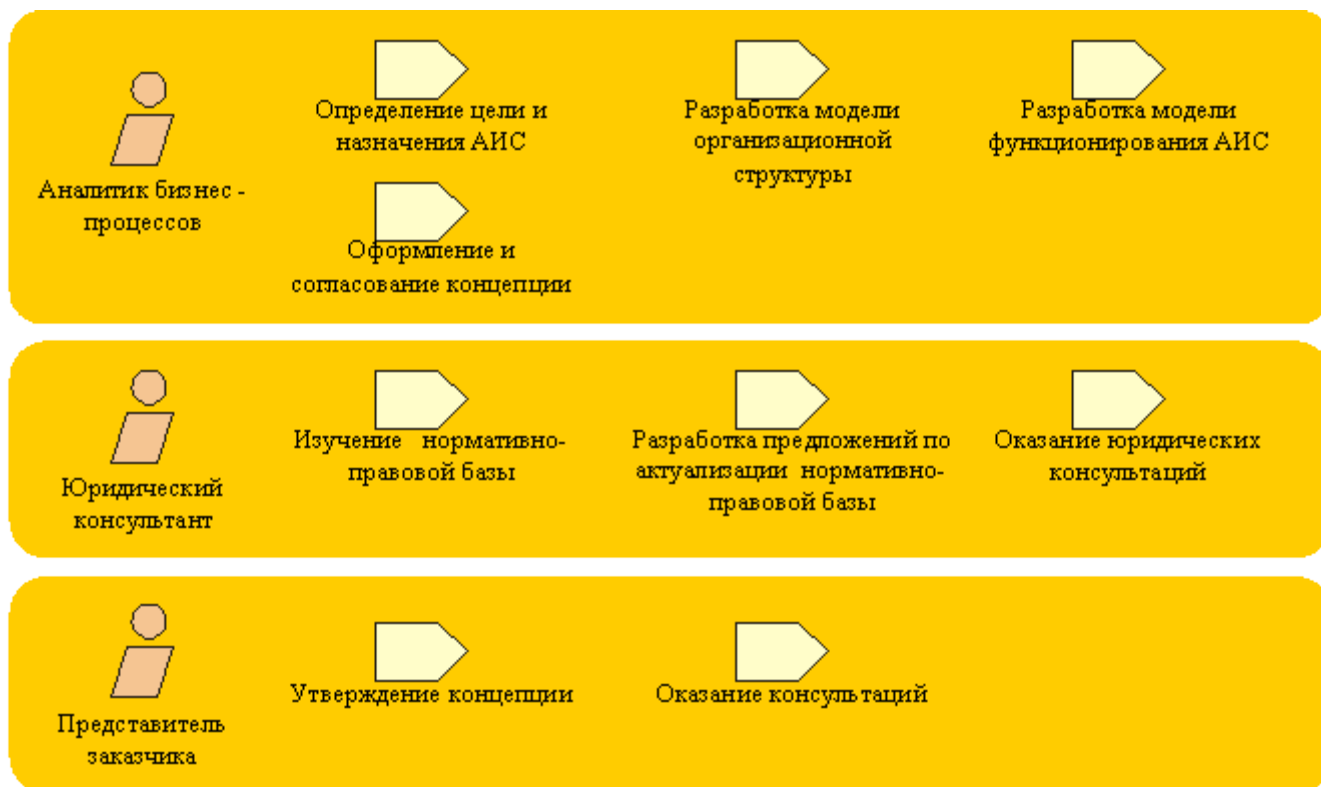


Рисунок 6. Деятельности процесса концептуального моделирования

4.10.2. Процесс прикладного моделирования

4.10.2.1. Цель процесса прикладного моделирования

Цель процесса прикладного моделирования заключается в уяснении назначения программной системы, степени автоматизации действий заказчика или АИС, способов и методов получения, обработки и хранения информации и/или выполнения программных услуг. Этим процессом создается описание варианта предлагаемого поставщиком программного продукта, указываются возможные изменения инфраструктуры пользователя, обеспечивающие выполнение необходимых бизнес-процессов, при необходимости делается предварительный расчет его стоимости или затрат на реализацию и эксплуатацию.

Описанный вариант программного продукта поставщик предлагает заказчику в виде бизнес-предложения, краткое содержание которого определено в приложении 1.

Заказчик анализирует предложенный вариант программного продукта, утверждает бизнес-предложения и передает их поставщику для дальнейшей работы по сбору требований и анализу.

4.10.2.2. Результаты процесса прикладного моделирования

В результате успешной реализации процесса прикладного моделирования:

- a) изучена предметная область;
- b) определены и согласованы цели, назначение и основная функциональность программной системы;
- c) разработан единый словарь терминов;
- d) определены заинтересованные стороны, а также взаимодействующие организации и сторонние АИС;
- e) определены бизнес-процессы и бизнес-роли программной системы;
- f) разработаны предварительная архитектура программной системы и схема движения информации;
- g) утверждены артефакты и передаются для дальнейшей работы.

4.10.2.3. Деятельности процесса прикладного моделирования

Деятельности данного процесса показаны на рисунке 7.

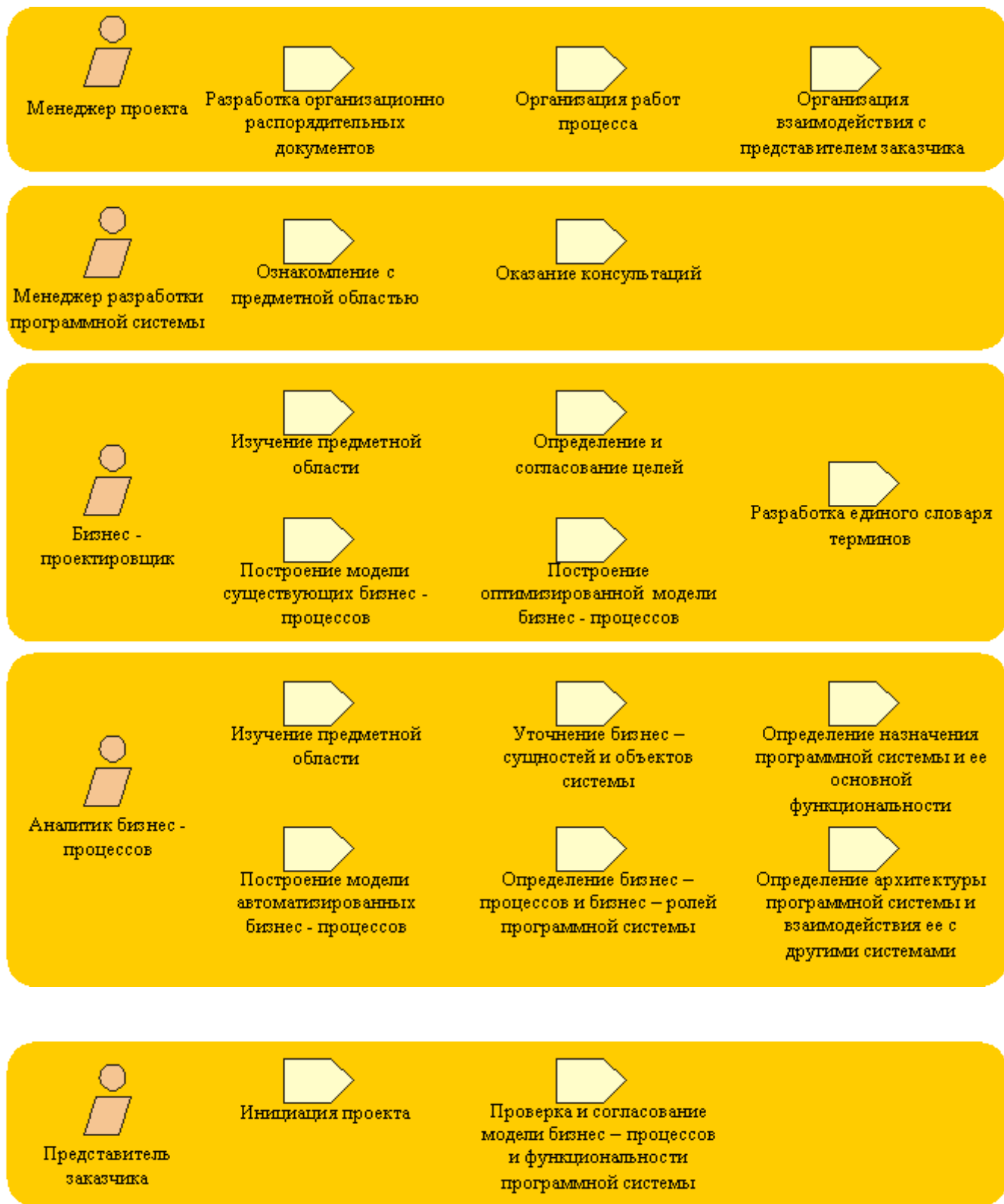


Рисунок 7. Деятельности процесса прикладного моделирования

4.10.3. Процесс определения требований заинтересованного лица

4.10.3.1. Цель процесса определения требований заинтересованного лица

Цель процесса определения требований заинтересованного лица заключается в определении требований к программной системе, которая должна предоставлять услуги, требуемые заказчиком.

Этим процессом определяются требования заказчика, которые анализируются и преобразуются в приемлемые требования к программной системе. Требования заказчика должны включать потребности и желания всех сторон, заинтересованных в жизненном цикле программной системы. Требования заказчика выражают предполагаемое взаимодействие программной системы с заинтересованными лицами и являются эталоном, по которому проверяется функциональность программного продукта и его способность предоставлять услуги.

Для этих целей заказчик выделяет в проект своего полномочного представителя и наделяет его функциями принятия решения. Степень участия заказчика в разработке может оговариваться в договоре или плане руководства проектом.

Также этим процессом уточняются заинтересованные лица или группы заинтересованных лиц, причастных к программной системе в течение ее жизненного цикла и выражаются потребности, желания и ожидания заинтересованных лиц вместе с ограничениями, накладываемыми этими лицами и условиями эксплуатации.

4.10.3.2. Результаты процесса определения требований заинтересованного лица

В результате успешной реализации процесса определения требований заинтересованного лица:

- детализированы бизнес-процессы, определены бизнес-функции процессов и создана модель артефактов;
- оговорены требуемые характеристики программной системы и инфраструктура пользователя, в которой она будет использоваться;
- определены ограничения программной системы и ее элементов;
- достигнута постоянная отслеживаемость требований заинтересованного лица;
- определена основа для анализа требований к программной системе;
- обеспечена основа для ведения переговоров и согласования поставки услуг или продукта.

4.10.3.3. Деятельности процесса определения требований заинтересованного лица

Деятельности данного процесса показаны на рисунке 8.





Рисунок 8. Деятельности процесса определения требований заинтересованного лица

4.10.4. Процесс анализа требований

4.10.4.1. Цель процесса анализа требований

Цель процесса анализа требований к программному обеспечению заключается в преобразовании требований заинтересованного лица в техническое видение требуемого программного продукта. Этим процессом создается представление о будущей программной системе, которая должна удовлетворять потребностям заинтересованного лица без описания какого-либо конкретного воплощения.

В требованиях к программной системе или услугам, с точки зрения разработчика, оговаривается, что должна делать программная система (услуга), чтобы удовлетворить требованиям заинтересованного лица. Эти требования могут быть функциональными, количественными или качественными.

На основании анализа требований заинтересованного лица разработчик с участием заказчика разрабатывает техническое задание.

В техническом задании разработчик обязан установить оговоренные требования к разрабатываемой программной системе, включая спецификации качественных характеристик. Требования, задаваемые в техническом задании, не должны ограничивать разработчика в поиске и реализации наиболее эффективных решений, но могут оговаривать применяемые технологии и методологии разработки. Структура технического задания указана в приложении 4.

Техническое задание согласовывается с руководителями организаций (подразделений), участвовавших в его разработке, и утверждается заказчиком.

Заказчик обязан принять меры к адаптации инфраструктуры пользователя и приспособить его деятельность к разрабатываемой программной системе, т.е. принять решение относительно бизнес-процессов, подлежащих автоматизации и бизнес-ролей, участвующих в процессе. Эти действия также могут включать адаптацию штатной структуры, разработку организационно-распорядительной документации, организацию

необходимых помещений, приобретение оборудования, организацию учебы сотрудников и т.д.

4.10.4.2. Результаты процесса анализа требований

В результате успешной реализации процесса анализа требований:

- a) утверждены модель бизнес-процессов и функциональность системы, а также установлены требуемые характеристики программной системы;
- b) определены проектные ограничения и квалификационные требования к программной системе, а также требования к реализации проекта;
- c) установлена основа для постоянного контроля реализации требований заинтересованного лица;
- d) внедрена система управления изменениями;
- e) обеспечена основа для адаптации инфраструктуры пользователя к требованиям программной системы;
- f) определены требования к техническим средствам и сетевым решениям;
- g) утверждены артефакты и переданы для дальнейшей работы.

4.10.4.3. Деятельности процесса анализа требований

Деятельности данного процесса показаны на рисунке 9.

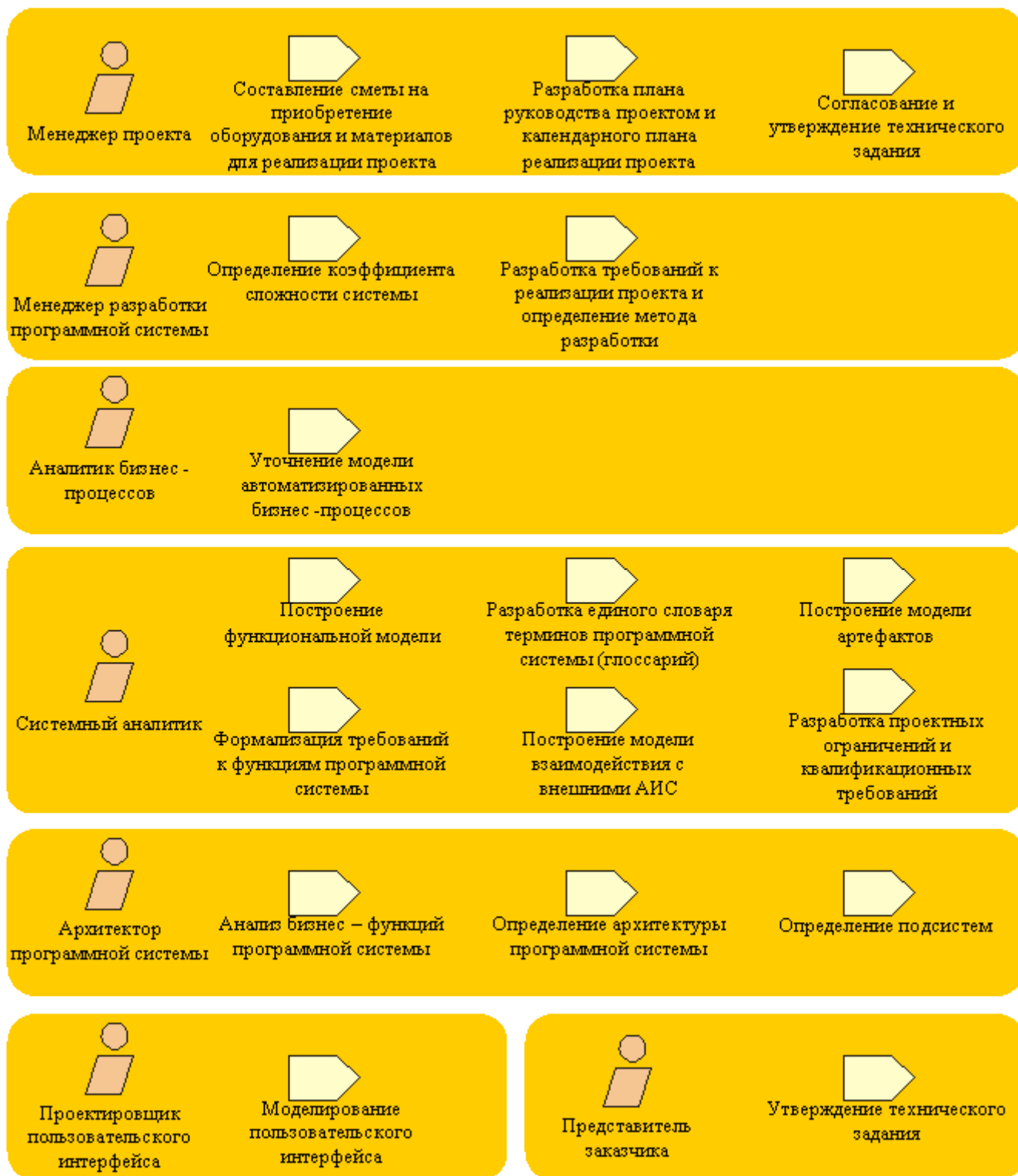


Рисунок 9. Деятельности процесса анализа требований

4.10.5. Процесс структурного проектирования

4.10.5.1. Цель процесса структурного проектирования

Цель процесса структурного проектирования заключается в разработке технического решения, которое удовлетворяет требованиям к программной системе.

Решения структурного проектирования определяют полный набор технически и коммерчески жизнеспособных системных элементов, из которых конфигурируется программная система. Это является основой для проверки соответствия реализуемой

программной системы, а также для планирования и выработки стратегии сборки и тестирования.

Этот процесс охватывает действия и задачи разработчика. Разработчик управляет этим процессом на уровне проекта, создает инфраструктуру процесса и приспособливает его к требованиям проекта.

Решения этого процесса оформляются в техническом проекте. Технический проект необходим для трансформации требований технического задания в описание реализации конкретных модулей или элементов программной системы. Технический проект служит средством общения между лицами, участвующими в проектировании и реализации программной системы.

На основании описанных в техническом задании бизнес-процессов и разработанных моделей разработчик должен создать технический проект, описывающий структуры более низкого уровня для компонентов программной системы. Необходимо разработать архитектуру программной системы и ее структурную схему с ассоциациями между элементами, определить интерфейсы, внешние по отношению к системе и между компонентами самой системы. Краткое содержание технического проекта - в соответствии с приложением 1.

Технический проект согласовывается с менеджером разработки и утверждается руководителем должностных лиц, участвующих в разработке технического проекта.

На основании результатов данного процесса определяется стратегия воплощения программной системы, детализирующая план разработки.

4.10.5.2. Результаты процесса структурного проектирования

В результате успешной реализации процесса структурного проектирования:

- a) определена архитектура программной системы и ее элементов;
- b) для проектируемого системного элемента оговорены методы и технология реализации в виде описания спецификаций, диаграмм, схем, процедур и т.д., которые удовлетворяют оговоренным требованиям заказчика;
- c) проектное решение приведено в соответствие с взаимодействующими программными системами и элементами систем;
- d) определена основа для проверки соответствия (тестирования) программных элементов;
- e) определена основа для приобретения или сборки и интеграции программных элементов;
- f) определена последовательность реализации функций программной системы;
- g) утвержден технический проект.

4.10.5.3. Деятельности процесса структурного проектирования

Деятельности данного процесса показаны на рисунке 10.



Менеджер проекта

Уточнение календарного плана проекта

Уточнение состава рабочей группы и потребности в ресурсах

Согласование с заказчиком списка функций и установление последовательности их реализации

Согласование и утверждение технического проекта



Менеджер разработки программной системы

Утверждение архитектуры программной системы

Организация системы управления изменениями

Планирование первой итерации

Определение языков программирования и средств разработки



Архитектор программной системы

Уточнение архитектуры программной системы и ее элементов

Утверждение иерархии классов

Разработка модели взаимодействия элементов программной системы



Проектировщик программной системы

Выявление всех ассоциаций между элементами программной системы

Детализация модели взаимодействия элементов системы

Разработка интерфейсов

Разработка классов

Разработка артефактов

Разработка модели реализации (логическая структура)

Формирование требований к техническим средствам и сетевым решениям



Проектировщик пользовательского интерфейса

Разработка пользовательского интерфейса



Проектировщик баз данных

Проектирование баз данных и механизмов конвертирования (переноса) данных



Бизнес проектировщик

Разработка предложений по адаптации штатной структуры и должностных инструкций

Разработка предложений по конфигурации рабочих мест



Представитель заказчика

Консультации и согласование проектных решений и артефактов

Согласование инфраструктуры пользователя

Рисунок 10. Деятельности процесса структурного проектирования

4.10.6. Процесс воплощения

4.10.6.1. Цель процесса воплощения

Цель процесса воплощения заключается в создании оговоренной программной системы. Этим процессом проектное решение в соответствии с требованиями заказчика трансформируется в действия создания, в результате которых по выбранной методологии разработки и технологии воплощения создается программный продукт. Задачами программной системы, полученной в результате создания или приспособления, является полное удовлетворение требованиям структурного проекта.

Процесс воплощения включает в себя действия разработчика по созданию и тестированию программной системы.

Разработчик должен создать (запрограммировать) код программных элементов, готовых для компоновки в рабочую систему. Технология и методология разработки должны быть согласованы с поставщиком, а методы программирования соответствовать общепринятым ограничениям. Разработчик обязан снабжать тексты программных элементов, написанных на языке программирования, комментариями, достаточными для понимания техники реализации.

В процессе программирования разработчик может обновлять, изменять или дополнять документацию технического проектирования, если это не противоречит техническому заданию.

Кроме программной системы разработчик должен разработать оговоренные в контракте (договоре) документы для конечного пользователя:

- a) инструкцию по эксплуатации (программной системы или ее элементов);
- b) руководство администратора (программной системы или ее элементов).

Краткое содержание документов - в соответствии с приложением 1.

В процессе воплощения в обязательном порядке должна выполняться деятельность по тестированию программных элементов.

Разработчик должен протестировать каждый программный элемент, чтобы убедиться в их работоспособности, полной функциональности и соответствии техническому заданию. Этот вид тестирования проводится силами разработчика и называется внутренним тестированием.

Деятельность по внутреннему тестированию должна быть непрерывной. Тесты для проверки программных элементов должны создаваться сразу, как только эти элементы реализованы в коде. Первоначально написанный тест должен использоваться многократно.

Результатом выполнения тестов должен быть список ошибок, который заносится в протокол тестирования. Содержание протокола тестирования - в соответствии с приложением 1.

Эффективность внутреннего тестирования обеспечивается применением двух методов тестирования: тестирование элементов программной системы непосредственными разработчиками и тестирование программной системы тестами, реализованными третьими лицами.

Все тесты, созданные в процессе реализации очередной версии, сохраняются и применяются при тестировании следующих версий.

4.10.6.2. Результаты процесса воплощения

В результате успешной реализации процесса воплощения:

- a) определены стратегия воплощения и методология реализации;
- b) приобретены или изготовлены элементы программной системы;
- c) проведено тестирование элементов и программной системы в целом;
- d) при необходимости, обеспечена сертификация программного продукта или получены данные о его проверке;

- е) изготовлена оговоренная документация;
- ф) программная система поставлена или помещена на хранение в соответствии с договором на его поставку.

4.10.6.3. Деятельности процесса воплощения

Деятельности данного процесса показаны на рисунке 11.

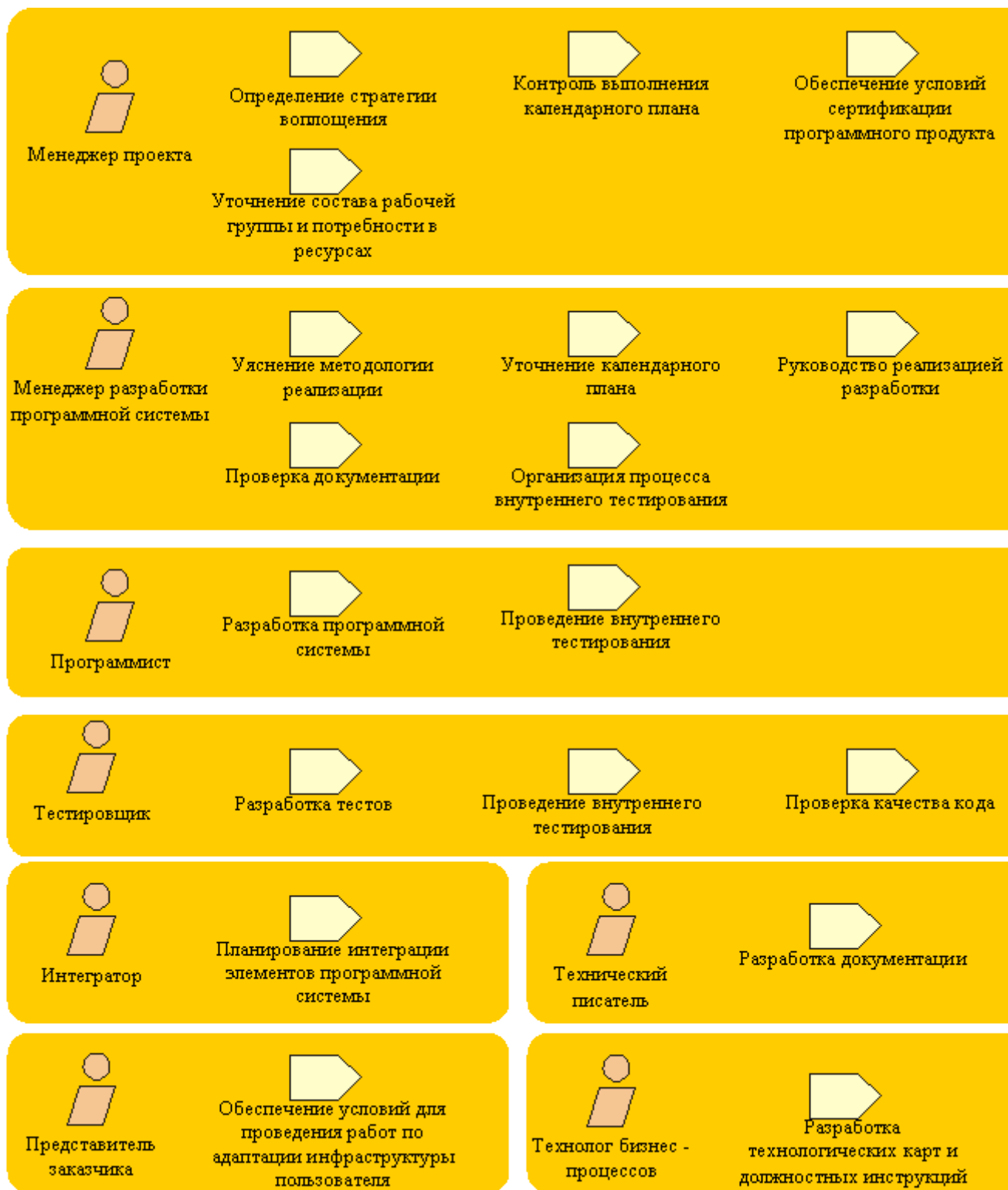


Рисунок 11. Деятельности процесса воплощения

4.10.7. Процесс интеграции

4.10.7.1. Цель процесса интеграции

Цель процесса интеграции заключается в сборке программной системы в соответствии со структурой проекта. Этим процессом программные элементы собираются в работоспособную систему с частичной или полной конфигурацией с целью создания программного продукта, предусмотренного требованиями заказчика.

Созданная конфигурация регистрируется и предоставляется заказчику для дальнейшей проверки и приемки. В процессе жизненного цикла программной системы процесс интеграции может проходить многократно, при каждой итерации, вплоть до полного соответствия требованиям технического задания.

4.10.7.2. Результаты процесса интеграции

В результате успешной реализации процесса интеграции:

- а) определены методы интеграции системы;
- б) собрана и интегрирована программная система, которую можно проверить по оговоренным проектным требованиям и утвердить согласно ожиданиям заинтересованного лица;
- с) зарегистрирована версия программной системы.

4.10.7.3. Деятельности процесса интеграции

Деятельности данного процесса показаны на рисунке 12.

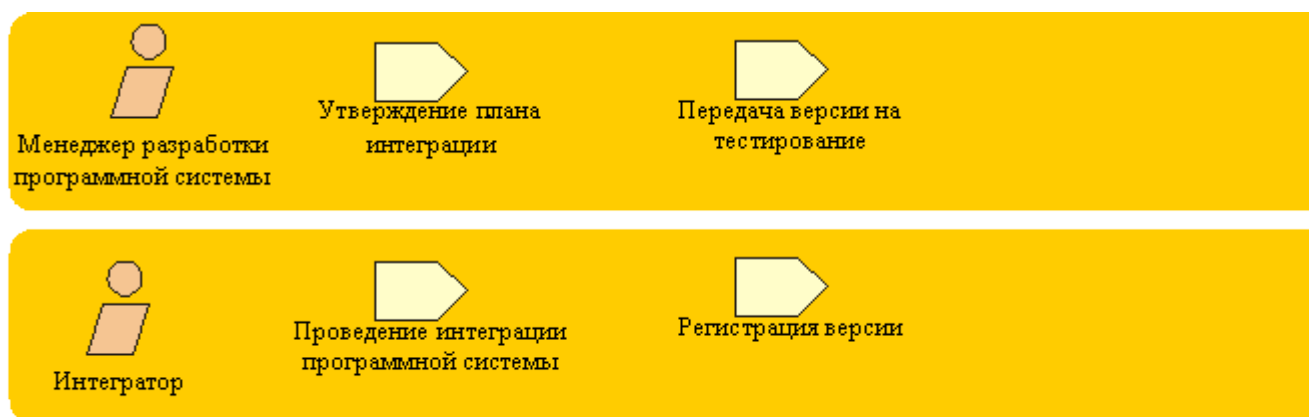


Рисунок 12. Деятельности процесса интеграции

4.10.8. Процесс проверки

4.10.8.1. Цель процесса проверки

Цель процесса проверки заключается в том, чтобы продемонстрировать, что характеристики и функциональность программного продукта соответствуют оговоренным требованиям к программной системе, согласно которым продукт был реализован. Этот процесс предоставляет аналитическую информацию, необходимую для проведения корректирующих действий. Такие действия направлены на корректировку несоответствий в реализованной программной системе/ программном элементе или в процессах, воздействующих на них.

Процесс проверки начинается после окончания разработки и интеграции программной системы и заключается в планировании и проведении квалификационного тестирования. Содержание календарного плана квалификационного тестирования - согласно приложению 1. За действия настоящего процесса отвечает разработчик.

Во время квалификационного тестирования проводятся следующие проверки:

- а) функциональное тестирование программного продукта на соответствие требованиям технического задания;
- б) нагрузочное тестирование, имитирующее нагрузки на программный продукт больше пиковых значений процесса эксплуатации;

- с) проверки граничных значений входных данных, при которых программная система меняет свое поведение;
- д) удобство использования программного продукта и дружелюбность пользовательского интерфейса;
- е) анализ примененных технологий, алгоритмов и программного кода;
- ф) проверка документации на наличие, содержание и качество исполнения, а также соответствия требованиям руководящих документов.

По итогам квалификационного тестирования составляется протокол тестирования. Содержание протокола тестирования - в соответствии с приложением 1.

Ошибки и замечания, выявленные во время квалификационного тестирования, анализируются поставщиком и разработчиком совместно с заказчиком и сортируются по трем группам:

- а) критические ошибки - ошибки, повлекшие остановку технологического процесса или сбой в программе;
- б) умеренные ошибки - ошибки, вызывающие неудобство в работе или оказывающие отрицательное влияние на производительность или безопасность системы, а также ограничивающие функциональность системы (без прерывания технологического процесса) в рамках технического задания;
- с) ошибки процесса проектирования - ошибки или замечания, требующие расширения или изменения функциональности программной системы, выходящие за пределы технического задания.

После анализа проблем возможны следующие решения:

- а) ошибки первой группы устраняются до передачи в опытную эксплуатацию, для чего выделяется дополнительное время и корректируется календарный план. После устранения ошибок первой группы программный продукт проходит повторное квалификационное тестирование;
- б) ошибки второй группы устраняются во время опытной эксплуатации;
- с) замечания и ошибки третьей группы анализируются поставщиком и разработчиком совместно с заказчиком. По каждому пункту этой группы принимается отдельное решение, которое оформляется совместным протоколом заседания. На основании совместного решения заключается новый договор (контракт) или дополняется существующий. Модификации программной системы в рамках нового договора (дополнения) должны проходить полный жизненный цикл согласно настоящему техническому регламенту.

В случае отсутствия критических ошибок и замечаний, не позволяющих начать этап внедрения, разработчик передает программную систему заказчику для проведения опытной эксплуатации. При этом составляется акт передачи в опытную эксплуатацию. Форма акта приведена в приложении 5.

4.10.8.2. Результаты процесса проверки

В результате успешной реализации процесса проверки:

- а) осуществлено планирование проверки;
- б) разработаны проверочные требования, условия тестирования и тесты;
- с) осуществлена проверка;
- д) предоставлена аналитическая информация о несоответствиях для проведения корректирующих действий;
- е) заключен новый договор или предоставлены доказательства, что продукт соответствует системным требованиям;
- ф) оформлена передача программного продукта на этап внедрения.

4.10.8.3. Деятельности процесса проверки

Деятельности данного процесса показаны на рисунке 13.



Рисунок 13. Деятельности процесса проверки

4.10.9. Процесс утверждения

4.10.9.1. Цель процесса утверждения

Цель процесса утверждения заключается в предоставлении объективного доказательства, что услуги, предоставляемые разработанным программным продуктом или любым программным элементом, удовлетворяют утвержденным требованиям к программной системе и потребностям заинтересованных лиц.

Действия процесса утверждения обычно инициируют проведение опытной эксплуатации. Опытная эксплуатация программного продукта предоставляет информацию о поведении системы в реальной обстановке эксплуатации с использованием реальных данных. Опытная эксплуатация необходима для ознакомления пользователя с внедряемым программным продуктом и выявления отклонений в его работе.

В процессе утверждения выполняется сравнительная оценка и подтверждается, что были правильно реализованы требования заинтересованных лиц к программной системе. При выявлении отклонений они регистрируются и осуществляются корректирующие действия.

Все отклонения в работе и замечания (в рамках технического задания), выявленные во время опытной эксплуатации, устраняются разработчиком до окончания опытной эксплуатации. В случае необходимости корректируется календарный план.

Утверждение программного продукта подтверждается заинтересованными лицами путем подписания акта о завершении разработки. Форма акта - согласно приложению 6.

Утверждение акта должно производиться сторонами согласно договорным процессам. На основании акта о завершении разработки начинается деятельность по развертыванию программного продукта в местах его функционирования.

4.10.9.2. Результаты процесса утверждения

В результате успешной реализации процесса утверждения:

- a) разработан план проведения опытной эксплуатации;
- b) устранены выявленные несоответствия;
- c) подтверждено наличие программных услуг, требуемых заинтересованными лицами;
- d) в отчетности отражены результаты процесса утверждения;
- e) программный продукт передан для развертывания или отправлен на доработку.

4.10.9.3. Деятельности процесса утверждения

Деятельности данного процесса показаны на рисунке 14.



Рисунок 14. Деятельности процесса утверждения

4.10.10. Процесс перехода

4.10.10.1 Цель процесса перехода

Цель процесса перехода заключается в обеспечении способности программного продукта предоставлять услуги, предусмотренные требованиями заказчика. Этим процессом осуществляются развертывание утвержденного программного продукта в местах его функционирования и запуск АИС в рамках, оговоренных в договоре (контракте). Календарный план процесса перехода разрабатывают заинтересованные стороны и утверждает заказчик. Форма плана приведена в приложении 2. Работы этапа перехода могут оговариваться отдельным договором (контрактом).

Возможно поэтапное развертывание программного продукта. Например, в ограниченных местах его функционирования для проведения опытной эксплуатации или по мере развития АИС в территориальных подразделениях.

Во время процесса перехода возможна одновременная работа существующей и внедряемой программных систем. В этом случае необходимо обеспечить переход на новый программный продукт и, по требованию заказчика, согласно договору (плану работ) осуществить перенос данных, при этом обеспечивая их целостность.

Процесс перехода завершается выполнением разработчиком договорных обязательств, но развертывание может продолжаться все время эксплуатации программного продукта. Завершение процесса перехода подтверждается актом передачи в эксплуатацию программного продукта. Утверждение акта должно происходить согласно договорным обязательствам. Форма акта - согласно приложению 7.

4.10.10.2. Результаты процесса перехода

В результате успешной реализации процесса перехода:

- a) разработан календарный план процесса перехода;
- b) программный продукт развернут в местах его функционирования;
- c) программная система утверждена в соответствии с требованиями заказчика;
- d) зарегистрирована конфигурация установленной системы;
- e) осуществлено конвертирование или перенос данных;
- f) факт завершения процесса подтвержден актом передачи в эксплуатацию.

4.10.10.3. Деятельности процесса перехода

Деятельности данного процесса показаны на рисунке 15.



Рисунок 15. Деятельности процесса перехода

4.10.11. Процесс эксплуатации

4.10.11.1. Цель процесса эксплуатации

Цель процесса эксплуатации заключается в использовании программного продукта для получения услуг и предоставлении операционной поддержки пользователям. Начало эксплуатации устанавливается пользователем на основании акта ввода в эксплуатацию. Форма акта - согласно приложению 1. За действия в рамках данного процесса отвечает пользователь.

Пользователь управляет процессом эксплуатации на уровне проекта, создает инфраструктуру этого процесса, приспособливает ее к требованиям проекта.

Процессом эксплуатации пользователь контролирует получение услуг от программного продукта, оценивает взаимодействие непосредственных пользователей и программной системы, а также оказывает консультации непосредственным пользователям и осуществляет подготовку кадров. Для поддержания процесса получения программных

услуг проблемы, выявленные в ходе эксплуатации, должны регистрироваться и анализироваться. После анализа пользователь препровождает запросы непосредственного пользователя к сопроводителю программного продукта. Предложения и запросы оформляются в виде предложения о модификации или сообщения о проблеме. Форма этих документов - согласно приложению 1. Все запросы должны быть прослежены вплоть до их завершения. Взаимодействие пользователя и сопроводителя должно быть определено процессами договора.

4.10.11.2. Результаты процесса эксплуатации

В результате успешной реализации процесса эксплуатации:

- a) определены и применяются процедуры эксплуатации;
- b) предоставляются программные услуги, удовлетворяющие потребностям заинтересованного лица;
- c) успешно выполняются запросы на проведение утвержденных корректирующих действий;
- d) удовлетворены интересы пользователей.

4.10.11.3. Деятельности процесса эксплуатации

Деятельности данного процесса показаны на рисунке 16.



Рисунок 16. Деятельности процесса эксплуатации

4.10.12. Процесс сопровождения

4.10.12.1. Цель процесса сопровождения

Цель процесса сопровождения заключается в поддержании способности программной системы предоставлять услуги, а также модификации программного продукта при сохранении его целостности. Этим процессом контролируется работа программного продукта, регистрируются проблемы для анализа в журнале регистрации обращений, предпринимаются предупреждающие и корректирующие действия, а также действия по адаптации и усовершенствованию программного продукта. Перечень граф журнала - согласно приложению 1.

Процесс сопровождения заключается в модификации (изменении) текста или настроек программы и соответствующих документов вследствие обнаруженных проблем (несоответствий) или необходимости усовершенствования программной системы.

Управляет процессом сопровождения сопроводитель программного продукта на уровне проекта. Он должен организовать управление процессом, в рамках договорных

отношений создать инфраструктуру процесса и приспособить процесс к требованиям конкретного проекта.

Данный процесс включает следующие действия:

- подготовка процесса сопровождения;
- анализ проблем и изменений;
- внесение изменений;
- проверка и приемка при сопровождении;
- переход (перенос).

4.10.12.2. Результаты процесса сопровождения

В результате успешной реализации процесса сопровождения:

- а) разрабатываются концепция и план сопровождения;
- б) определены ограничения обслуживания, влияющие на сопровождение;
- с) приобретаются сменные системные элементы;
- д) поддерживаются услуги, удовлетворяющие требованиям заинтересованного лица;
- е) проводится анализ проблемы и принимается решение;
- ф) проводятся модификации программного продукта с последующим тестированием и вводом в эксплуатацию.

Структура концепции сопровождения программного продукта - согласно приложению 8. Структура плана сопровождения - согласно приложению 9.

4.10.12.3. Деятельности процесса сопровождения

Деятельности данного процесса показаны на рисунке 17.



Рисунок 17. Деятельности процесса сопровождения

4.10.13. Процесс изъятия

4.10.13.1. Цель процесса изъятия

Цель процесса изъятия заключается в завершении существования программной системы. Этим процессом производится вывод программного продукта из эксплуатации, разборка и удаление программной системы и ее элементов. Изъятие программного продукта из эксплуатации также может быть инициировано окончанием срока лицензии на эксплуатацию данного продукта или по просьбе владельца. За действия процесса изъятия отвечает сопроводитель.

Если изъятие инициируется владельцем, то необходимо проведение анализа, подтверждающего решение об изъятии программного продукта из эксплуатации. Как правило, подобное решение должно быть экономически обосновано.

Сопроводитель, выполняющий изъятие программного продукта из эксплуатации, должен решить следующие задачи: разработать календарный план изъятия из эксплуатации, уведомить пользователей и всех заинтересованных субъектов об изъятии программного продукта из эксплуатации и архивировать соответствующие данные. Форма календарного плана изъятия - в соответствии с приложением 2.

Владелец определяет способ и место хранения выведенных из эксплуатации программных элементов и данных.

Процесс изъятия завершается составлением акта о списании программного продукта согласно приложению 1.

4.10.13.2. Результаты процесса изъятия

В результате успешной реализации процесса изъятия:

- a) разработан календарный план изъятия из эксплуатации;
- b) заинтересованным сторонам направлены уведомления о намерениях по изъятию из эксплуатации;
- c) уничтожены или помещены на хранение программные элементы;
- d) сохранена информация, полученная в результате создания и эксплуатации системы;
- e) факт изъятия подтвержден актом о списании.

4.10.13.3. Деятельности процесса изъятия

Деятельности данного процесса показаны на рисунке 18.

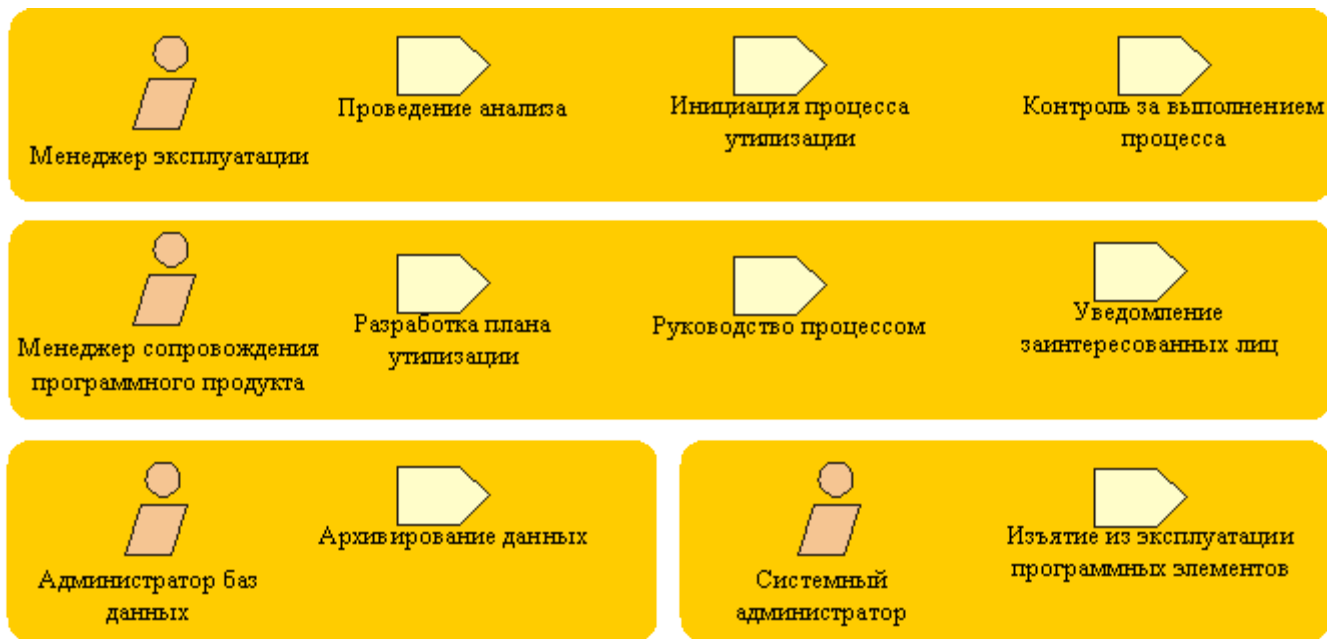


Рисунок 18. Деятельности процесса изъятия

5. МЕТОДЫ РАЗРАБОТКИ ПРОГРАММНОЙ СИСТЕМЫ

В зависимости от сложности программной системы, степени участия заказчика и установленных сроков реализации проекта разработчик выбирает метод разработки. Основным методом разработки программной системы является итерационный, структура которого приведена на рисунке 19.

Итерационный метод разработки программной системы - это метод, при котором определение требований, анализ, проектирование, программирование, интеграция и проверка реализуются постепенно в ходе разработки программной системы. Итерационный метод подразумевает тесное сотрудничество заказчика и разработчика во время создания программной системы. Итерационный метод характеризуется быстрым вводом в эксплуатацию программного продукта с ограниченными (основными) возможностями, которые определяет сам заказчик, с постепенным добавлением остальных функций, не прерывая эксплуатацию программного продукта. Итерационный метод должен быть основным методом разработки программных систем.

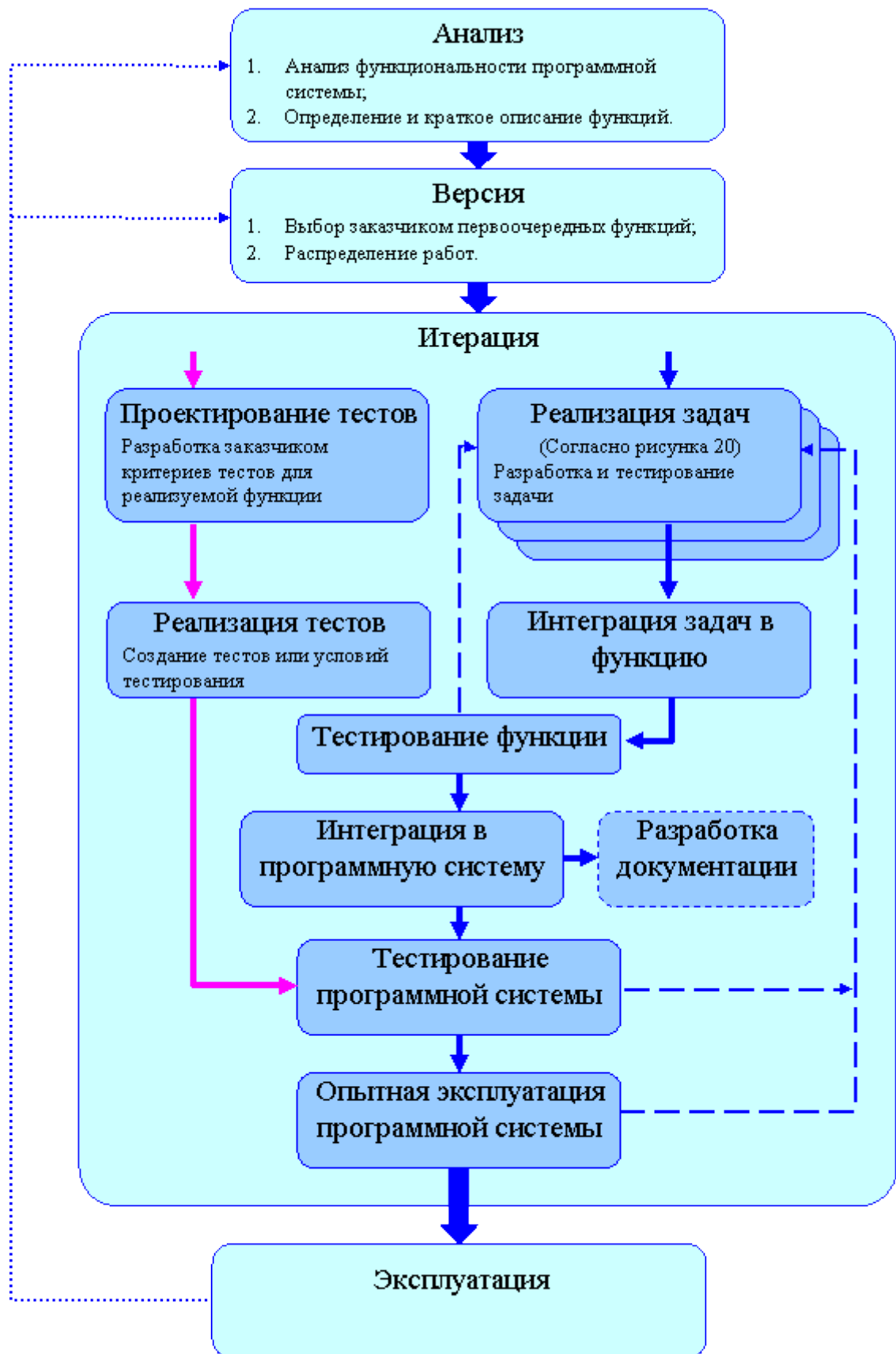


Рисунок 19. Структура итерационного метода разработки программной системы

Фазами разработки программой системы при итерационном методе разработки являются:

- а) анализ;

- b) версия;
- c) итерация;
- d) эксплуатация.

Фазы на каждом витке разработки программной системы повторяются.

5.1. Анализ

Разработчик совместно с заказчиком проводит анализ разрабатываемой программной системы, определяет, какими возможностями и функциями она должна обладать по мере ее реализации.

На основании проведенного анализа разработчик создает модель реализации в виде «Списка функций» с их кратким описанием в соответствии с приложением 10. Список может пополняться и уточняться по мере реализации программной системы.

Каждый элемент списка должен быть определен как законченная и самостоятельная функция (элемент), которую можно протестировать и оценить ее работоспособность. На основании общих оценок определяется время, необходимое на разработку, тестирование и запуск в эксплуатацию описываемой функции.

5.2. Версия

При инкрементном подходе заказчик из списка функций выбирает одну или небольшое множество первоочередных функций, возможно, логически связанных между собой, которые разрабатываются и запускаются в эксплуатацию в первую очередь. Эта группа функций объединяется в версию. Остальные функции реализуются позже, в следующих версиях.

Заказчик определяет очередную инкрементную версию программной системы, выбирая, с его точки зрения, наиболее ценные функции. Ценность функций определяется рисками, материальными и временными затратами на их реализацию.

На основании выбора, сделанного заказчиком, распределяются работы между конкретными исполнителями и составляется календарный план реализации версии, согласно приложению 1, или дополняется календарный план реализации.

5.3. Итерация

Цель каждой итерации - запустить в эксплуатацию одну или несколько новых протестированных и готовых к использованию функций.

Итерации являются инкрементными в соответствии с той функцией, которую они выполняют. Каждая итерация добавляет очередные конструкции к возможностям программной системы, реализованными во время предыдущих итераций.

На каждой итерации некоторая часть существующего кода может заново создаваться с целью сделать его более гибким. При этом может возникнуть необходимость в проведении реорганизации.

Реорганизацию следует выполнять в следующих случаях:

- a) если при расширении функциональности программной системы обнаруживаются проблемы с существующим кодом или методом реализации;
- b) если код или структура программной системы становятся трудными для понимания.

Во время итерации заказчик разрабатывает критерии тестов для реализуемой функции. На основании этого разработчик реализует тесты в виде готовых модулей или создает необходимые условия для тестирования. К концу итерации тесты должны функционировать.

Итерация завершается опытной эксплуатацией программной системы, которая может проводиться на всех или ограниченном количестве рабочих мест пользователей. При выполнении опытной эксплуатации используются реальные данные. Информация, полученная и обработанная на стадии опытной эксплуатации, по решению заказчика может удаляться или использоваться в дальнейшем.

5.3.1. Задача

Разработчик разбивает функции на задачи - программные элементы, реализуемые в минимальные сроки. Задачи распределяются между конкретными разработчиками. В зависимости от сложности задач время на их разработку может перераспределяться между исполнителями.

По мере завершения задач их код интегрируется в законченную функцию и тестируется разработчиком. При отрицательном результате теста функция вновь разбивается на задачи и проверяется их реализация.

Если тестирование отдельной функции прошло успешно, код интегрируется в общую систему и тестируется на базе описанных заказчиком тестов или условий тестирования. В случае если тестирование не прошло успешно, код отправляется на доработку. При тестировании всей программной системы должны работать все тесты, разработанные ранее для других итераций.

Реализация задач осуществляется в процессе итерации в соответствии с рисунком 20. Одновременно могут разрабатываться несколько задач. Цель при выполнении задачи - сконцентрироваться на деталях и реализовать конкретную проблему.

После уточнения предмета и методов реализации, задача реализуется в коде и параллельно создаются тесты или условия тестирования задачи.

Готовая задача тестируется разработчиками при помощи созданных тестов или условий тестирования.



Рисунок 20. Структура реализации задачи

5.4. Эксплуатация

Программный продукт при итерационном методе разработки может находиться в эксплуатации сразу после завершения реализации первой версии программного продукта.

Каждая следующая версия добавляет функциональность программного продукта, вплоть до полной реализации требований заказчика. После этого разработка завершается и начинается полнофункциональная эксплуатация программного продукта.

6. ДОКУМЕНТАЦИЯ

В процессе жизненного цикла программной системы с каждым процессом связаны артефакты, которые либо подаются на вход, либо получаются на выходе данного процесса. Полученные артефакты используются как исходные данные для последующей деятельности. Они содержат справочные сведения о проекте или выступают в роли поставляемых по контракту составляющих.

Одним из основных видов артефактов является документация проекта.

Документация - это результат записи информации, полученной в ходе действий или процессов жизненного цикла программной системы.

Содержание разрабатываемых документов должно соответствовать приложениям настоящего технического регламента.

Обозначения документам присваиваются в соответствии с рисунком 21.

Оформление документов должно соответствовать требованиям стандарта Молдовы SM 1-5:2005 «Принципы и методология стандартизации. Построение, изложение и содержание стандартов Молдовы».



Рисунок 21. Структура обозначения документов

Пояснения к рисунку 21:

а) идентификаторы АИС и подсистемы присваиваются Министерством информационного развития и представляют собой их учетный номер в Государственном регистре информационных ресурсов и систем;

б) номер элемента в программной системе присваивается разработчиком;

с) код документа присваивается согласно приложению 1;

д) номер части документа присваивается разработчиком;

е) версия редакции документа присваивается разработчиком;

ф) номер дополнения присваивается разработчиком.

Документы могут быть предоставлены или распространены на бумаге или в электронном виде.

Перечень документов, необходимых для программного продукта, определяется на этапе разработки, в процессе определения и анализа требований заказчика.

7. ЗАКЛЮЧИТЕЛЬНЫЕ И ПЕРЕХОДНЫЕ ПОЛОЖЕНИЯ

Настоящий технический регламент применяется при разработке новых и реинжиниринге действующих автоматизированных информационных систем. После

внедрения настоящего технического регламента, будут пересмотрены национальные стандарты в области информационных технологий.

Настоящий технический регламент вступает в силу по истечении тридцати дней со дня опубликования.

До вступления в действие настоящего технического регламента требования национальных стандартов на процессы жизненного цикла программного обеспечения сохраняют обязательный характер.

Приложение 1
к RT 38370656 - 002:2006

ПЕРЕЧЕНЬ И КРАТКОЕ СОДЕРЖАНИЕ ДОКУМЕНТОВ ТЕХНИЧЕСКИХ ПРОЦЕССОВ ЖИЗНЕННОГО ЦИКЛА ПРОГРАММНОЙ СИСТЕМЫ

Таблица 1

№ п/п	Код документа	Наименование документа	Краткое содержание документа
1	P1	План руководства проектом	1 Стороны, участвующие в разработке и их полномочные представители 2 Полномочие и ответственность сторон, включая внешние организации 3 Руководящие лица проекта. Управление разработкой и внедрением проекта 4 Управление субподрядчиком, если таковой намечается* 5 Комплектование кадрами и материальными ресурсами* 6 Определение основных процессов по этапам. Сроки разработки, внедрения, начала эксплуатации* 7 Состав проекта, и определение его сложности. 8 Порядок квалификационного тестирования программной системы* 9 Санкции, необходимые спецификации, имущественные, гарантийные и лицензионные права, юридическая основа проекта* 10 Обучение персонала*
2	P2	Календарный план реализации проекта	Согласно приложению 2. ПРИМЕЧАНИЕ: - план согласовывается с заказчиком и утверждается поставщиком; - может выполняться в Microsoft Project
3	C1	Концепция системы	Согласно приложению 3
4	PВ	Бизнес-предложение	1 Общие сведения, назначение и цели создания программной системы 2 Нормативные ссылки* 3 Терминология и сокращения* 4 Описание области автоматизации 5 Бизнес-модель области автоматизации 6 Функциональные возможности системы 7 Архитектура системы и взаимодействие с внешними системами* 8 Перспективы развития системы*
5	ST	Техническое задание	Согласно приложению 4
6	PT	Технический проект	1 Введение 2 Ссылки* 3 Терминология и сокращения* 4 Описание технологии и методов реализации 5 Описание архитектуры программной системы и ее

			<p>элементов</p> <p>6 Описание модели взаимодействия элементов</p> <p>7 Описание структур данных, классов, интерфейсов, артефактов</p> <p>8 Описание алгоритмов обработки информации*</p> <p>9 Описание структуры пользовательского интерфейса*</p> <p>10 Информационное наполнение справочников*</p>
7	IE	Инструкция по эксплуатации (программного продукта или его элементов)	<p>1 Назначение</p> <p>2 Терминология и сокращения*</p> <p>3 Описание экранных форм и диалоговых окон*</p> <p>4 Описание запросов, выходных документов, отчетов и формируемых графиков*</p> <p>5 Описание порядка работы</p>
8	IA	Руководство администратора (программного продукта или его элементов)	<p>1 Введение</p> <p>2 Терминология и сокращения*</p> <p>3 Требования к аппаратному обеспечению</p> <p>4 Комплект поставки. Необходимые дополнительные модули, библиотеки и ПО сторонних производителей, их допустимые версии</p> <p>5 Инструкция по установке</p> <p>6 Описание настроек*</p> <p>7 Перечень основных сообщений и действия по ним*</p>
9	-	Протокол тестирования	<p>1 Вид тестирования</p> <p>2 Дата проведения тестирования</p> <p>3 Место проведения тестирования*</p> <p>4 Наименование программного продукта или элемента</p> <p>5 Состав проводимых испытаний или ссылка на план тестирования*</p> <p>6 Описание ошибок (проблем)</p> <p>7 Группа ошибки</p> <p>8 Место проявления ошибки</p> <p>9 Отметка об исправлении*</p> <p>10 Примечания*</p> <p>11 Подписи лиц, участвовавших в тестировании</p>
10	P4	Календарный план квалификационного тестирования	<p>Согласно приложению 2.</p> <p>ПРИМЕЧАНИЕ:</p> <p>- план согласовывается с заказчиком и утверждается поставщиком;</p> <p>- может выполняться в Microsoft Project</p>
11	-	Акт передачи в опытную эксплуатацию	<p>Согласно приложению 5.</p> <p>ПРИМЕЧАНИЕ:</p> <p>- акт разрабатывается поставщиком и утверждается заказчиком или согласно договорным обязательствам</p>
12	-	Акт о завершении разработки	<p>Согласно приложению 6.</p> <p>ПРИМЕЧАНИЕ:</p> <p>- акт разрабатывается поставщиком и утверждается заказчиком или согласно договорным обязательствам</p>
13	P6	Календарный план процесса перехода	<p>Согласно приложению 2.</p> <p>ПРИМЕЧАНИЕ:</p> <p>- план разрабатывается разработчиком и утверждается заказчиком;</p> <p>- может выполняться в Microsoft Project</p>
14	-	Акт передачи в эксплуатацию	<p>Согласно приложению 7.</p> <p>ПРИМЕЧАНИЕ:</p> <p>- акт разрабатывается поставщиком и утверждается заказчиком или согласно договорным обязательствам</p>
15	-	Акт ввода в эксплуатацию	<p>Составляется заказчиком или пользователем согласно внутриведомственным требованиям делопроизводства</p>
16	-	Предложение о модификации	<p>В электронной форме - согласно требованиям программного обеспечения.</p> <p>В бумажной форме - в произвольной форме с набором</p>

			данных, достаточных для принятия решения и конкретных действий
17	-	Сообщение о проблеме	В электронной форме - согласно требованиям программного обеспечения. В бумажной форме - в произвольной форме с набором данных, достаточных для принятия решения и конкретных действий
18	-	Журнал регистрации обращений	В электронной форме - согласно требованиям программного обеспечения. В бумажной форме: 1 регистрационный номер обращения; 2 дата и время обращения; 3 место, дата и время обнаружения проблемы*; 4 краткое описание проблемы или предложения о модификации; 5 должность и фамилия обратившегося; 6 принятое решение; 7 кому доложена или перенаправлена проблема; 8 отметка о закрытии проблемы
19	C5	Концепция сопровождения	Согласно приложению 8
20	P7	План сопровождения	Согласно приложению 9
21	P9	Календарный план изъятия из эксплуатации	Согласно приложению 2. ПРИМЕЧАНИЕ: - план разрабатывается сопровождающим и утверждается заказчиком; - может выполняться в Microsoft Project
22	-	Акт о списании	Составляется заказчиком или пользователем согласно внутриведомственным требованиям делопроизводства
23	-	Список функций программной системы	Согласно приложению 10. ПРИМЕЧАНИЕ: - список функций подготавливается разработчиком и утверждается заказчиком
24	P3	Календарный план реализации версии	Согласно приложению 2. ПРИМЕЧАНИЕ: - план согласовывается с заказчиком и утверждается разработчиком; - может выполняться в Microsoft Project

* Наличие пунктов зависит от состава и сложности программной системы, а также ее функциональности.

Приложение 2
к RT 38370656 - 002:2006

ФОРМА КАЛЕНДАРНОГО ПЛАНА

УТВЕРЖДАЮ

(должность, подпись, Ф.И.О.)
« ____ » _____ 200__ г.

КАЛЕНДАРНЫЙ ПЛАН

(наименование этапа или процесса)

(наименование программной системы)

№	Наименование	Сроки начала	Чем заканчивается каждый этап или деятельность	Исполнитель
---	--------------	--------------	--	-------------

	этапов и работ	и окончания работ	(техническая информация, промежуточный отчет, принципиальная схема, макет, экспериментальная установка, окончательный отчет и т.д.)	
1	2	3	4	5

Примечание:

—

СОГЛАСОВАНО

(должность, подпись, Ф.И.О.)

« _____ » _____ 200__ г.

Приложение 3
к RT 38370656 - 002:2006

СТРУКТУРА КОНЦЕПЦИИ СИСТЕМЫ

1. ОБЩИЕ ПОЛОЖЕНИЯ

Концепция является первоначальным документом, разрабатываемым при создании системы, содержащим результаты выполнения предпроектных научно-исследовательских и/или опытных работ и служит основой для дальнейшей разработки технической документации.

В зависимости от поставленных целей концепция может быть представлена заказчику как основная или рамочная. Рамочная концепция разрабатывается в случае, когда описываемая система состоит из множества самостоятельных или взаимосвязанных систем, для которых впоследствии будут разработаны основные концепции. Рамочная концепция отличается от основной более кратким изложением материала либо отсутствием отдельных разделов или подразделов, являющимися обязательными для основной концепции.

Основное назначение концепции заключается в предоставлении заказчику общего видения системы, выполняемых ею функций, описания информационного и правового пространства и взаимодействия с другими информационными системами.

2. ТРЕБОВАНИЯ К СТРУКТУРЕ И СОДЕРЖАНИЮ ОСНОВНОЙ КОНЦЕПЦИИ

2.1. Структура основной концепции

Основная концепция должна состоять из следующих обязательных разделов:

- a) «Введение»;
- b) «Общие положения»;
- c) «Нормативно-правовое пространство функционирования системы»;
- d) «Функциональное пространство системы»;
- e) «Организационная структура системы»;
- f) «Документы системы»;
- g) «Информационное пространство системы»;
- h) «Технологическое пространство системы»;
- i) «Обеспечение информационной безопасности системы»;

ж) «Заключение».

При изложении разделов концепции допускается ссылка на концепции, утвержденные ранее.

2.2. Краткое содержание разделов основной концепции

2.2.1. Введение

Введение должно содержать информацию об области деятельности, в которой будет функционировать система, анализ уровня информатизации данной области, а также причины создания или модификации системы.

2.2.2. Общие положения

Содержание данного раздела концепции должно включать в себя:

- название системы (полное, краткое, аббревиатура);
- определение системы;
- место системы в Едином информационном пространстве;
- основные понятия;
- назначение системы;
- цели создания системы;
- основные принципы создания системы;
- основные задачи, которые должны быть решены при эксплуатации системы.

2.2.3. Нормативно-правовое пространство функционирования системы

Раздел должен содержать перечень основных нормативно-правовых актов (законов, указов Президента, постановлений Правительства и т.д.), относящихся к области деятельности, в которой будет функционировать система, а также нормативно-правовые акты, регламентирующие деятельность и взаимоотношения в области информатизации.

При необходимости в разделе приводятся предложения по актуализации существующей нормативно-правовой базы и разработке недостающих нормативно-правовых актов, необходимых для создания и эксплуатации системы.

В разделе должен быть приведен перечень основных стандартов, используемых при разработке и создании системы или ссылки на другие документы, содержащие такие перечни.

2.2.4. Функциональное пространство системы

В разделе описываются функции, которые должна выполнять разрабатываемая система. В первую очередь должны быть определены:

- базовые функции системы;
- базовые функциональные контуры;
- взаимосвязи, возникающие при реализации функций системы;
- результаты функционирования системы.

При необходимости может быть приведена краткая характеристика каждой функции.

Целесообразно проиллюстрировать функционирование системы функциональной схемой, на которой графически было бы отображено взаимодействие всех компонентов системы.

2.2.5. Организационная структура системы

В данном разделе приводится перечень ведомств и их подразделений, задействованных в формировании и управлении информационными ресурсами и краткое описание межведомственного взаимодействия. Для каждого ведомства указываются основные функции, связанные с созданием и эксплуатацией системы.

2.2.6. Документы системы

В разделе должен быть приведен перечень основных документов, используемых в системе. Все документы должны быть разделены на следующие категории:

- входные документы, являющиеся основанием для внесения данных в систему (по возможности необходимо привести ссылки на другие системы, в которых они изготавливаются);
- выходные документы, изготавливаемые в результате функционирования системы;

- технологические документы (заявления, анкеты, реестры и т.д.).

Если при создании системы предполагается введение новых документов или изменение образцов существующих, желательно привести их краткое описание и образец.

2.2.7. Информационное пространство системы

Совокупность объектов, их атрибутов и сценариев определяют информационное пространство системы.

Данный раздел является ключевым для определения и понимания системы и включает в себя следующие подразделы:

- перечень информационных объектов, учитываемых (контролируемых) в системе;
- структуру идентификатора для каждого информационного объекта;
- сценарий поведения каждого объекта, учитываемый в системе;
- данные (или атрибуты объектов), содержащиеся в системе;
- классификаторы;
- информационные потоки;
- интеграция с другими информационными системами.

Определение состава информационных объектов, их идентификации и сценариев поведения каждого объекта является основной задачей разработчика концепции.

Информационный объект должен характеризоваться тремя основными особенностями:

- уникальностью (уникальность объекта означает наличие уникального идентификатора, отличающего данный объект от других подобных ему объектов);
- состоянием (состояние объекта описывается набором атрибутов, описывающих изменяемые свойства объекта, учитываемые в системе);
- поведением (поведение объекта определяется перечнем событий, которые происходят с объектом и учитываются в данной системе).

Необходимо определить, является ли объект собственным (т.е. он первоначально учитывается и идентифицируется в данной системе) или заимствованным (т.е. берется вместе с идентификатором из другой системы, и в этом случае не допускается изменение идентификатора, а набор атрибутов и перечень событий могут частично заимствоваться и/или дополняться). Затем описывается структура идентификатора для каждого объекта, как для собственного, так и для заимствованного.

Состав информационных объектов должен определяться назначением системы. Далее определяется перечень событий, учитываемых в системе для каждого объекта, при этом особое внимание должно быть уделено описанию первоначальной постановки на учет и условий, при которых осуществляется идентификация вновь созданного объекта, а также снятия с учета объектов системы.

Следующим шагом является определение перечня атрибутов объектов, хранящихся в системе и в совокупности определяющих информационное пространство системы. Допускается указание блоков данных без их детализации. После этого должны быть указаны основные классификаторы, используемые в системе, с подразделением на международные, национальные и внутрисистемные.

В заключительной части данного раздела концепции приводятся основные информационные потоки, циркулирующие в системе, с кратким описанием информационных систем (как существующих, так и проектируемых), которые предоставляют информацию, необходимую для обеспечения функционирования системы, и, в частности, для контроля качества информации. Дополнительно могут быть указаны основные системы, использующие информацию данной системы.

2.2.8. Технологическое пространство системы

Технологическое пространство определяет выбор и реализацию программно-технических решений для всех компонентов системы, включая вопросы информационной безопасности и обеспечения качества.

В данном разделе описывается инфраструктура системы. Раздел включает в себя следующие подразделы:

- уровни системы;
- информационно-телекоммуникационная сеть;
- программно-технические комплексы (ПТК) для каждого уровня.

При описании уровней системы должно быть указано их количество и месторасположение каждого уровня (в столице, районных центрах, коммунах и т.д.).

Описание информационно-телекоммуникационной сети включает в себя описание топологии сети и основных технических характеристик или ссылку на документ, содержащий указанные описания.

При описании программно-технического комплекса должно быть указано местоположение ПТК (организация), его назначение, состав технических средств и основных программных продуктов (операционные системы, системы управления базами данных, сервера приложений, средства разработчиков, пользовательские приложения).

2.2.9. Обеспечение информационной безопасности системы

Информационная безопасность системы достигается применением соответствующего набора средств контроля, которыми могут быть политика, практические меры, процедуры, организационные структуры и программные функции. Необходимо определить эти средства контроля, чтобы обеспечить достижение специфических целей организации в области безопасности.

Обеспечение информационной безопасности системы принято рассматривать по следующим направлениям:

- общие требования к информационной безопасности системы;
- основные меры по обеспечению информационной безопасности системы.

2.2.10. Заключение

Заключение содержит информацию о планируемых результатах внедрения системы и о первоочередных мерах по ее созданию.

3. ПОРЯДОК РАЗРАБОТКИ, ОФОРМЛЕНИЯ, СОГЛАСОВАНИЯ И УТВЕРЖДЕНИЯ КОНЦЕПЦИИ

Разработка концепции включает в себя следующие деятельности:

- a) изучение нормативно-правовой базы функционирования данной области деятельности системы;
- b) изучение организационной структуры, осуществляющей управление данной областью;
- c) определение набора документов, циркулирующих в данной области;
- d) анализ уровня информатизации в данной области;
- e) определение функций системы;
- f) определение состава информационных объектов, их идентификаторов, сценариев и атрибутов;
- g) определение информационных потоков, их движения и вопросов интеграции с другими системами;
- h) разработка предложений по актуализации нормативно-правовой базы (при необходимости - разработка проектов новых нормативно-правовых актов), модификации организационной структуры и и/или создания новых организационных структур, созданию новых документов и/или новых образцов уже существующих документов;
- i) разработка предложений по технологической структуре системы, а также по вопросам информационной безопасности;
- j) оформление проекта концепции и подготовка документов, необходимых для ее согласования.

Оформление, согласование и утверждение концепции осуществляется в соответствии с:

- Законом Республики Молдова о нормативных актах Правительства и других органов центрального и местного публичного управления № 317-XV от 18.07.2003 г., „Monitorul Oficial al Republicii Moldova” № 208-210/783 от 03.10.2003 г.

- Постановлением Правительства о порядке проведения юридической экспертизы и государственной регистрации ведомственных нормативных актов №1104 от 28.11.1997 г., „Monitorul Oficial al Republicii Moldova” № 6-7/10 от 21.01.1998 г.

Приложение 4
к RT 38370656 - 002:2006

СТРУКТУРА ТЕХНИЧЕСКОГО ЗАДАНИЯ

1. ОБЩИЕ ПОЛОЖЕНИЯ

Техническое задание (ТЗ) является основным документом, определяющим требования заказчика к системе, в соответствии с которыми осуществляется разработка программного продукта.

ТЗ может разрабатываться на систему в целом и/или на ее составные части.

В ТЗ на систему, включающей группу взаимосвязанных подсистем, следует включать только требования, общие для группы подсистем. Специфические требования для отдельной подсистемы следует отражать в ТЗ на подсистему.

2. СТРУКТУРА ТЕХНИЧЕСКОГО ЗАДАНИЯ

ТЗ должно состоять из следующих разделов и подразделов:

- a) общие сведения;
- b) нормативные ссылки;
- c) терминология и сокращения;
- d) назначение системы;
- e) бизнес-модель объекта автоматизации:
 - основные процессы автоматизируемого объекта;
 - бизнес-роли;
 - услуги;
 - сценарии выполнения услуг;
- f) функциональные требования к системе:
 - функциональная модель системы;
 - требования к бизнес-функциям системы;
- g) требования к системе в целом:
 - требования по безопасности и защищенности;
 - требования по сохранности информации;
 - требования к аппаратному обеспечению и каналам связи;
 - надежность системы;
 - порядок испытания и приемки;
 - требования к документации.

В зависимости от вида, назначения, специфических особенностей и условий функционирования объекта автоматизации допускается вводить дополнительные разделы и подразделы, исключать или объединять разделы и подразделы ТЗ.

Структурная схема ТЗ приведена на рисунке 1.

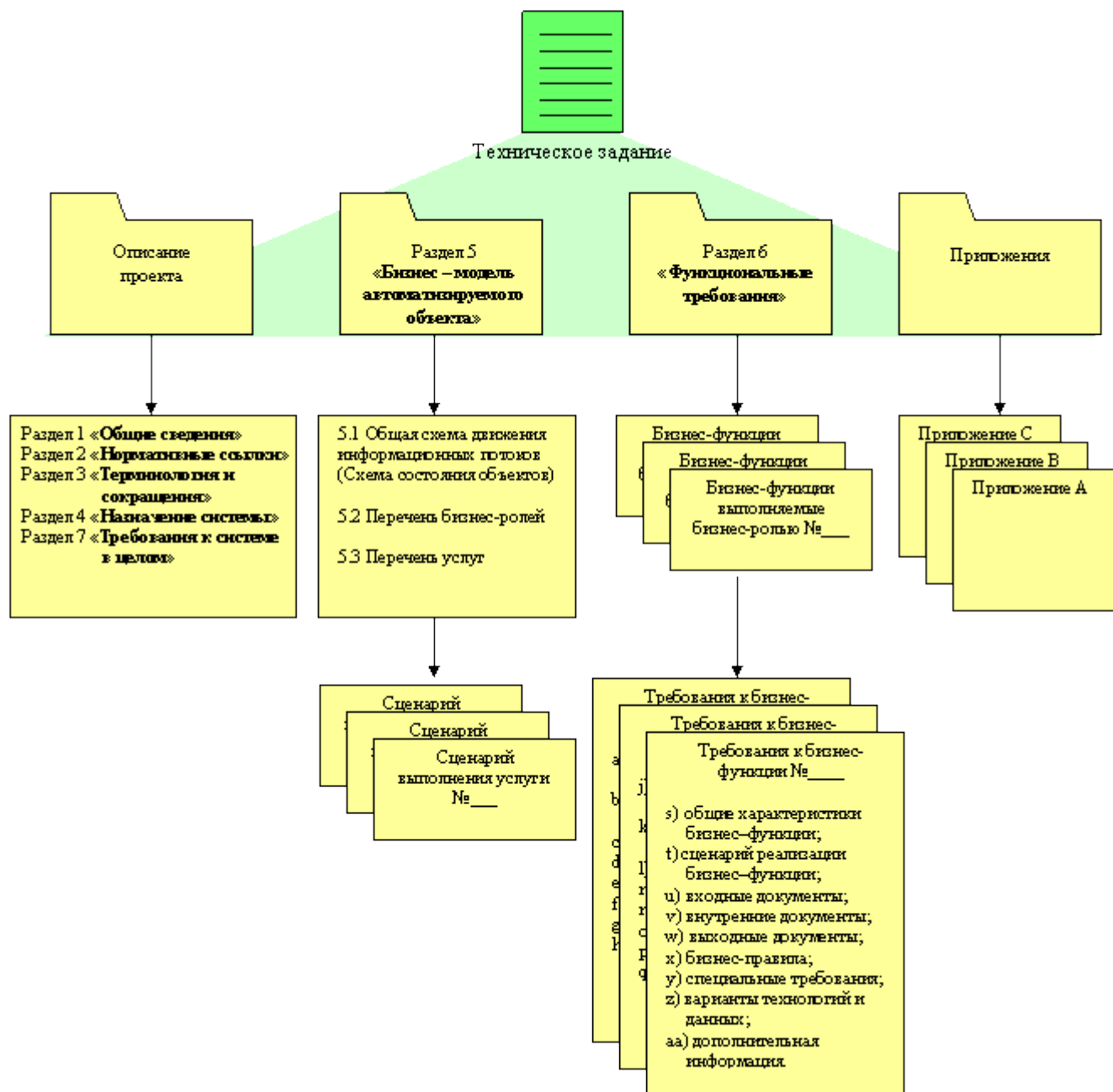


Рисунок 1. Структурная схема ТЗ

2.1. Раздел "Общие сведения"

В разделе "Общие сведения" указывают:

- полное наименование системы;
- перечень организационно-распорядительных документов, на основании которых была начата разработка системы.

ПРИМЕР

Подсистема «Сбор информации» (далее - подсистема «СИ») является составной частью автоматизированной информационной системы «Государственный регистр». Разработка подсистемы «СИ» осуществляется на основании ... (договора, распоряжения и т.д.).

2.2. Раздел "Нормативные ссылки"

В разделе "Нормативные ссылки" должен быть указан перечень нормативных документов, на которые даны ссылки в тексте ТЗ. Ссылки в тексте оформляются по примеру:

ПРИМЕР

Разработка подсистемы «СИ» осуществляется в соответствии с требованиями SF 00000001 - 001.

2.3. Раздел "Терминология и сокращения"

В разделе "Терминология и сокращения" приводят:

- перечень терминов и определений к ним;
- перечень сокращений и полных форм к ним.

2.4. Раздел "Назначение системы "

В разделе "Назначение системы " указывают:

- назначение разрабатываемой системы;
- перечень целей создания системы.

ПРИМЕР

Основным назначением системы «СИ» является автоматизация процесса сбора информации в территориальных подразделениях.

Подсистема «СИ» создается с целью ускорения и оптимизации процессов сбора и предварительной обработки информации, улучшения качества и повышения достоверности данных.

2.5. Раздел «Бизнес-модель объекта автоматизации»

В разделе «Бизнес-модель объекта автоматизации» приводят:

- основные процессы автоматизируемого объекта;
- перечень бизнес-ролей;
- перечень услуг, оказываемых системой;
- сценарии выполнения услуг.

2.5.1. Основные процессы автоматизируемого объекта

В описании основных процессов автоматизируемого объекта приводят общую схему движения информационных потоков и/или схему состояний объектов.

Общую схему движения информационных потоков приводят в виде диаграммы. Внешний вид схемы движения информационных потоков приведен на рисунке 2.

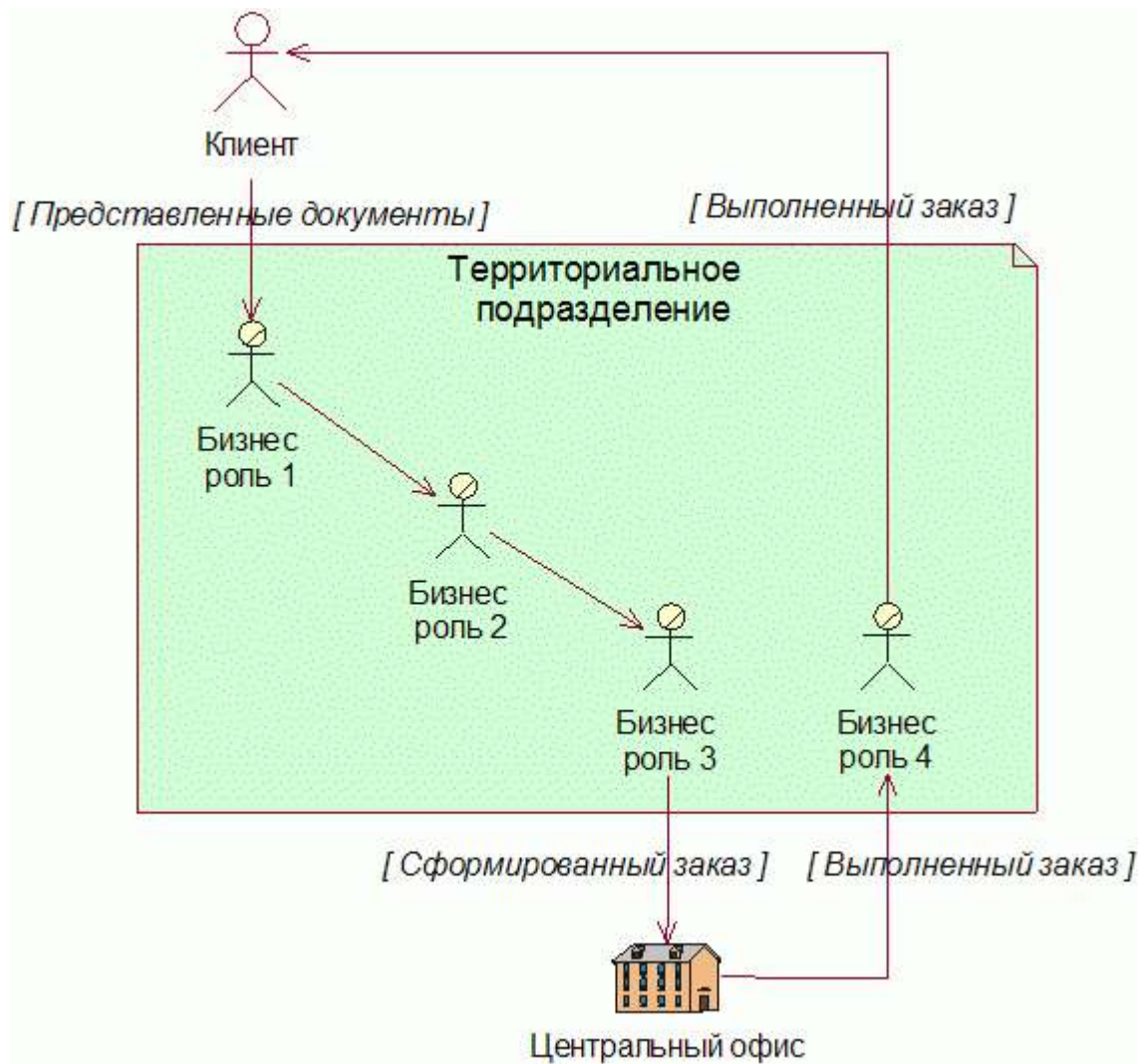

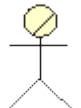
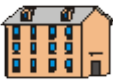
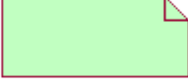


Рисунок 2. Схема движения информационных потоков

Графические стереотипы, используемые на диаграмме общей схемы движения информационных потоков приведены в таблице 1.

Таблица 1. Графические стереотипы

№	Стереотип	Описание (description)
1	 actor (роль)	Используется для отображения бизнес-ролей, находящихся за пределами области автоматизации, которые взаимодействуют с объектами автоматизации
2	 business actor (бизнес-роль)	Используется для отображения бизнес-ролей, находящихся внутри области автоматизации
3	 enterprise	Используется для отображения организационных объектов, расположенных вне области автоматизации, внутренняя деятельность которых не раскрывается

	(предприятие)	
4	 note (комментарий)	Используется для отображения объектов автоматизации. Объединяет бизнес-роли, находящиеся внутри объектов автоматизации

На диаграмме отображают объекты автоматизации, а также основные объекты, которые с ними взаимодействуют. Объекты автоматизации отображают внутри UML-стереотипа «note». Размещенные на диаграмме взаимодействующие объекты связывают стрелками, показывающими направление движения информационных потоков. Используя спецификацию, допускается указывать на стрелках тип передаваемой информации или передаваемый артефакт. Приводимую на стрелках информацию заключают в квадратные скобки.

Схему состояний объектов приводят в виде диаграммы. Внешний вид схемы состояний объектов приведен на рисунке 3.

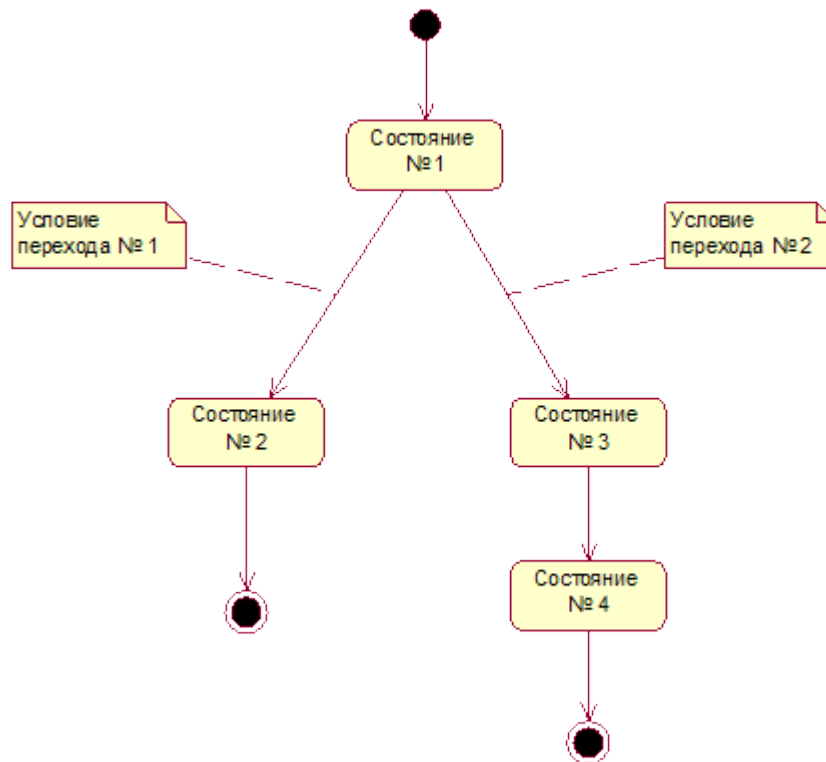






Рисунок 3. Схема состояний объекта

Графические стереотипы, используемые на схеме состояния объектов, приведены в таблице 2.

Таблица 2. Графические стереотипы

№	Стереотип	Описание
1	 start_state (начальное состояние)	Используется для отображения начального состояния объекта
2	 end_state (финальное состояние)	Используется для отображения финального состояния объекта

3	 state (состояние)	Используется для отображения одного из возможных состояний объекта
4	 note (комментарий)	Используется для отображения на диаграмме комментариев

На диаграмме отображают ключевые состояния рассматриваемого объекта. Оформление диаграммы - в соответствии с правилами построения диаграмм типа «Statechart».

2.5.2. Перечень бизнес-ролей приводят в виде таблицы или в виде диаграммы.

Перечень бизнес-ролей приводят в виде таблицы по форме в соответствии с таблицей 3.

Таблица 3. Перечень бизнес-ролей

№	Название объекта автоматизации	Название бизнес-роли	Примечание
1	Объект автоматизации № 1	Бизнес-роль № 1.1	Примечание 1
		Бизнес-роль № 1.2	
		Бизнес-роль № 1.3	Примечание 2
		Бизнес-роль № 1.4	
2	Объект автоматизации № 2	Бизнес-роль № 2.1	
		Бизнес-роль № 2.2	Примечание 3
		Бизнес-роль № 2.3	
3	...		

Внешний вид диаграммы бизнес-ролей представлен на рисунке 4.

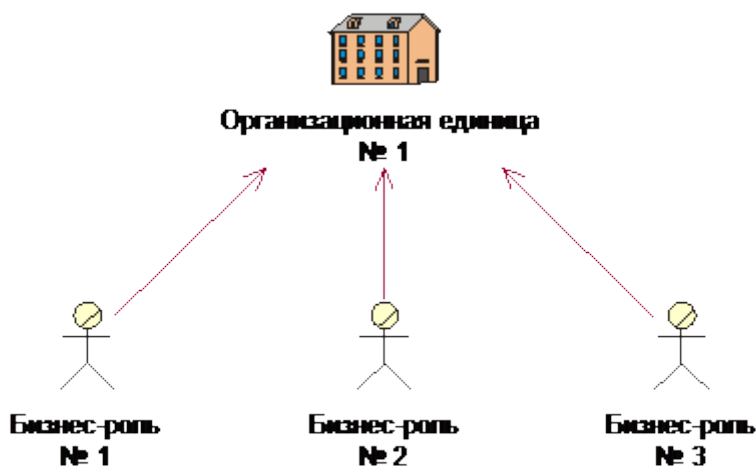
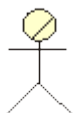



Рисунок 4. Диаграмма бизнес-ролей

Графические стереотипы, используемые на диаграмме бизнес-ролей, приведены в таблице 4.

Таблица 4. Графические стереотипы

№	Стереотип	Описание
1	 Role (бизнес-роль)	Используется для отображения бизнес-ролей
2	 enterprise (предприятие)	Используется для отображения организационных единиц

На диаграмме отображают организационные единицы и ключевые бизнес-роли, входящие в их состав. При распределении объектов на диаграмме, организационную единицу располагают сверху, а входящие в нее бизнес-роли и другие организационные единицы - снизу. Нахождение бизнес-роли внутри организационной единицы указывается с помощью стрелок. Направление стрелок должно быть от бизнес-роли к организационной единице.

2.5.3. Услуги

Перечень оказываемых услуг приводят в виде таблицы по форме в соответствии с таблицей 5.

Таблица 5. Перечень услуг

№	Название услуги	Примечание
1	Основная услуга № 1	Примечание 1
1.1	Дополнительная услуга № 1.1	
1.2	Дополнительная услуга № 1.2	Примечание 2
2	Основная услуга № 2	
2.1	Дополнительная услуга № 2.1	
2.2	Дополнительная услуга № 2.2	
2.3	Дополнительная услуга № 2.3	Примечание 3
3	Основная услуга № 3	
....		

При формировании заказа выбирают только одну из основных услуг. Дополнительно к выбранной основной услуге можно выбрать соответствующие ей дополнительные услуги.

2.5.4. Сценарии выполнения услуг

Для каждой оказываемой услуги разрабатывают сценарий ее выполнения. Для услуг, схожих по технологии выполнения, допускается представление сценариев их выполнения в виде единой диаграммы. Одинаковые деятельности для разных услуг представляются в виде одинаковых стереотипов «activity».

Сценарий выполнения услуги приводят в виде диаграммы деятельности (Activity). Возможны два вида диаграммы. Внешний вид первого вида диаграммы сценария выполнения услуги приведен на рисунке 5.

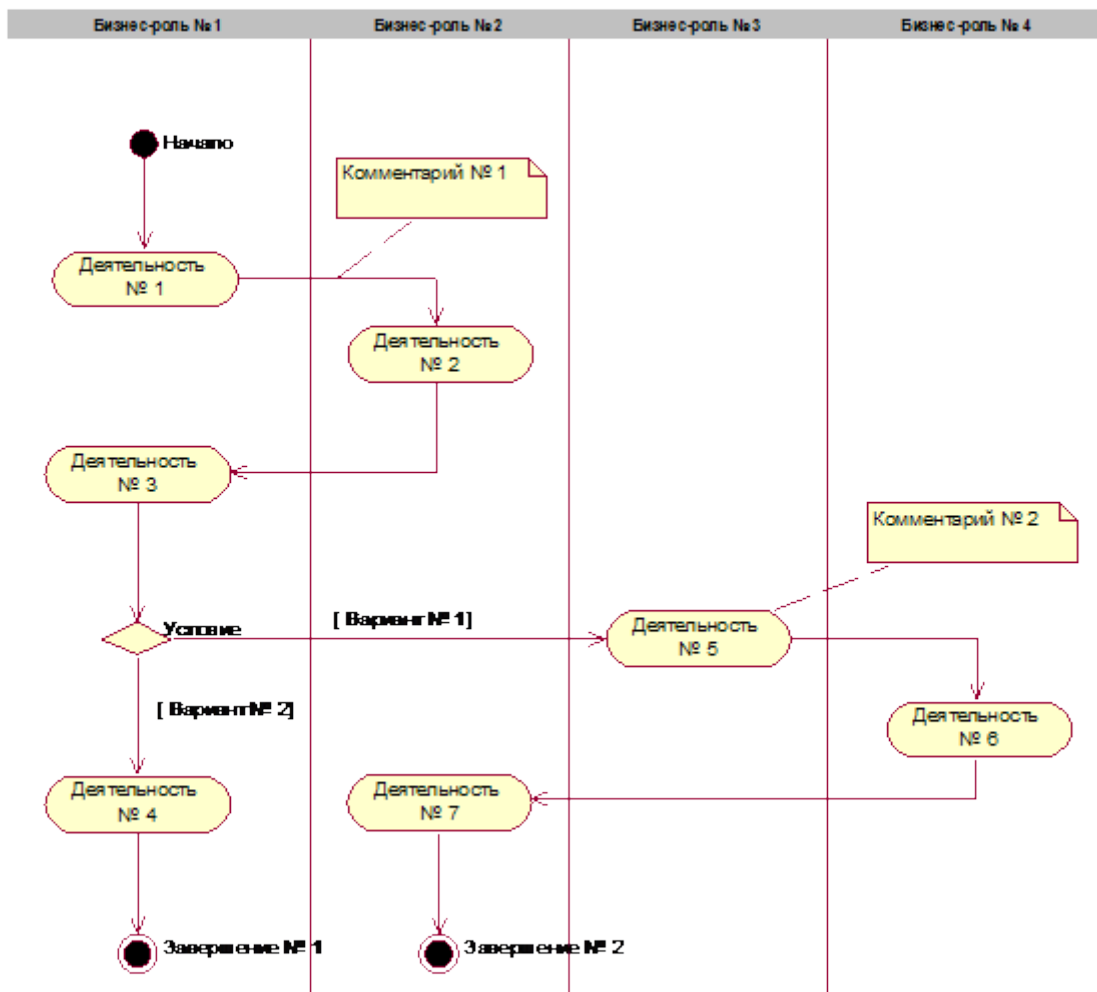








Рисунок 5. Сценарий выполнения услуги (вариант 1)

Графические стереотипы, применяемые в диаграмме сценария выполнения услуги, приведены в таблице 6.

Таблица 6. Графические стереотипы

№	Стереотип	Описание
1	 start_state (начальное состояние)	Используется для отображения начального состояния объекта
2	 end_state (финальное состояние)	Используется для отображения финального состояния объекта
3	 swimlane (дорожка)	Используется для разделения диаграммы на области, в которых приводятся действия, выполняемые бизнес-ролями
4	 activity (деятельность)	Используется для отображения действий, выполняемых бизнес-ролями

5	 Decision (решение)	Используется для отображения разветвления движения информации в зависимости от выполнения определенных условий
6	 note (комментарий)	Используется для отображения на диаграмме комментариев

На диаграмме отображают ключевые бизнес-роли, участвующие в выполнении услуги. Деятельности, выполняемые бизнес-ролями, отображают в областях, отведенных для каждой из них.

Если бизнес-роль выполняет несколько последовательных действий, неразрывных во времени, их необходимо объединять в одну деятельность и отображать на диаграмме с помощью одного стереотипа «activity».

На стрелках, исходящих от стереотипа «decision», используя его спецификацию, отображают условия движения по каждому из направлений. Размещенные на диаграмме стереотипы связывают стрелками, показывающими последовательность выполнения действий бизнес-ролями. К расположенным на диаграмме объектам рекомендуется приводить комментарии.

Наименование деятельности (значение стереотипа «activity») должно начинаться с глагола в настоящем времени, в третьем лице, без упоминания бизнес-роли, выполняющей эту деятельность.

ПРИМЕР

«Заполняет бланк анкеты»

Второй вариант внешнего вида диаграммы сценария выполнения услуг, приведен на рисунке 6.

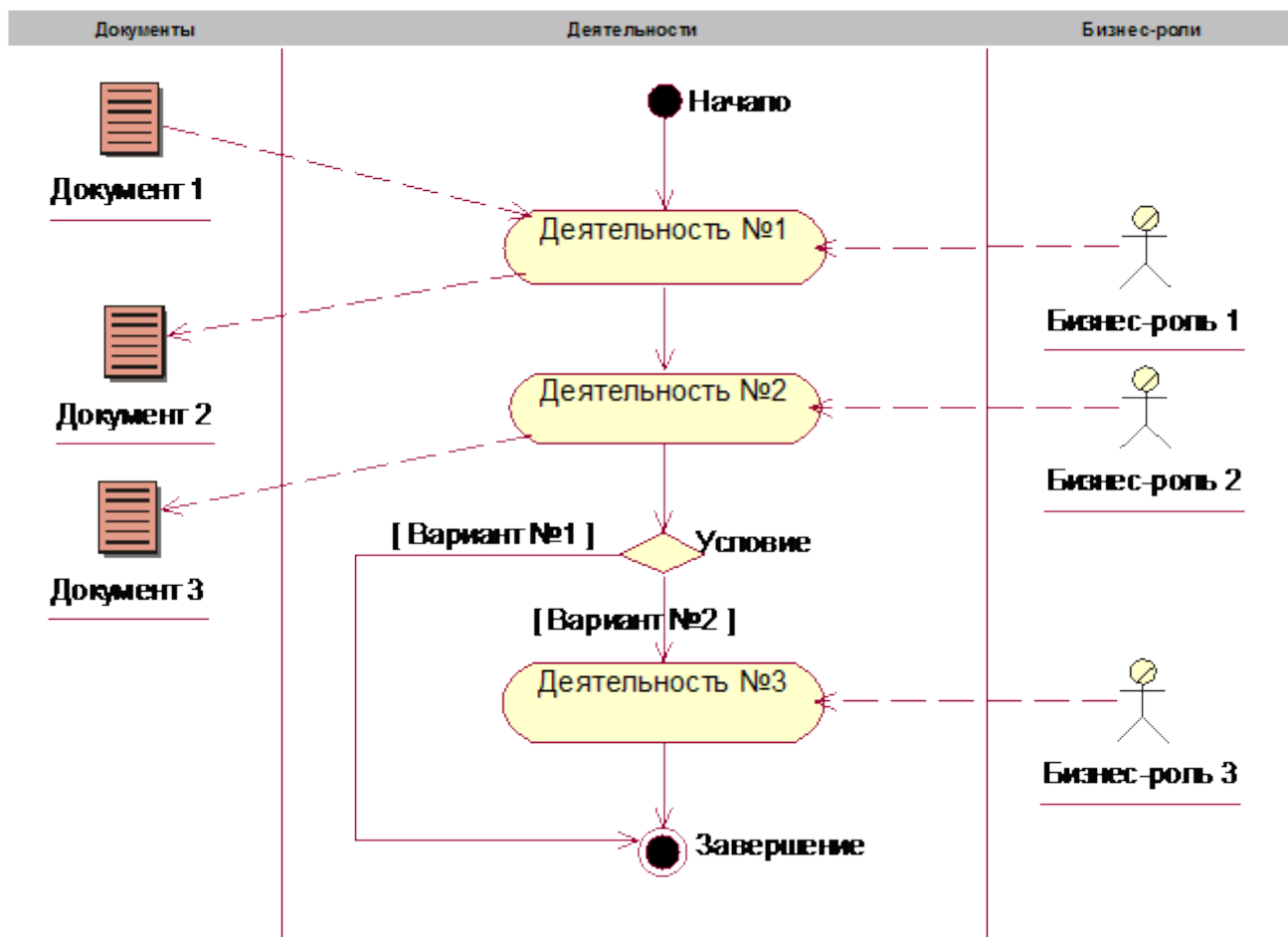


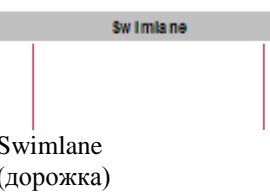





Рисунок 6. Сценарий выполнения услуги (вариант 2)

Графические стереотипы, применяемые в диаграмме сценария выполнения услуги, приведены в таблице 7.

Таблица 7. Графические стереотипы

№	Стереотип	Описание
1	 start_state (начальное состояние)	Используется для отображения начального состояния объекта
2	 end_state (финальное состояние)	Используется для отображения финального состояния объекта
3	 Swimlane (дорожка)	Используется для разделения диаграммы на три области, в которых соответственно приводят документы, деятельности и бизнес-роли
4	 Activity (деятельность)	Используется для отображения действий, выполняемых бизнес-ролями

5	 Decision (решение)	Используется для отображения разветвления движения информации в зависимости от выполнения определенных условий
6	 Document (документ)	Используется для отображения входящих и исходящих документов деятельности

Правила оформления диаграммы такие же, как и для первого варианта, но со следующими дополнениями: входящие документы деятельности связывают с деятельностью стрелкой, исходящей из документа. Исходящие документы деятельности связывают с деятельностью стрелкой, исходящей из деятельности.

2.6. Раздел "Функциональные требования к системе"

Раздел "Функциональные требования к системе" состоит из подразделов:

- функциональная модель системы;
- требования к бизнес-функциям системы.

2.6.1. Подраздел «Функциональная модель системы»

В подразделе «Функциональная модель системы» приводят одну или более диаграмм бизнес-функций, образующих функциональную модель.

Внешний вид диаграммы бизнес-функций приведен на рисунке 7.

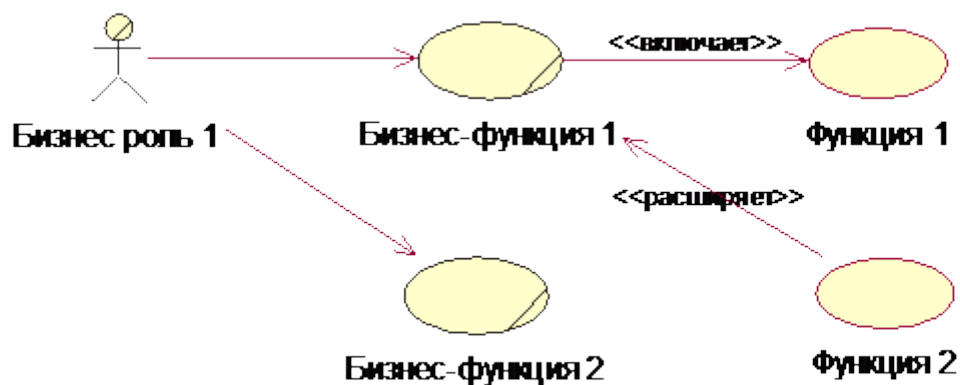
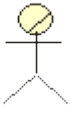



Рисунок 7. Внешний вид диаграммы бизнес-функций

На диаграмме бизнес-функций указывают все бизнес-функции, выполняемые конкретными ролями, бизнес-функции системы и отношения между ними.

В таблице 8 приведены стереотипы, используемые при построении функциональной модели.

Таблица 8. Используемые стереотипы

Стереотип	Описание
 Роль	Одно из заинтересованных лиц, обращающееся к системе за получением ее сервиса
	Бизнес-функция, выполнение которой приводит к достижению результата, ожидаемого ролью

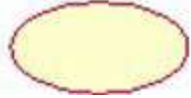
Бизнес-функция	
	Функция, выполнение которой приводит к достижению результата, ожидаемого ролью
Функция	

Диаграмма бизнес-функций строится в соответствии со стандартными правилами построения диаграммы Use Case (диаграмма функций) на языке UML.

2.6.2. Подраздел «Требования к бизнес-функциям системы»

Название подраздела «Требования к бизнес-функции системы» должно иметь вид: «Требования к бизнес-функции (приводится ее название)». Данный подраздел повторяется для описания требований к каждой бизнес-функции системы.

Подраздел «Требования к бизнес-функции системы» состоит из следующих пунктов:

- a) общие характеристики бизнес-функции;
- b) сценарий реализации бизнес-функции;
- c) входные документы;
- d) внутренние документы;
- e) выходные документы;
- f) бизнес-правила;
- g) специальные требования;
- h) варианты технологий и данных;
- i) дополнительная информация.

В зависимости от специфики бизнес-функции разрешается исключать, добавлять или объединять пункты.

2.6.2.1. В пункте «Общие характеристики бизнес-функции» указывают следующие характеристики бизнес-функции:

- результат успешного выполнения;
- минимальный результат;
- предусловия выполнения бизнес-функции;
- условия инициализации бизнес-функции.

Общие характеристики бизнес-функции указывают в виде таблицы по форме в соответствии с таблицей 9, в которой необходимо заполнить графу «Значение». В зависимости от специфики бизнес-функции разрешается исключать некоторые характеристики.

Таблица 9. Общие характеристики бизнес-функции

Характеристика	Значение
Результат успешного выполнения	
Минимальный результат	
Предусловия выполнения бизнес-функции	
Условия инициализации бизнес-функции	

В строке «Результат успешного выполнения» указывают, что получают роли в результате успешного завершения бизнес-функции по окончании главного успешного сценария либо его альтернативной ветви.

В строке «Минимальный результат» указывают результат работы бизнес-функции при отсутствии достижения конечного результата.

В строке «Предусловия выполнения бизнес-функции» указывают данные и условия, которые система должна проверить на истинность, прежде чем разрешит запуск бизнес-функции.

В строке «Условия инициализации бизнес-функции» указывают событие, наступление которого приводит к запуску бизнес-функции.

2.6.2.2. В пункте «Сценарий реализации бизнес-функции» приводят успешный сценарий работы бизнес-функции и его расширения.

Успешный сценарий описывается при помощи Activity diagram - диаграммы деятельности. Внешний вид диаграммы успешного сценария приведен на рисунке 8.

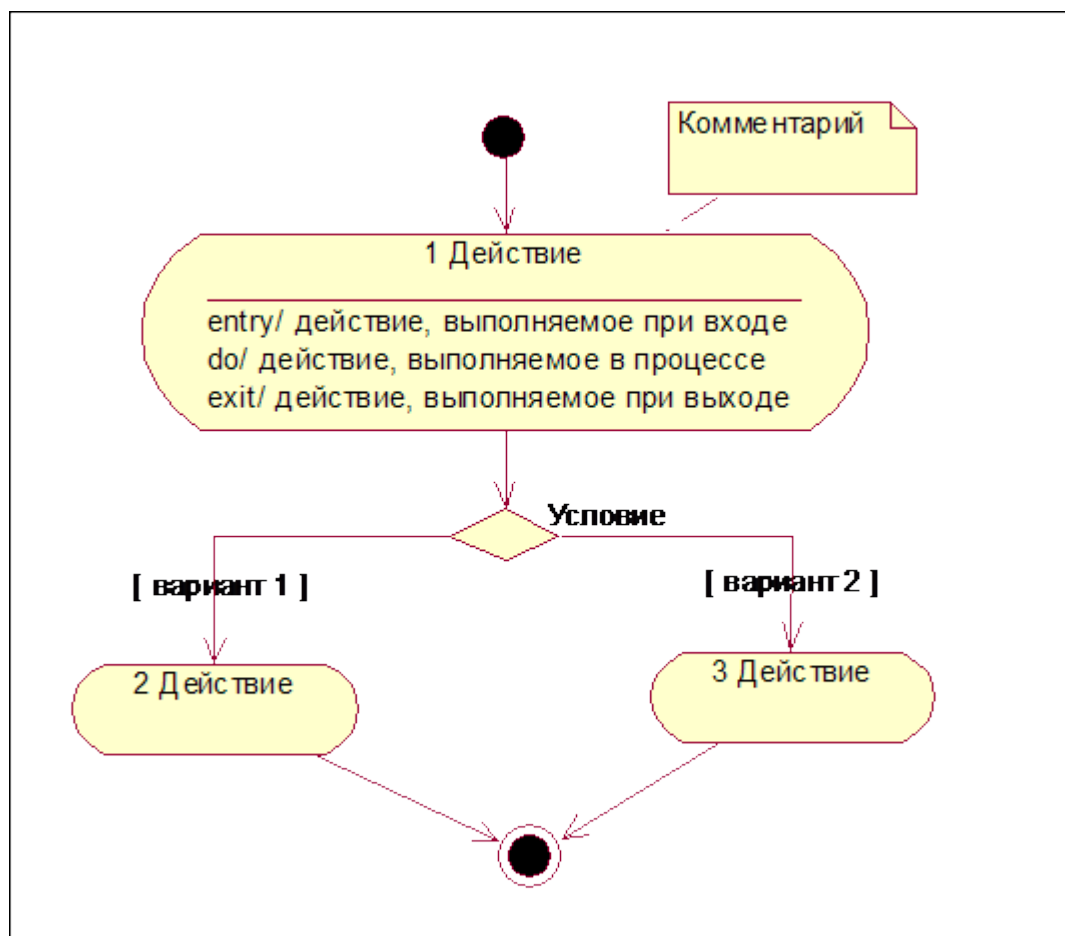


Рисунок 8. Внешний вид диаграммы успешного сценария

Диаграмма деятельности строится в соответствии с правилами построения диаграммы Activity diagram (диаграмма деятельности) на языке UML.

Нумерация действий начинается с «1». Номер действия указывается перед его наименованием и отделяется от наименования пробелом.

Разрешается детализировать действие посредством добавления элементарных действий, выполняемых при начале действия (On Entry), в процессе выполнения действия (Do) и при завершении действия (On Exit).

Допускается включать дополнительные условия или бизнес-правила в виде комментариев.

Расширения успешного сценария оформляются в виде таблицы по форме в соответствии с таблицей 10.

Таблица 10. Расширения успешного сценария

Успешный сценарий	Условия	Действия
...

В графе «Успешный сценарий» указывают наименование действий успешного сценария из диаграммы деятельностей.

В графе «Условия» указывают условия нарушения нормального выполнения успешного сценария. Нумерация условий образуется из номера действия успешного сценария, к которому относится условие, и порядкового номера условия для данного действия, разделенных точкой.

В графе «Действия» указывают действия, которые необходимо выполнить при наступлении условия. Нумерация этих действий образуется из номера условия, при котором необходимо выполнить данное действие, и порядкового номера действия, разделенных точкой.

В случае, если условие может наступить в любой момент выполнения сценария, его необходимо вынести за пределы успешного сценария. Нумерация такого условия всегда начинается с 0. Данное условие должно быть указано в таблице перед описанием расширений успешного сценария.

Разрешается делать ссылки на бизнес-правила, приведенные в пункте ТЗ «Бизнес-правила», в любой графе таблицы.

Пример оформления расширения успешного сценария приведен в таблице 11.

Таблица 11. Пример оформления расширения успешного сценария

Успешный сценарий	Условия	Действие
	0.1 Отсутствие электричества	0.1.1 Прекращение процесса 0.1.2 Выполнение действий, предписанных инструкцией
	0.2 Компьютер оператора завис	0.2.1 Прекращение процесса 0.2.2 Выполнение действий, необходимых для перезагрузки системы 0.2.3 Начать процесс сначала
1. Заполняет анкету.		
2. Печатает анкету	2.1 Устройство печати выдало сообщение об ошибке	2.1.1 Выполнение действий, предписанных инструкцией по эксплуатации печатающего устройства

2.6.2.3. В пункте «Входные документы» приводится перечень входных документов по отношению к бизнес-функции.

Перечень входных документов приводят в виде таблицы по форме в соответствии с таблицей 12.

Таблица 12. Входные документы

Документ	Тип	Внешний вид	Требования
...

В графе «Документ» указывают название документа.

В графе «Тип» указывают тип документа:

- бумажный;
- электронный;
- на электронной микросхеме (чип);
- артефакт.

В графе «Внешний вид» дают ссылку на приложение, в котором приведен внешний вид документа.

В графе «Требования» указывают требования к содержанию или заполнению документа.

Разрешается приводить требования к содержанию или заполнению документа в бизнес-правиле или приложении. В этом случае в графе «Требования» указывают название бизнес-правила или приложения соответственно.

В зависимости от информационной структуры или особенностей документа разрешается удалять и добавлять колонки в таблицу.

2.6.2.4. В пункте «Внутренние документы» приводят перечень внутренних документов бизнес-функции. Внутренние документы создаются в процессе выполнения бизнес-функции и не выходят за ее пределы.

Описание внутренних документов производят аналогично описанию входных документов (см. п.2.6.2.3).

2.6.2.5. В пункте «Выходные документы» приводят перечень выходных документов бизнес-функции. Выходные документы являются информационным отражением результата какой-либо деятельности в процессе выполнения бизнес-функции.

ПРИМЕЧАНИЕ - Если результатом выполнения бизнес-функции является артефакт, то его указывают в перечне выходных документов с указанием типа «артефакт».

Описание выходных документов осуществляется аналогично описанию входных документов (см. 2.6.2.3).

2.6.2.6. В пункте «Бизнес-правила» приводят перечень бизнес-правил. Пункт «Бизнес-правила» должен быть разбит на подпункты. Каждый подпункт соответствует одному бизнес-правилу. Название подпункта должно соответствовать названию бизнес-правила.

Бизнес-правила, распространяющиеся на несколько бизнес-функций, допускается приводить в приложениях, при этом название приложения должно соответствовать названию бизнес-правила. В этом случае в подпункте указывают только ссылку на это приложение.

2.6.2.7. В пункте «Специальные требования» указывают специальные требования к реализации бизнес-функции, которые нет возможности указать в других пунктах.

В данном пункте приводят специальные требования по безопасности и защищенности, сохранности информации, надежности системы, если они являются специфическими по отношению к требованиям, приведенным в разделе «Требования к системе в целом».

2.6.2.8. В пункте «Варианты технологий и данных» приводят список различных способов выполнения действий сценария бизнес-функции. Действия выполняются те же, но способ их выполнения меняется.

Список способов выполнения действий сценария бизнес-функции приводят в виде таблицы по форме в соответствии с таблицей 13.

Таблица 13. Варианты технологий и данных

Действие	Варианты
...	...

В графу «Действие» вносится наименование действия из пункта «Сценарий реализации бизнес-функции», а в графу «Варианты» заносят варианты технологии или данных.

Пример заполнения приведен в таблице 14.

Таблица 14. Варианты технологий и данных

Действие	Варианты
2 Печать справки	Предусмотреть возможность печати на матричном и лазерном принтерах

2.6.2.9. В пункт «Дополнительная информация» включают любую дополнительную информацию общего характера, не включенную ни в один из других пунктов.

2.7. Раздел "Требования к системе в целом"

Раздел "Требования к системе в целом" состоит из подразделов:

- требования по безопасности и защищенности;
- требования по сохранности информации;
- требования к надежности системы;
- порядок испытания и приемки;
- требования к аппаратному обеспечению и каналам связи;
- требования к документации.

В подразделе «Требования по безопасности и защищенности» указывают требования к защите информации от несанкционированного доступа.

В подразделе «Требования по сохранности информации» приводят перечень требований к обеспечению сохранности информации в системе, а также перечень мероприятий и/или действий для обеспечения сохранности информации.

В подразделе «Требования к надежности системы» указывают требования к надежности программного обеспечения.

В подразделе «Порядок испытания и приемки» указывают требования к порядку испытаний и порядку приемки.

В подразделе «Требования к аппаратному обеспечению и каналам связи» указывают требования к аппаратному обеспечению и каналам связи. Требования к аппаратному обеспечению и каналам связи устанавливаются на основании требований, предъявляемых программным обеспечением к аппаратному обеспечению и каналам связи.

В подразделе «Требования к документации» указывают требования к составу разрабатываемой документации.

Приложение 5
к RT 38370656 - 002:2006

ФОРМА АКТА ПЕРЕДАЧИ В ОПЫТНУЮ ЭКСПЛУАТАЦИЮ УТВЕРЖДАЮ

(должность, подпись, Ф.И.О.)
« ____ » _____ 200__ г.

АКТ передачи в опытную эксплуатацию

(наименование программной системы)

« ____ » _____ 200__ г. _____
(место издания)

Основание: в соответствии с _____, согласно техническому заданию _____
в период с _____ по _____ производилась разработка _____

(наименование программной системы)

В соответствии с протоколом квалификационного тестирования № _____ от _____

(наименование программной системы)

признана в основном соответствующей требованиям технического задания и должна быть передана для проверки работоспособности и освоения пользователями в опытную эксплуатацию.

Опытную эксплуатацию провести в _____
(наименование организации)
с _____ по _____.

Все ошибки в программном продукте, выявленные в период опытной эксплуатации устранить

(ответственный разработчик)

Поставщик _____
(подпись)

(должность, Ф.И.О.)

Приложение 6
к РТ 38370656 - 002:2006

ФОРМА АКТА О ЗАВЕРШЕНИИ РАЗРАБОТКИ ПРОГРАММНОГО ПРОДУКТА

УТВЕРЖДАЮ

Заказчик _____
(должность, подпись, Ф.И.О.)

« ____ » _____ 200__ г.

АКТ о завершении разработки

(наименование программного продукта)

« ____ » _____ 200__ г.

(место издания)

Основание разработки: _____

Составлен комиссией в составе:

Председатель _____
(должность, имя, фамилия)

Члены комиссии:

1. _____
(должность, имя, фамилия)

2. _____
(должность, имя, фамилия)

3. _____
(должность, имя, фамилия)

Комиссия в период с _____ по _____ проводила анализ результатов
опытной эксплуатации _____,
(наименование программного продукта)
разработанного _____
(наименование организации)

Изучив техническое задание и представленный программный продукт, комиссия установила:

разработанный программный продукт _____

(наименование программного продукта)

_____ техническому заданию _____
(соответствует, не соответствует)

Программный продукт выполнен на _____
(хорошем, удовлетворительном, плохом)
научно-техническом уровне.

Комиссия предлагает программный продукт _____

(наименование программного продукта)

В _____
(передать в эксплуатацию, передать для развертывания, вернуть на доработку)
с _____ 200__ г. (срок может не указываться).
(наименование организации)

Для инсталляции программного продукта с _____
(дата)
к _____
(перечень ресурсов)
сотрудникам _____ :
(наименование организации)
1. _____
2. _____
3. _____

Поставщику программного продукта _____
(наименование организации)
передать _____
(инсталляционный пакет, исходные коды, проектную и техническую документации, эксплуатационную документацию)
В _____
(техническую библиотеку поставщика, наименование организации заказчика)
до _____ 200__ г.

Председатель _____
(подпись)

(расшифровка подписи)

Члены комиссии: _____
(подписи)

(расшифровка подписей)

Приложение 7
к RT 38370656 - 002:2006

ФОРМА АКТА ПЕРЕДАЧИ В ЭКСПЛУАТАЦИЮ

УТВЕРЖДАЮ

(должность, подпись, Ф.И.О.)
« ____ » _____ 200__ г.

АКТ передачи в эксплуатацию

(наименование программного продукта / информационной системы)

« ____ » _____ 200__ г.

(место издания)

На основании договора (контракта, распоряжения) _____,

в соответствии с календарным планом _____

_____ (наименование программного продукта / информационной системы)

было развернуто в следующих местах функционирования:

1. _____,

2. _____,

3. _____

Версии компонентов программного продукта (информационной системы) - согласно приложению к настоящему акту.

Развернутый программный продукт (информационная система) _____

_____ (наименование программного продукта / информационной системы)

соответствует требованиям технического задания _____

и признана готовой к эксплуатации.

Председатель _____
(подпись)

_____ (расшифровка подписи)

Члены комиссии: _____
(подписи)

_____ (расшифровка подписей)

Приложение 8
к RT 38370656 - 002:2006

СТРУКТУРА КОНЦЕПЦИИ СОПРОВОЖДЕНИЯ

Концепция сопровождения должна быть разработана сразу после окончания опытной эксплуатации программного продукта.

Концепцию сопровождения разрабатывает сопроводитель для каждого программного продукта, полученного для сопровождения, которую затем утверждает заказчик.

Концепция сопровождения должна содержать следующие разделы:

- область сопровождения программного продукта;
- практическое применение процесса сопровождения;
- определение ответственных за сопровождение;
- оценка стоимости сопровождения.

1. Область сопровождения

Данный раздел должен определять обязанности сопроводителя, а также, какую поддержку программного продукта он обязан обеспечить. Область сопровождения может зависеть от наличия бюджетных средств. В данном разделе должны быть описаны:

- типы выполняемого сопровождения;
- порядок сопровождения программной документации;
- действия сопроводителя при разных типах сопровождения;

- необходимый уровень обученности персонала сопровождения;
- организация службы поддержки клиентов («Hot line»);
- порядок поставки программного продукта.

2. Практическое применение процесса сопровождения

Концепция сопровождения должна отражать задачи сопровождения программного продукта после его поставки. В данном разделе необходимо определить все организации (подразделения), участвующие в процессе сопровождения, их роли и основные задачи. Также необходимо описать выбранный процесс сопровождения для этого программного продукта.

3. Определение ответственных за сопровождение

В данном разделе необходимо определить организации (подразделения) и конкретных физических лиц, отвечающих за сопровождение программного продукта. Требуется определить роли физических лиц и уточнить их обязанности.

При выполнении сопровождения по соглашению с третьей стороной, это должно быть отмечено в концепции сопровождения.

4. Оценка стоимости сопровождения

В данном разделе должна быть проведена оценка стоимости сопровождения. При наличии соглашения на сопровождение программного продукта для внешней организации необходимо учитывать:

- проезд до места расположения пользователя;
- обучение как сопровождающих, так и пользователей;
- необходимая техническая поддержка приобретенных (стандартных) программных и технических средств;
- зарплаты и премии персонала.

При разработке концепции стоимость оценивают на основе ограниченных данных. Эта стоимость может быть уточнена в процессе сопровождения.

В качестве исходных данных могут быть использованы данные по аналогичным проектам.

Приложение 9
к RT 38370656 - 002:2006

СОДЕРЖАНИЕ ПЛАНА СОПРОВОЖДЕНИЯ

В зависимости от объема работ по сопровождению, должно быть принято решение о внесении тех или иных разделов в конкретный план сопровождения.

Разделы, рекомендованные для включения в план сопровождения:

1. ВВЕДЕНИЕ

- 1.1. Описание сопровождаемого программного продукта
- 1.2. Определение исходных состояний программного продукта
- 1.3. Описание уровня требуемой поддержки
- 1.4. Определение организации (подразделений), проводящей сопровождение
- 1.5. Описание условий (протоколов), согласованных между заказчиком и поставщиком

2. ОРГАНИЗАЦИОННЫЕ РАБОТЫ И РАБОТЫ ПО СОПРОВОЖДЕНИЮ

- 2.1. Роли и обязанности сопровождающего до поставки программного продукта

- 2.1.1. Реализация процесса сопровождения
- 2.1.2. Определение инфраструктуры процесса сопровождения
- 2.1.3. Установление процесса обучения
- 2.1.4. Установление процесса сопровождения
- 2.2. Роли и обязанности сопровождаителя после поставки программного продукта
 - 2.2.1. Реализация процесса сопровождения
 - 2.2.2. Анализ проблем и модификаций
 - 2.2.3. Реализация (внесение) модификаций
 - 2.2.4. Рассмотрение и принятие модификаций
 - 2.2.5. Перенос программного продукта в новые условия
 - 2.2.6. Изъятие программного продукта из эксплуатации
 - 2.2.7. Решение проблем (включая службу поддержки клиентов)
 - 2.2.8. При необходимости - обучение персонала (сопровождаителя и пользователя)
 - 2.2.9. Усовершенствование процесса сопровождения
- 2.3. Роль пользователя
 - 2.3.1. Приемочные испытания
 - 2.3.2. Взаимосвязи с другими организациями

3. РЕСУРСЫ

- 3.1. Персонал
 - 3.1.1. Состав персонала для конкретного проекта
- 3.2. Программные средства
 - 3.2.1. Определение программных средств, необходимых для поддержки эксплуатации системы (с учетом системных требований)
- 3.3. Технические средства
 - 3.3.1. Определение технических средств, необходимых для поддержки эксплуатации системы (с учетом системных требований)
- 3.4. Оборудование (аппаратура)
 - 3.4.1. Определение требований к оборудованию (аппаратуре) системы (помимо технических средств вычислительной техники)
- 3.5. Документы
 - 3.5.1. План обеспечения качества
 - 3.5.2. План управления конфигурацией (может быть совместным с процессом разработки)
 - 3.5.3. Документы разработки (передаются из процесса разработки)
 - 3.5.4. Инструкции по сопровождению
 - 3.5.5. Порядок проведения верификации и аттестации программного продукта
 - 3.5.6. Процедуры тестирования и акты о тестировании
 - 3.5.7. План обучения
 - 3.5.8. Инструкции по эксплуатации программных продуктов (передаются из процесса разработки)
- 3.6. Информационные ресурсы

4. ПРОЦЕСС ВЫПОЛНЕНИЯ КОНКРЕТНОЙ ДЕЯТЕЛЬНОСТИ

- 4.1. Процессы, выполняемые сопровождаителем

5. ОБУЧЕНИЕ

- 5.1. Определение уровня обучения, необходимого для сопровождаителя и пользователя

6. ПРОТОКОЛЫ И ОТЧЕТЫ ПО СОПРОВОЖДЕНИЮ

- 6.1. Перечень запросов пользователя на оказание услуг по сопровождению, ПМ и СП
- 6.2. Состояние запросов по категориям

6.3. Приоритеты запросов

6.4. Контрольные данные, собранные при работах по сопровождению

Приложение 10
к RT 38370656 - 002:2006

ФОРМА СПИСКА ФУНКЦИЙ ДЛЯ ИТЕРАЦИОННОГО МЕТОДА РАЗРАБОТКИ

УТВЕРЖДАЮ

(должность, подпись, Ф.И.О.)
« ____ » _____ 200__ г.

СПИСОК ФУНКЦИЙ

(Наименование программной системы)

№ п/п	Наименование функции	Краткое описание функции	Приоритет ¹⁾	Необходимое время на реализацию	Отметка о выполнении	Примечание

¹⁾ Приоритет определяется заказчиком (кроме приоритета «1») и выставляется по 5-балльной системе:
«1» - системные функции (обязательные для работоспособности программной системы, определяются разработчиком);
«2» - первоочередные функции;
«3» - основные функции;
«4» - функции общего значения;
«5» - дополнительные функции.