



ДОКУМЕНТИРОВАНИЕ И СЕРТИФИКАЦИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное

образовательное учреждение высшего образования

«КАЗАНСКИЙ ГОСУДАРСТВЕННЫЙ ЭНЕРГЕТИЧЕСКИЙ УНИВЕРСИТЕТ»

ДОКУМЕНТИРОВАНИЕ И СЕРТИФИКАЦИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Практикум

Казань

2023

УДК 004.4
ББК 32.972
Д63

Документирование и сертификация программного обеспечения :
Д63 практикум / составители: А. И. Хабибрахманова, Н. Г. Бикеева. – Казань :
КГЭУ, 2023. – 92 с.

Содержит практические работы по дисциплине «Документирование и сертификация программного обеспечения». Изложены основные сведения о работе с Единой системой программной документации, рассмотрены государственные и международные стандарты в области информатизации с целью получения базовых навыков при работе с ними, включая оценку качества программного обеспечения.

Предназначен для обучающихся по образовательной программе направления подготовки 09.03.01 Информатика и вычислительная техника, направленность (профиль) «Технологии разработки программного обеспечения».

УДК 004.4
ББК 32.972

ВВЕДЕНИЕ

Практикум разработан в рамках дисциплины «Документирование и сертификация программного обеспечения» и включает в себя материал, изучение которого способствует овладению обучающимися:

- знаниями об основных видах программной документации, об официальной нормативной базе в области документирования программных средств (ПС) и в смежных областях;

- умениями применять стандарты оформления технической документации программного обеспечения на различных стадиях жизненного цикла, составлять плановую и отчетную документации по разработке программного обеспечения, оценивая и согласовывая с заинтересованными лицами сроки выполнения поставленных задач;

- методиками составления технической и сопроводительной документации и сертификации программного обеспечения на различных этапах жизненного цикла, документирования программного обеспечения, описания продукта и пользовательской документации, формирования содержания программного продукта с описанием сроков выполнения поставленных задач и разработкой сопроводительной документации.

Для проверки уровня полученных знаний в конце каждой лабораторной работы приведены контрольные вопросы.

Таким образом, использование практикума при проведении занятий способствует достижению следующих целей:

- развитию представлений, навыков, умений, а также получению знаний об основополагающих стандартах Единой системы программной документации (ЕСПД);

- освоению системы понятий и терминов стандартизации, сертификации и лицензирования программного обеспечения (ПО);

- развитию понимания работы с особенностями сертификации средств разработки программного обеспечения и оценки его качества и надежности;

- развитию представлений, навыков и умений разрабатывать, согласовывать и выпускать все виды программной документации в соответствии с требованиями ЕСПД.

В конце практикума приведен список рекомендуемой литературы, которым обучающиеся могут воспользоваться как при подготовке к практическим занятиям [1–24], так и при самостоятельном изучении дисциплины [25–31].

Продолжительность каждого практического занятия – 4 ч.

Практическое занятие № 1

ДОКУМЕНТИРОВАНИЕ ПРОГРАММНОГО КОДА

Цель занятия

Овладеть навыками документирования программного кода согласно ГОСТ 19.402 ЕСПД «Описание программы» [1].

Основные теоретические сведения

В документах Международной организации по стандартизации (ИСО) термин «стандартизация» определяется следующим образом:

Стандартизация – деятельность, заключающаяся в нахождении решений для повторяющихся в сферах науки, техники и экономики задач, направленная на достижения оптимальной степени упорядочения в определенной области. В общем случае эта деятельность проявляется в процессах разработки, опубликования и применения стандартов.

Это определение отражает все многообразие стандартизации, характеризует ее как активную деятельность, направленную на упорядочение не только в технике, но и в других областях, предусматривает обязательное участие в ней всех заинтересованных сторон, подчеркивает, что стандартизация – это не механический отбор устоявшихся характеристик, а выбор или разработка наиболее оптимальных решений, рассчитанных не только на сегодняшний уровень науки и техники, но и учитывающих тенденции и направления технического прогресса.

Важный результат стандартизации – улучшение соответствия продукции или услуг их функциональному назначению. Стандартизация увязывает технические нормы и требования к взаимобмениваемой продукции, гарантирует ее технический уровень, надежность, долговечность и качество, создает необходимые предпосылки для углубления и расширения специализации и кооперирования производства, активно воздействует на экономию всех видов природных, материальных и энергетических ресурсов, а также приводит к постепенному выравниванию уровней технических норм и требований в национальных стандартах и доведению их до высших мировых научно-технических образцов.

В дальнейшем, говоря о стандартизации и сертификации, будет принято во внимание и понятие «совместимость», которое определяется следующим образом: **совместимость** – пригодность изделий и систем

информатизации к совместному использованию для выполнения при определенных условиях потребностей, не вызывая при этом нежелательных последствий.

Правовые основы стандартизации, обязательные для всех государственных органов управления, объектов хозяйственной деятельности и общественных объединений Российской Федерации (РФ), определены Федеральным законом «О стандартизации в Российской Федерации» от 29 июня 2015 г. № 162-ФЗ. Общее руководство работами по стандартизации в РФ возложено на Федеральное агентство по техническому регулированию и метрологии (Росстандарт).

Возрастание роли информатизации, расширение областей применения ее средств и повышение уровня ответственности решаемых с их помощью задач обуславливают в настоящее время резкое усиление требований к качеству систем и средств информатизации.

Качество средств и систем информатизации сегодня определяется:

- содержанием элементной базы средств информатизации и их безопасностью;
- совместимостью с другими средствами;
- уровнем помех;
- степенью экологичности;
- функциональными характеристиками;
- устойчивостью к внешним воздействиям;
- надежностью;
- конструкцией;
- параметрами электропитания;
- соответствием принципам открытых систем.

Основная задача работ по стандартизации в сфере информатизации – создание нормативной базы, отражающей современный научно-технический уровень и тенденции развития средств и систем информатизации.

Непосредственное выполнение и координация этих работ возложены на Министерство цифрового развития, связи и массовых коммуникаций РФ (Минцифры России).

Применительно к информатизации стандартизация заключается в определении требований к средствам, системам, процессам и др., излагаемым соответствующим образом в утвержденных документах (стандартах), обязательных для применения в установленной для них области действия.

Основу отечественной нормативной базы в области документирования программных средств (ПС) составляют стандарты, входящие в комплекс Единой системы программной документации (ЕСПД), при этом большая

часть из них была разработана в 70-е и 80-е гг., и охватывающие ту ее область, которая создается в процессе разработки ПС и связана в большей степени с документированием их функциональных характеристик.

Согласно ЕСПД программный документ – это документ, содержащий сведения, необходимые для разработки, изготовления, эксплуатации и сопровождения программного изделия. Номенклатуру программных документов определяет ГОСТ 19.101 «ЕСПД. Виды программ и программных документов» [2]. В качестве основных видов программ стандартом определяются:

компоненты – программы, рассматриваемые как единое целое, выполняющие законченную функцию и применяемые самостоятельно или в составе комплекса;

комплексы – программы, состоящие из двух или более компонентов, выполняющие взаимосвязанные функции и применяемые самостоятельно или в составе другого комплекса.

Виды программных документов и их краткое содержание представлены в стандарте описаниями, приведенными в табл. 1.1.

Таблица 1.1

Виды программных документов

Вид документа	Содержание документа
Спецификация	Состав программы и документация на нее
Ведомость держателей подлинников	Перечень предприятий, на которых хранятся подлинники программных документов
Текст программы	Запись программы с необходимыми комментариями
Описание программы	Сведения о логической структуре и функционировании программы
Программа и методика испытаний	Требования, подлежащие проверке при испытании программы, а также порядок и методы их контроля
Техническое задание	Назначение и область применения программы; технические, технико-экономические и специальные требования, предъявляемые к программе; необходимые стадии и сроки разработки; виды испытаний
Пояснительная записка	Схема алгоритма, общее описание алгоритма и (или) функционирования программы, а также обоснование принятых технических и технико-экономических решений
Эксплуатационные документы	Сведения для обеспечения функционирования и эксплуатации программы

ГОСТ 19.701 (ИСО 5807-85) «ЕСПД. Схемы алгоритмов, программ, данных и систем. Обозначения условные и правила выполнения» [3] распространяется на условные обозначения (символы) в схемах алгоритмов, программ, данных и систем и устанавливает правила выполнения схем, используемых для отображения различных видов задач обработки данных и средств их решения.

В РФ действует ряд стандартов в части документирования ПС, разработанных на основе прямого применения международных стандартов ИСО.

Так, ГОСТ Р ИСО/МЭК 9294 «Информационная технология. Руководство по управлению документированием программного обеспечения» [4] устанавливает рекомендации по эффективному управлению документированием ПС для руководителей, отвечающих за их создание. Целью стандарта является оказание помощи:

- в определении стратегии документирования ПС и необходимых ресурсов;
- при выборе стандартов по документированию, процедур документирования;
- при составлении планов документирования.

В контексте ГОСТ Р ИСО 9127 «Системы обработки информации. Документация пользователя и информация на упаковке для потребительских программных пакетов» [5]:

- потребительский программный пакет – программная продукция (ПП), спроектированная и продаваемая для выполнения определенных функций; программа и соответствующая ей документация, упакованные для продажи как единое целое;
- документация пользователя – документация, которая обеспечивает конечного пользователя информацией по установке и эксплуатации ПП;
- информация на упаковке – информация, воспроизводимая на внешней упаковке ПП, целью которой является предоставление потенциальным покупателям первичных сведений о ПП.

Задание для самостоятельного выполнения

1. Ознакомьтесь с основными стандартами в области информатизации.
2. Выберите тему проекта, для которого будет разрабатываться документация. Перечень тем приведен в прил. А.
3. Согласно ГОСТ 19.402 [1] выполните документирование из созданного Вами проекта либо законченного фрагмента программы, либо функции, либо процедуры.

Методические указания по выполнению практической работы

Согласно ГОСТ 19.402 [1] описание программы включает в себя следующие разделы:

1. Общие сведения.
2. Функциональное назначение.
3. Описание логической структуры.
4. Используемые технические средства.
5. Вызов и загрузка.
6. Входные данные.
7. Выходные данные.

В разделе «**Общие сведения**» указывают:

- обозначение и наименование программы;
- ПО, необходимое для функционирования программы;
- языки программирования, на которых написана программа.

В разделе «**Функциональное назначение**» указывают классы решаемых задач и/или назначение программы, сведения о функциональных ограничениях на применение.

Раздел «**Описание логической структуры**» должен содержать:

- алгоритм программы;
- используемые методы;
- структуру программы с описанием функций составных частей и связей между ними;
- связи программы с другими программами.

В разделе «**Используемые технические средства**» указывают типы ЭВМ и устройств, которые применяются при работе программы.

При описании раздела «**Вызов и загрузка**» указывают способ вызова программы с соответствующего носителя данных и входные точки в программу.

Раздел «**Входные данные**» отражает:

- характер, организацию и предварительную подготовку входных данных;
- формат, описание и способ кодирования входных данных.

Раздел «**Выходные данные**» отражает:

- характер и организацию выходных данных;
- формат, описание и способ кодирования выходных данных.

Требования к оформлению отчета

Отчет оформляется обучающимся индивидуально и должен содержать:

- номер и тему практической работы;
- описание информационной системы или программных средств либо программного продукта;
- ответы на контрольные вопросы.

Контрольные вопросы

1. Какие документы относятся к нормативным документам по стандартизации, действующим на территории Российской Федерации?

2. Какие основные направления работ по стандартизации в сфере информатизации существуют?

3. Какие непосредственно работы по стандартизации проводит Минцифры России?

4. Дайте определение правительственному профилю взаимосвязи открытых систем (Government Open Systems Interconnection Profile – GOSIP).

5. Опишите основные положения государственного профиля взаимосвязи открытых систем России (Госпрофиль ВОС России).

6. Перечислите некоторые организации, занимающиеся международной стандартизацией.

Практическое занятие № 2

ЖИЗНЕННЫЙ ЦИКЛ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ: МОДЕЛИ И СТАДИИ

Цель занятия

1. Ознакомиться со стандартами в области обеспечения жизненного цикла программных средств.
2. Определить модели и стадии жизненного цикла программного обеспечения.

Основные теоретические сведения

В основе деятельности по созданию и использованию программных средств лежит понятие жизненного цикла, являющегося моделью создания и использования программного обеспечения, отражающей его различные состояния, начиная с момента возникновения необходимости в ПС и заканчивая моментом его полного выхода из употребления у пользователей.

Основными целями применения стандартов и нормативных документов в жизненном цикле ПО являются:

- снижение трудоемкости, длительности, стоимости и улучшение других технико-экономических показателей проектов ПС;
- повышение качества разрабатываемых и/или применяемых компонентов и ПС в целом при их приобретении, разработке, эксплуатации и сопровождении;
- предоставление возможности расширять и масштабировать в зависимости от размерности решаемых задач набор прикладных функций;
- обеспечение переносимости прикладных программ и данных между разными аппаратно-программными платформами.

Применение стандартов позволяет ориентироваться на построение систем из крупных функциональных узлов, отвечающих их требованиям, внедрять отработанные и проверенные проектные решения. Они определяют унифицированные интерфейсы и протоколы взаимодействия компонентов таким образом, что разработчику системы, как правило, не требуется вдаваться в детали внутреннего устройства этих компонентов.

В России жизненный цикл разработки ПО установлен стандартом ГОСТ 19.102 «Стадии разработки программ и программной документации» [6] и содержит стадии и соответствующие им этапы, приведенные в табл. 2.1.

Таблица 2.1

Стадии и этапы разработки ПС

Стадии разработки	Этапы работ
Техническое задание	Обоснование необходимости разработки программы
	Научно-исследовательские работы
	Разработка и утверждение технического задания
Эскизный проект	Разработка эскизного проекта
	Утверждение эскизного проекта
Технический проект	Разработка технического проекта
	Утверждение технического проекта
Рабочий проект	Разработка программы и программной документации
	Испытания программы
Внедрение	Подготовка и передача программы

Следует также отметить существование жизненного цикла автоматизированных систем (АС). Так, действие ГОСТ 34.601 «Информационная технология. Автоматизированные системы. Стадии создания» [7] распространяется на АС, используемые в различных видах деятельности (исследование, проектирование, управление и т. п.), включая их сочетания, создаваемые в организациях, объединениях и на предприятиях. Стандарт устанавливает стадии и этапы, а также содержание работ на каждом из них (табл. 2.2), выполнение которых необходимо и достаточно для создания АС, соответствующей заданным требованиям.

Таблица 2.2

Стадии и этапы создания АС

Стадии	Этапы работ
1	2
1. Формирование требований к АС	Обследование объекта и обоснование необходимости создания АС. Формирование требований пользователя АС. Оформление отчета о выполненной работе и заявки на разработку АС (тактико-технического задания)
2. Разработка концепции АС	Изучение объекта. Проведение необходимых научно-исследовательских работ. Разработка вариантов концепции АС и выбор варианта концепции АС, удовлетворяющего требованиям пользователя. Оформление отчета о выполненной работе

1	2
3. Техническое задание	Разработка и утверждение технического задания на создание АС
4. Эскизный проект	Разработка предварительных проектных решений по системе в целом и ее частям, а также документации на АС и ее части
5. Технический проект	Разработка проектных решений по системе и ее частям, документации на АС и ее части, заданий на проектирование в смежных частях проекта объекта автоматизации. Создание и оформление документации на поставку изделий для комплектования АС и/или технических требований (технических заданий) на их разработку.
6. Рабочая документация	Разработка рабочей документации на систему и ее части. Создание или адаптация программ

В зависимости от специфики создаваемых АС и условий их создания возможно:

- выполнение отдельных этапов работ до завершения предшествующих стадий;
- параллельное выполнение этапов работ, а также включение новых.

ГОСТ Р ИСО/МЭК 12207 «Информационная технология. Процессы жизненного цикла программных средств» [8] наиболее полно на уровне международных стандартов отражает жизненный цикл, технологию разработки и обеспечения качества сложных ПС.

Жизненный цикл ПС представлен набором этапов, частных работ и операций в последовательности их выполнения и взаимосвязи, регламентирующих ведение разработки на всех стадиях от подготовки технического задания до завершения испытаний ряда версий и окончания эксплуатации ПС. В жизненный цикл включаются описания исходной информации, способов выполнения операций и работ, устанавливаются требования к результатам и правилам их контроля, а также к содержанию технологических и эксплуатационных документов. Определяется организационная структура коллективов, распределение и планирование работ, а также контроль за реализацией жизненного цикла ПС.

Стандарт может использоваться как непосредственный директивный, руководящий или рекомендательный документ, а также как организационная база при создании средств автоматизации соответствующих технологических этапов или процессов. Для реализации положений стандарта должны быть выбраны инструментальные средства, совместно образующие взаимосвязанный комплекс технологической поддержки и автоматизации жизненного цикла и не противоречащие предварительно скомпонованному

набору нормативных документов. Имеющиеся в стандарте пробелы следует заполнять спецификациями или нормативными документами, регламентирующими применение выбранных или созданных инструментальных средств автоматизации разработки и документирования ПС.

Под **моделью** жизненного цикла ПО понимается структура, определяющая последовательность выполнения и взаимосвязи процессов, действий, задач на протяжении жизненного цикла. Модель зависит от специфики, масштаба и сложности проекта и специфики условий, в которых система создается и функционирует.

Наибольшее распространение получили каскадная (рис. 2.1) и спиральная (рис. 2.2) модели жизненного цикла ПО. Согласно этим моделям проект реализуется в строго определенной последовательности: формирование требований к ПО, проектирование, реализация, тестирование, ввод в действие, эксплуатация и сопровождение, снятие с эксплуатации.

Принципиальной особенностью каскадного подхода является переход на следующую стадию, который осуществляется только после того, как будет полностью завершена работа на текущей стадии, и возвратов на пройденные стадии не предусматривается. Каждая стадия заканчивается получением некоторых результатов, которые служат в качестве исходных данных для следующей стадии.

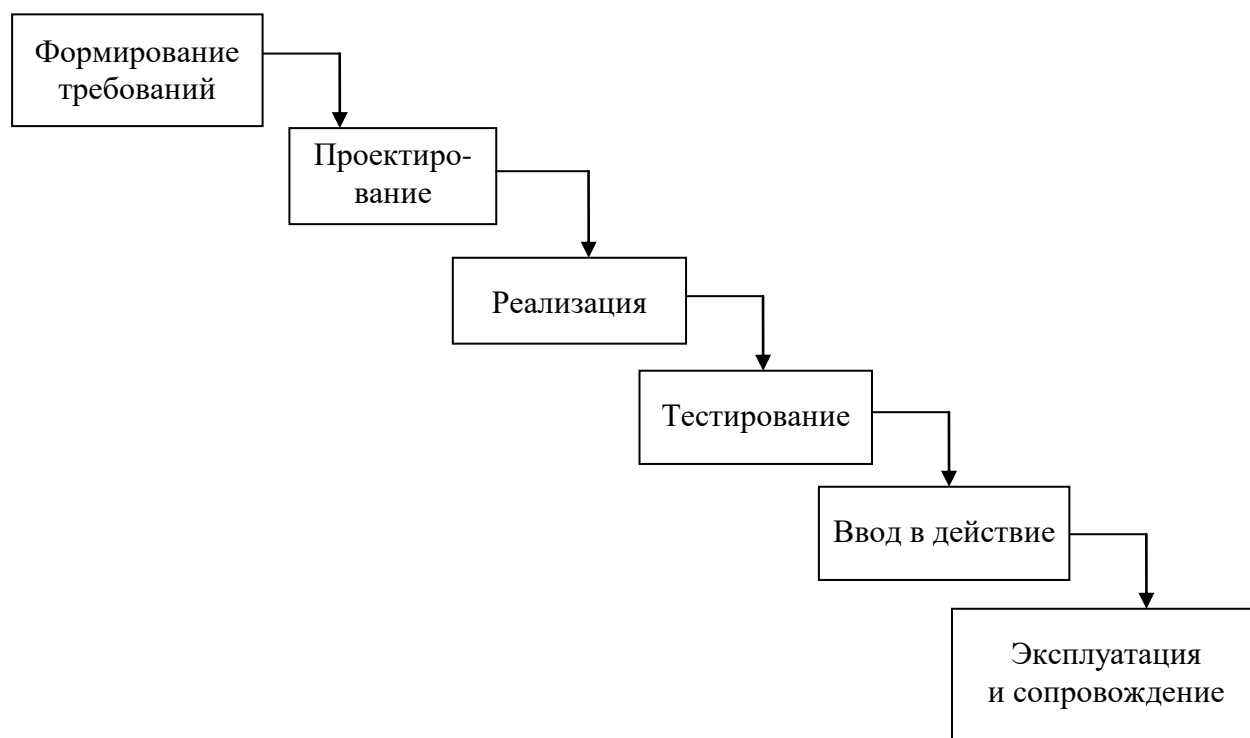


Рис. 2.1. Каскадная модель

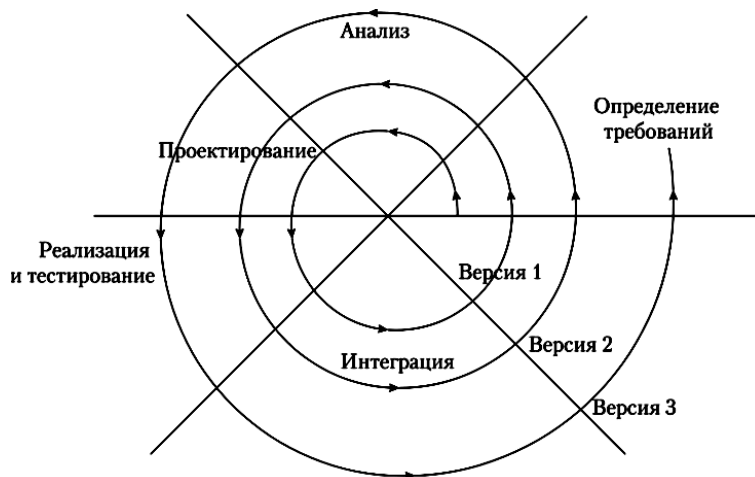


Рис. 2.2. Спиральная модель

Преимущества применения каскадного способа:

- на каждой стадии формируется законченный набор проектной документации, отвечающий требованиям полноты и согласованности;
- выполняемые в логической последовательности стадии работ позволяют планировать сроки их завершения и соответствующие затраты.

Каскадный подход хорошо зарекомендовал себя при построении систем, для которых в самом начале разработки можно достаточно точно и полно сформулировать все требования. В эту категорию попадают сложные расчетные системы, системы реального времени и другие подобные задачи.

В то же время этот подход обладает рядом недостатков, вызванных, прежде всего тем, что реальный процесс создания программного обеспечения никогда полностью не укладывается в такую жесткую схему. Процесс создания ПО носит, как правило, итерационный характер: результаты очередной стадии часто вызывают изменения в проектных решениях, выработанных на предыдущих стадиях. Таким образом, постоянно возникает потребность в возврате к предыдущим стадиям и уточнении или пересмотре ранее принятых решений.

Для преодоления перечисленных проблем была предложена спиральная модель, делающая упор на начальные этапы жизненного цикла: анализ и проектирование.

Каждый виток спирали соответствует созданию фрагмента или версии ПО, на нем уточняются цели проекта, определяется его качество и планируются работы следующего витка спирали. Углубляются и последовательно конкретизируются детали проекта, и в результате выбирается обоснованный вариант, который доводится до реализации.

Неполное завершение работ на каждом этапе позволяет переходить на следующий этап, не дожидаясь полного завершения работы на текущем. Основная проблема спирального цикла – определение момента перехода на следующий этап. Для ее решения необходимо ввести временные ограничения на каждый из этапов жизненного цикла.

Задание для самостоятельного выполнения

1. Выберите тему проекта (прил. А) и модель жизненного цикла разрабатываемой информационной системы.
2. Определите процессы для первой стадии «Формирование требований к ПО».
3. Оформите отчет.

Требования к оформлению отчета

Отчет оформляется обучающимся индивидуально и должен содержать:

- номер и тему практической работы;
- обоснование выбора модели жизненного цикла разрабатываемого ПО;
- описание процессов, характерных для первой стадии;
- ответы на контрольные вопросы.

Контрольные вопросы

1. Опишите особенности каскадной модели жизненного цикла ПС.
2. Опишите особенности спиральной модели жизненного цикла ПС.
3. Какая модель жизненного цикла ПС характерна для периода 1970–1985 гг.?
4. В чем состоит отличие спиральной модели жизненного цикла ПС от каскадной?
5. Перечислите этапы работ согласно ГОСТ 19.102 «Стадии разработки программ и программной документации».

Практическое занятие № 3

КАЧЕСТВО ПРОГРАММНЫХ СРЕДСТВ

Цель занятия

Ознакомиться с процедурой оценки качества программных продуктов.

Основные теоретические сведения

Одной из важнейших проблем обеспечения качества ПС является формализация соответствующих характеристик и методология их оценки. При определении адекватности качества функционирования, наличия технических возможностей ПС к взаимодействию, совершенствованию и развитию необходимо воспользоваться соответствующими стандартами.

Показатели качества программного обеспечения устанавливают ГОСТ 28.195 «Оценка качества программных средств. Общие положения» [9] и ГОСТ Р ИСО/МЭК 9126 «Информационная технология. Оценка программной продукции. Характеристика качества и руководства по их применению» [10]. Одновременное существование двух действующих стандартов, нормирующих одни и те же показатели, ставит вопрос об их гармонизации. Ниже рассмотрен каждый из перечисленных стандартов.

ГОСТ 28.195 «Оценка качества программных средств. Общие положения» устанавливает общие положения по оценке качества ПС, номенклатуру и применяемость показателей качества.

Оценка качества ПС представляет собой совокупность операций, включающих выбор номенклатуры показателей качества оцениваемого ПС, определение значений этих показателей и сравнение их с базовыми.

Методы определения показателей качества ПС различаются:

- по способам получения информации о ПС – измерительный, регистрационный, органолептический, расчетный;
- по источникам получения информации – экспертный, социологический.

Измерительный метод основан на получении информации о свойствах и характеристиках ПС с использованием инструментальных средств. Например, так определяется объем ПС – число строк исходного текста программ и строк-комментариев, количество операторов и операндов, исполненных операторов, ветвей в программе, точек входа (выхода), а также время выполнения ветви программы, реакции и другие показатели.

Регистрационный метод основан на получении информации во время испытаний или функционирования ПС, когда регистрируются и подсчитываются определенные события, например: время и число сбоев и отказов, время передачи управления другим модулям, время начала и окончания работы.

Органолептический метод основан на использовании информации, получаемой в результате анализа восприятия органов чувств (зрения, слуха), и применяется для определения таких показателей, как удобство, эффективность и т. п.

Расчетный метод основан на использовании теоретических и эмпирических зависимостей (на ранних этапах разработки), статистических данных, накапливаемых при испытаниях, эксплуатации и сопровождении ПС. При помощи расчетного метода определяются длительность и точность вычислений, время реакции, необходимые ресурсы.

Определение значений показателей качества ПС экспертным методом осуществляется группой экспертов-специалистов, компетентных в решении данной задачи, на базе их опыта и интуиции. Экспертный метод применяется в случаях, когда задача не может быть решена никаким другим из существующих способов или другие способы являются значительно более трудоемкими. Экспертный метод рекомендуется использовать при определении показателей наглядности, полноты и доступности программной документации, легкости освоения, структурности.

Социологические методы основаны на обработке специальных анкет-вопросников.

Показатели качества объединены в систему из четырех уровней, где каждый вышестоящий уровень содержит в качестве составляющих показатели нижестоящих уровней (рис. 3.1).

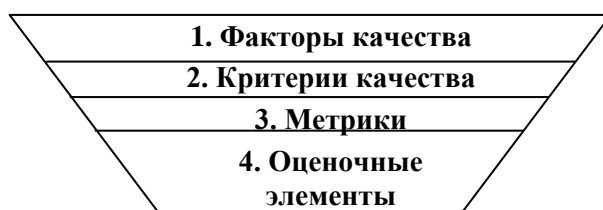


Рис. 3.1. Уровни системы показателей качества

Определенные ИСО 9126 (ГОСТ Р ИСО/МЭК 9126) «Информационная технология. Оценка программной продукции. Характеристика качества и руководства по их применению» [10] характеристики дополнены рядом требований по выбору метрик и их измерению для различных проектов ПС.

Они применимы к любому типу ПС, включая компьютерные программы и данные, содержащиеся в программируемом оборудовании. Эти характеристики обеспечивают согласованную терминологию для анализа качества ПС. Кроме того, они определяют схему для выбора и специфицирования требований к качеству ПС, а также для сопоставления возможностей различных программных продуктов, таких как функциональные возможности, надежность, практичность, эффективность и др. (рис. 3.2).



Рис. 3.2. Модель качества программного обеспечения (ISO 9126)

Все множество атрибутов качества ПС может быть классифицировано в структуру иерархического дерева характеристик и субхарактеристик. Самый высший уровень этой структуры состоит из характеристик качества, а самый нижний уровень – из их атрибутов. Эта иерархия не строгая, поскольку некоторые атрибуты могут быть связаны с более чем одной субхарактеристикой. Таким же образом, внешние свойства (такие как пригодность, корректность, устойчивость к ошибкам или временная эффективность) влияют на наблюдаемое качество. Недостаток качества в использовании (например, пользователь не может закончить задачу) может быть прослежен к внешнему качеству (например, функциональная пригодность или простота применения) и связанным с ним внутренним атрибутам, которые необходимо изменить.

Внутренние метрики могут применяться в ходе проектирования и программирования к неисполняемым компонентам ПС (таким как спецификация или исходный программный текст). При разработке ПС промежуточные продукты следует оценивать с использованием внутренних метрик, которые измеряют свойства программ, и могут быть выведены из моделируемого поведения. Основная цель внутренних метрик – обеспечивать, чтобы было достигнуто требуемое внешнее качество. Внутренние метрики дают возможность пользователям, испытателям и разработчикам оценивать качество жизненного цикла программ и заниматься вопросами технологического обеспечения качества задолго до того, как ПС становится готовым исполняемым продуктом.

Внутренние метрики позволяют измерять внутренние атрибуты или формировать признаки внешних атрибутов путем анализа статических свойств промежуточных или поставляемых программных компонентов. Измерения внутренних метрик используют категории, числа или характеристики элементов из состава ПС, которые, например, имеются в процедурах исходного программного текста, в графе потока управления, в потоке данных и в представлениях изменения состояний памяти. Документация также может оцениваться с использованием внутренних метрик.

Внешние метрики используют меры ПС, выведенные из поведения системы, частью которых они являются, путем испытаний, эксплуатации или наблюдения исполняемого ПС или системы. Перед приобретением или эксплуатацией ПС его следует оценить с применением метрик, основанных на деловых и профессиональных целях, связанных с использованием, эксплуатацией и управлением продуктом в определенной организационной и технической среде. Внешние метрики обеспечивают заказчикам, пользователям, испытателям и разработчикам возможность определять качество ПС в ходе испытаний или эксплуатации.

Когда требования к качеству ПС определены, в них должны быть перечислены характеристики и субхарактеристики, которые составляют полный набор показателей качества. Затем определяются подходящие внешние метрики и их приемлемые диапазоны значений, устанавливающие количественные и качественные критерии, которые подтверждают, что ПС удовлетворяет потребностям заказчика и пользователя. Далее определяются и специфицируются внутренние атрибуты качества, чтобы спланировать удовлетворение требуемых внешних характеристик качества в конечном продукте и обеспечивать их в промежуточных продуктах в ходе разработки. Подходящие внутренние метрики и приемлемые диапазоны специфицируются для получения числовых значений или категорий внутренних

характеристик качества, чтобы их можно было использовать для проверки того, что промежуточные продукты в процессе разработки удовлетворяют внутренним спецификациям качества. Рекомендуется использовать внутренние метрики, которые имеют наиболее сильные связи с целевыми внешними метриками, чтобы они могли помогать при прогнозировании значений внешних метрик.

Метрики качества показывают, в какой степени продукт удовлетворяет потребности конкретных пользователей в достижении поставленных целей с результативностью, продуктивностью и удовлетворением в заданном контексте использования. При этом результативность подразумевает точность и полноту достижения определенных целей пользователями при применении ПС; продуктивность соответствует соотношению израсходованных ресурсов и результативности при эксплуатации ПС, а удовлетворенность – психологическое отношение к качеству использования продукта. Эта метрика не входит в число шести базовых характеристик ПС, регламентируемых стандартом ИСО 9126, однако рекомендуется для интегральной оценки результатов функционирования программных комплексов.

Оценивание качества в использовании должно подтверждать его для определенных сценариев и задач, оно составляет полный объединенный эффект характеристик качества ПС для пользователя.

Качество в использовании – это восприятие пользователем качества системы, содержащей ПС. Измеряется оно скорее в терминах результата использования комплекса программ, чем собственных внутренних свойств непосредственно ПС. Связь качества в использовании с другими характеристиками качества ПС зависит от типа пользователя. Так, например для конечного пользователя качество в использовании обуславливают, в основном, характеристики функциональных возможностей, надежности, практичности и эффективности, а для персонала сопровождения ПС качество в использовании определяет сопровождаемость. На качество в использовании могут влиять любые характеристики качества, и это понятие шире, чем практичность, которая связана с простотой применения и привлекательностью. Качество в использовании, в той или иной степени, характеризуется сложностью применения комплекса программ, которую можно описать трудоемкостью использования с требуемой результативностью. Многие характеристики и субхарактеристики ПС обобщенно отражаются неявными технико-экономическими показателями, которые поддерживают функциональную пригодность конкретного ПС. Однако их измерение и оценка влияния на показатели качества представляют сложную проблему.

Задание для самостоятельного выполнения

1. Ознакомьтесь со стандартами в области оценки характеристик их качества.

2. Выберите тему проекта (прил. А) и произведите оценку качественных показателей реализуемого ПС, рассчитайте среднее значение оценки качества.

Методические указания по выполнению практической работы

Оценка качественных показателей ПС основана на составлении его метрики. Для этого выполните следующее:

1. Выберите показатели качества (не менее 5) и сформулируйте их сущность. Каждый показатель должен быть существенным, т. е. должны быть ясны потенциальные выгоды его использования. Показатели запишите в табл. 3.1.

Таблица 3.1

Оценка качественных показателей ПС

Показатели качества	Сущность показателя	Экспертная оценка (вес) w_i	Оценка, установленная экспериментом r_i

2. Установите веса показателей w_i ($\sum w_i = 1$).

3. Для каждого показателя установите конкретную численную оценку r_i от 0 до 1, исходя из следующего:

- оценка 0 – свойство в ПС присутствует, но качество его неприемлемо;
- от 0,5 до 1 – свойство в ПС присутствует и обладает приемлемым качеством;
- оценка 1 – свойство в ПС присутствует и обладает очень высоким качеством.

Возможно присвоение промежуточных значений в соответствии с мнением оценивающего лица относительно полезности того или иного свойства ПС по формуле:

$$K = \frac{\sum w_i \cdot r_i}{\text{общее количество показателей}}$$

Чем выше показатель K ($K_{\max} = 1$), тем лучше показатели качества ПС. Сделайте выводы о готовности вашего ПС к эксплуатации.

Требования к оформлению отчета

Отчет оформляется обучающимся индивидуально и должен содержать:

- номер и тему практической работы;
- перечень проведенных тестов;
- обоснование выбора субхарактеристик качества проекта;
- рассчитанное среднее значение оценки качества ПС;
- ответы на контрольные вопросы.

Контрольные вопросы

1. Как определяется понятие «качество» государственным и международным стандартами?

2. Какие способы получения информации о программных средствах приведены в стандарте ГОСТ 28.195?

3. Какой стандарт необходимо применить для оценки качества программных средств, если нужно оценить уровень их автоматизации?

4. Дайте определение фактору надёжности программных средств и перечислите его критерии качества согласно ГОСТ 28.195.

5. Чем обусловлена объективная необходимость повышения качества продукции и программных средств в современных условиях?

Практическое занятие № 4

ТЕХНОЛОГИИ ПРОЕКТИРОВАНИЯ

Цель занятия

Ознакомиться с промышленными технологиями создания программных продуктов.

Основные теоретические сведения

Разработка ПО – довольно трудоемкая задача, причем сложность возрастает многократно при увеличении объема работы. Можно, конечно, разрабатывать ПО на интуитивном уровне, основываясь на знаниях и умениях менеджера текущего проекта и его команды. Но такой подход не позволит разрабатывать сложные системы, поскольку риски высоки, а процесс управления ими не отработан.

Поэтому существуют определенные стандарты для разработки программного обеспечения сложных систем.

В частности, Capability Maturity Model Integration (CMMI) – набор моделей (методологий) совершенствования процессов в организациях разных размеров и видов деятельности. CMMI содержит набор рекомендаций в виде практик, реализация которых, по мнению разработчиков модели, позволяет достигать целей, необходимых для полной реализации определённых областей деятельности.

Набор CMMI включает три модели: CMMI for Development (CMMI-DEV), CMMI for Services (CMMI-SVC) и CMMI for Acquisition (CMMI-ACQ). Наиболее известной является модель CMMI for Development, ориентированная на организации, занимающиеся разработкой программного и аппаратного обеспечения, а также комплексных систем. На данный момент действуют две версии моделей: три отдельные модели версии 1.3 (вышли в ноябре 2010 года) и частично опубликованная единая версия 2.0 (вышла в конце марта 2018 года).

Согласно стандарту CMMI, организации, которым можно доверять создание ПО, должны иметь уровень три или выше. Тогда вероятность получения продукта в поставленные сроки с заданной функциональностью и бюджетом достаточно велика. Организации, находящиеся на данном уровне, имеют корпоративные стандарты управления проектами, что позволяет им добиваться успеха с большой долей вероятности. В настоящее

время все ведущие компании-разработчики программных продуктов отработали свои технологические процессы создания ПО, многие создавали стандарты собственными силами, другие – адаптируя приобретенные технологии. Очевидно, что, имея отработанную схему процесса создания программного продукта, эти организации могут ее продавать как отдельное “*know-how*”. для этого имеющаяся схема реализуется в виде технологии создания программного продукта.

Технология разработки ПО – это совокупность процессов и методов создания программного продукта. Промышленные технологии создания программных продуктов имеют несколько обязательных характеристик, среди них:

- обеспечение поддержки жизненного цикла информационной системы, т. е. каждая технология основана на какой-либо модели жизненного цикла;
- гарантия достижения целей разработки, т. е. достижение требований к системе при соблюдении сроков и бюджета;
- соответствие принципам управляемости;
- независимость от средств реализации ИС.

Обычно промышленная технология представлена в виде упорядоченной совокупности взаимосвязанных технологических процессов в рамках жизненного цикла (ЖЦ) ПО, а сам технологический процесс в виде пошаговой процедуры, определяющей последовательность технологических операций проектирования.

Технологические операции заданы при помощи графических и текстовых нотаций. Кроме того, для оценки результатов выполнения технологических операций, используются критерии и правила, которые также заданы в промышленной технологии. Саму технологическую операцию можно представить в виде черного ящика с тремя входами и одним выходом (рис. 4.1). На вход поступают некоторые исходные данные в стандартном представлении, на выходе получают результаты. Помимо этого входом являются исполнители и технические средства, а также инструкции, стандарты и критерии оценки результатов.

Технологические операции объединяются в технологическую цепочку, результатом которой является продукт, ценный для потребителя.

Выбор конкретной технологии разработки программного продукта зависит от многих факторов, оценка которых должна привести к адекватному решению по данному вопросу. Результатом такой оценки будет одна из альтернатив:

- выбор конкретной технологии разработки ПО и ее приобретение;
- вывод об отсутствии адекватных технологий.

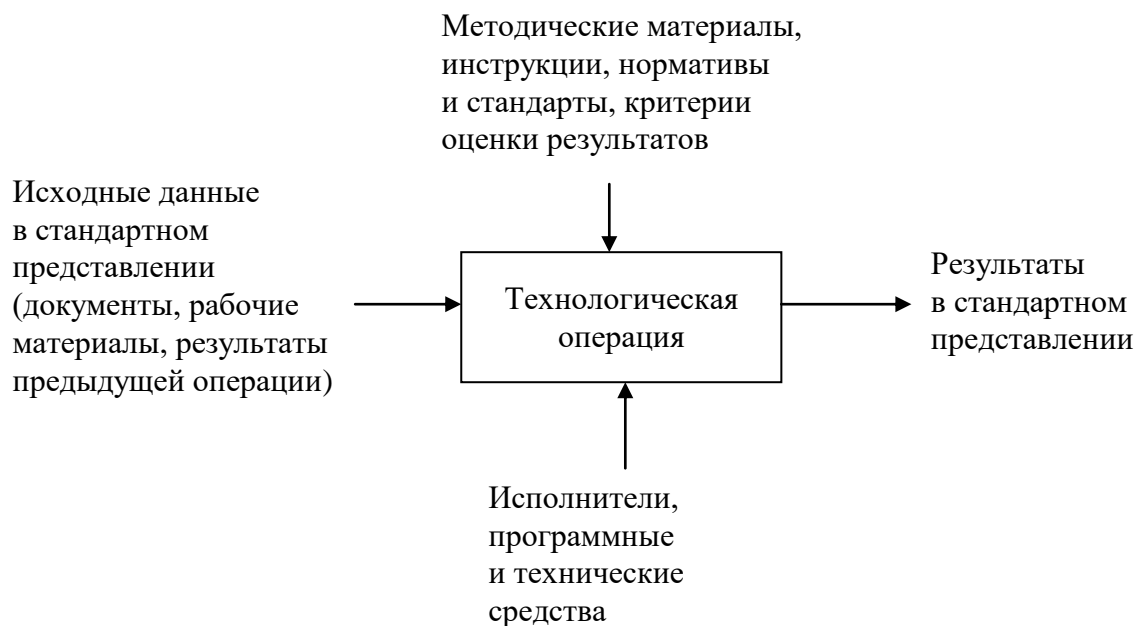


Рис. 4.1. Схема технологической операции

Во втором случае решением может быть модификация одной из существующих и удовлетворяющей определенным критериям технологий, а также разработка собственной технологии или отказ от внедрения.

Оценивать технологию разработки ПО можно по следующим технико-экономическим характеристикам:

- функциональные характеристики процессов жизненного цикла;
- функциональные характеристики применения (среда функционирования, совместимость с другими ТС ПО, соответствие технологическим стандартам);
- характеристики качества (надежность, удобство использования, эффективность, как осуществляется сопровождение, переносимость);
- общие характеристики (затраты на технологию, лицензионная политика, оценочный эффект от внедрения ТС ПО, инфраструктура, требуемая для внедрения ТС ПО, доступность и качество обучения, сертификация поставщика, поддержка поставщика).

Ниже кратко рассмотрены некоторые виды технологий проектирования.

Rational Unified Process является одним из корпоративных стандартов в области создания программного обеспечения. Данная технология была разработана компанией Rational Software.

Основой данной технологии является поэтапное моделирование продукта средствами UML, в ней реализуется итерационный и инкрементный подход

к созданию ПО. Разработка системы выполняется в виде нескольких краткосрочных мини-проектов фиксированной длительности (от 2 до 6 недель), называемых итерациями. Каждая итерация включает свои собственные этапы анализа требований, проектирования, реализации, тестирования, интеграции и завершается созданием работающей системы. Данная технология достаточно гибкая и имеет возможность масштабирования, т. е. подходит как для малых проектов и рабочих групп, так и для больших. Технология состоит из четырех основных этапов, сопровождающихся девятью видами деятельности (процессами).

Основные этапы и связанные с ними затраты по ресурсам и времени представлены графически на рис. 4.2, из которого видно, что фаза Конструирование является наиболее затратной как по ресурсам, так и по времени.

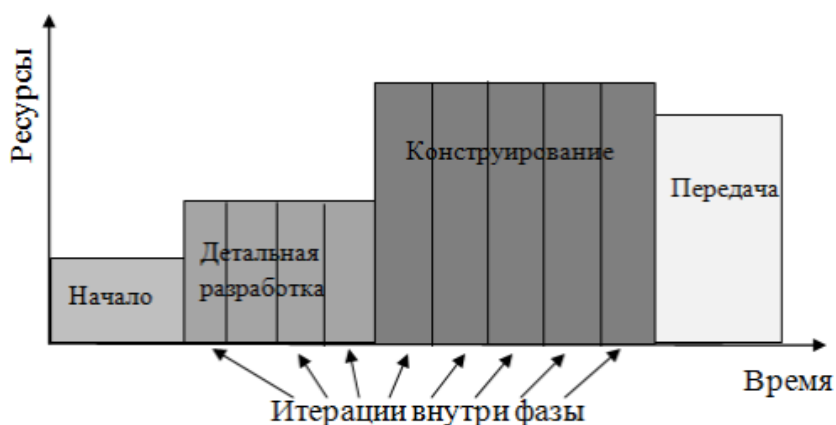


Рис. 4.2. Основные фазы RUP

При разработке ПО согласно технологии RUP, жизненный цикл разбивается на несколько итераций, на каждой из которых создается дополненная версия ПО. При этом внутри любая итерация обязательно должна содержать все четыре фазы технологии. Каждая фаза, в зависимости от трудоемкости, также может быть разбита на итерации. Все фазы выполняются последовательно друг за другом, каждая из них завершается оценкой полученных результатов в четко определенной контрольной точке, здесь принимается решение о дальнейшей разработке и осуществляется передача управления на следующую фазу.

Custom Development Method CDM (Oracle) – данная технология основана на использовании инструментального комплекса Oracle Developer Suite. Технология в основном ориентирована на разработку ПО, в котором приоритетным является создание и использование базы данных, включая конверсию при переходе на новое ПО.

Данная технология имеет две разновидности: CDM-classic и CDM-fast track, состоит из шести основных фаз и одиннадцати процессов.

CDM-classic – это технология разработки, рассчитанная на крупномасштабные проекты с временем от восьми месяцев до трех лет, CDM-fast track – для разработки маленьких проектов со временем функционирования от четырех до шестнадцати месяцев.

Основным девизом команды Microsoft при продвижении технологии Microsoft Solution Framework (MSF) является лозунг: «Лучшее из двух миров!» – это значит, что данная технология должна совместить лучшие черты из двух основных и конкурирующих между собой моделей жизненного цикла ПО – каскадной и спиральной. Из каскадной модели данная технология позаимствовала чёткость целей и переходов для каждого из этапов, а из спиральной – минимизацию рисков на каждом из них.

Каскадная модель предусматривает четкий переход от этапа к этапу: работы следующего этапа начинаются только после выполнения всех задач предыдущего. Такой стиль подходит для проектов, в которых проектные требования четко определяются заранее и с большой вероятностью не будут корректироваться потом. Данная схема организации разработки очень удобна с точки зрения управления проектом, так как позволяет четко сформулировать состав и обязанности его участников и контролировать графики выполнения проекта.

Спиральная модель обычно ориентируется на крайний случай, когда требования и параметры проекта непрерывно корректируются, а новые требования формулируются лишь по мере необходимости выполнения конкретных работ. Такая схема часто ассоциируется с понятием «экстремальной разработки»; при этом исполнитель и заказчик работают в постоянном тесном сотрудничестве, клиент привлекается на каждом этапе, формулируя свои соображения по поводу созданных компонентов.

Однако при такой организации очень велик риск, что процесс разработки выйдет из-под контроля, поэтому реально данная модель используется лишь в относительно небольших проектах. Однако проблема заключается в том, что чаще всего все требования на задание действительно практически невозможно определить заранее, к тому же даже сформулированные требования подвергаются коррекции. Но тогда требуется повысить уровень управляемости проектом, без чего создание сложного ПО просто невозможно.

Компромисс между этими противоречивыми требованиями и предоставляет модель процессов MSF, в которой сочетаются каскадная и спиральная модели разработки: проект реализуется поэтапно, с наличием соответствующих ключевых (контрольных) точек, а сама последовательность этапов может повторяться по спирали.

Данная технология претендует на всеобщность применения, т. е. предполагается, что с ее помощью можно работать над проектом любого масштаба, не внося существенных изменений в процесс применения.

Технология MSF состоит из четырех этапов, каждый из которых завершается «вехой» – ключевой точкой, в которой производится оценка достигнутого. Каждый из этапов сопровождается группой из шести процессов, в зависимости от этапа в процессе смещается точка фокуса. «Веха» каждого из этапов – это определенные задачи с необходимым уровнем качества их выполнения.

Extreme Programming (XP) и Rapid Application Development (RAD) – технологии, которые входят в группу методологий Agile (подвижный, сообразительный), имеют очень много общего, причем настолько, что в большинстве случаев они выступают как единое целое.

Основными или общими характеристиками данной группы технологий является:

- получение быстрых результатов в малом проекте с ограниченными ресурсами;

- малая рабочая группа (до 50 человек);

- короткий цикл разработки (до 6 месяцев).

При этом данный подход предполагает, что первые несколько версий, наверняка, не будут полностью работоспособны, поскольку для реализации выбирается несколько задач, возможно не связанных друг с другом. Гораздо важнее показать, что процесс движется в правильном направлении, с минимумом затрат по ресурсам и времени. Тем не менее, создание каждой версии предполагает последовательное прохождение четырех этапов, которые наглядно показаны на рис. 4.3.



Рис. 4.3. Графическое представление фаз технологии XP

Rapid Application Development (RAD) – это жизненный цикл процесса проектирования, созданный для достижения более высокой скорости разработки и повышения качества ПО, чем это возможно при традиционном подходе к проектированию.

RAD предполагает, что разработка ПО осуществляется небольшой командой разработчиков за срок порядка трех-четырех месяцев путем использования инкрементного прототипирования с применением инструментальных средств визуального моделирования и разработки. Технология RAD предусматривает активное привлечение заказчика уже на ранних стадиях – обследование организации, выработка требований к системе. Причины популярности RAD вытекают из тех преимуществ, которые обеспечивает эта технология.

Наиболее существенными из них являются:

- высокая скорость разработки;
- низкая стоимость;
- высокое качество.

Последнее из указанных свойств подразумевает полное выполнение требований заказчика как функциональных, так и нефункциональных, с учетом их возможных изменений в период разработки системы, а также получение качественной документации, обеспечивающей удобство эксплуатации и сопровождения системы. Это означает, что дополнительные затраты на сопровождение сразу после поставки будут значительно меньше. Таким образом, полное время от начала разработки до получения приемлемого продукта при использовании этого метода значительно сокращается.

Задания для самостоятельного выполнения

1. Ознакомьтесь с существующими технологиями разработки программных продуктов.
2. Определите наилучший вариант из существующих разработанных промышленных технологий для своего проекта (прил. А). Обоснуйте свой выбор.
3. Оформите отчет.

Требования к оформлению отчета

Отчет оформляется обучающимся индивидуально и должен содержать:

- номер и тему практической работы;
- краткий обзор существующих промышленных технологий разработки ПО;

- обоснование выбора технологии проектирования для своего проекта;
- письменные ответы на контрольные вопросы.

Контрольные вопросы

1. Дайте описание инкрементальной модели жизненного цикла. Какими преимуществами и недостатками обладает данная модель?

2. В чем заключается итерационный подход к созданию программного обеспечения? Для какого программного обеспечения он предназначен?

3. Рассмотрите технологию проектирования Rapid Application Development (RAD). Что общего у нее с технологией Extreme Programming (XP)? Чем эти технологии отличаются друг от друга?

4. Чем обуславливается выбор той или иной технологии создания программного продукта при разработке программного обеспечения?

Практическое занятие № 5

СОЗДАНИЕ ТЕХНИЧЕСКОЙ ДОКУМЕНТАЦИИ. ТЕХНИЧЕСКОЕ ЗАДАНИЕ

Цель занятия

Ознакомиться с процедурой разработки технического задания на создание программного продукта с применением ГОСТ 19.102 «ЕСПД. Стадии разработки программ и программной документации» [6] и ГОСТ 19.106 «ЕСПД. Требования к программным документам, выполненным печатным способом» [11].

Основные теоретические сведения

Стандарты, входящие в комплекс Единой системы программной документации (ЕСПД), в основном охватывают ту часть документации, которая создается в процессе разработки ПС, и связаны, по большей части, с документированием функциональных характеристик ПС. Следует отметить, что стандарты ЕСПД серии 19 (например, ГОСТ 19.701) носят рекомендательный характер. Дело в том, что в соответствии с Федеральным законом «О стандартизации в Российской Федерации» от 29 июня 2015 г. № 162-ФЗ эти стандарты становятся обязательными на контрактной основе, т. е. при ссылке на них в договоре на разработку (поставку) ПС.

Надо сказать, что наряду с комплексом ЕСПД официальная нормативная база РФ в области документирования ПС и в смежных областях включает ряд перспективных стандартов отечественного, межгосударственного и международного уровней (например, ГОСТ Р ИСО/МЭК 12207 «Информационная технология. Процессы жизненного цикла программных средств» [8]).

Перечень стандартов, входящих в комплекс ЕСПД

1. ГОСТ 19.001 ЕСПД. Общие положения.
2. ГОСТ 19.101 ЕСПД. Виды программ и программных документов.
3. ГОСТ 19.102 ЕСПД. Стадии разработки.
4. ГОСТ 19.103 ЕСПД. Обозначение программ и программных документов.
5. ГОСТ 19.104 ЕСПД. Основные надписи.

6. ГОСТ 19.105 ЕСПД. Общие требования к программным документам.
7. ГОСТ 19.106 ЕСПД. Требования к программным документам, выполненным печатным способом.
8. ГОСТ 19.201 ЕСПД. Техническое задание. Требования к содержанию и оформлению.
9. ГОСТ 19.202 ЕСПД. Спецификация. Требования к содержанию и оформлению.
10. ГОСТ 19.301 ЕСПД. Порядок и методика испытаний.
11. ГОСТ 19.401 ЕСПД. Текст программы. Требования к содержанию и оформлению.
12. ГОСТ 19.402 ЕСПД. Описание программы.
13. ГОСТ 19.404 ЕСПД. Пояснительная записка. Требования к содержанию и оформлению.
14. ГОСТ 19.501 ЕСПД. Формуляр. Требования к содержанию и оформлению.
15. ГОСТ 19.502 ЕСПД. Описание применения. Требования к содержанию и оформлению.
16. ГОСТ 19.503 ЕСПД. Руководство системного программиста. Требования к содержанию и оформлению.
17. ГОСТ 19.504 ЕСПД. Руководство программиста.
18. ГОСТ 19.505 ЕСПД. Руководство оператора.
19. ГОСТ 19.506 ЕСПД. Описание языка.
20. ГОСТ 19.508 ЕСПД. Руководство по техническому обслуживанию. Требования к содержанию и оформлению.
21. ГОСТ 19.604 ЕСПД. Правила внесения изменений в программные документы, выполняемые печатным способом.
22. ГОСТ 19.701 ЕСПД. Схемы алгоритмов, программ, данных и систем. Условные обозначения и правила выполнения.
23. ГОСТ 19.781 ЕСПД. Обеспечение систем обработки информации программное.

Области применения стандартов

ГОСТ 19.101 «ЕСПД. Виды программ и программных документов» [2] устанавливает виды программ и программных документов для вычислительных машин, комплексов и систем независимо от их назначения и области применения.

В зависимости от способа выполнения и характера применения программные документы подразделяются на подлинник, дубликат и копию (ГОСТ 2.102 «Виды и комплектность конструкторских документов»), предназначенные для разработки, сопровождения и эксплуатации программы.

ГОСТ 19.105 «ЕСПД. Общие требования к программным документам» [12] устанавливает общие требования к оформлению программных документов для вычислительных машин, комплексов и систем, независимо от их назначения и области применения и предусмотренных стандартами ЕСПД для любого способа выполнения документов на различных носителях данных.

Программный документ может быть представлен на различных типах носителей данных и состоит из следующих условных частей: титульной, информационной и основной.

Правила оформления документа и его частей на каждом носителе данных устанавливаются стандартами ЕСПД на правила оформления документов на соответствующих носителях данных.

Согласно **ГОСТ 19.106 «ЕСПД. Требования к программным документам, выполненным печатным способом»** [11] программные документы оформляют:

- при изготовлении документа машинописным или рукописным способом на листах формата А4 или А3 (ГОСТ 2.301 «Форматы»);

- при изготовлении документа машинным способом на листах форматов А4 и А3, предусматриваемых выходными характеристиками устройств вывода данных. Допускаются отклонения размеров листов, соответствующих форматам А4 и А3, определяемые возможностями применяемых технических средств;

- при изготовлении документа типографским способом на листах типографских форматов.

Расположение материалов программного документа осуществляется в следующей последовательности:

- 1) титульная часть:

- лист утверждения (не входит в общее количество листов документа);

- титульный лист (первый лист документа);

- 2) информационная часть:

- аннотация;

- лист содержания;

- 3) основная часть:

- текст документа (с рисунками, таблицами и т. п.);

- перечень терминов и их определений;
- перечень сокращений;
- приложения;
- предметный указатель;
- перечень ссылочных документов;

4) часть регистрации изменений:

- лист регистрации изменений.

При необходимости выполняются:

- перечни терминов и их определений и сокращений;
- приложения;
- предметный указатель;
- список ссылочных документов.

В соответствии с **ГОСТ 19.402 «ЕСПД. Описание программы»** [1] состав документа «Описание программы» в своей содержательной части может дополняться разделами и пунктами, почерпнутыми из стандартов для других описательных документов и руководств:

- ГОСТ 19.404 «ЕСПД. Пояснительная записка»;
- ГОСТ 19.502 «ЕСПД. Описание применения»;
- ГОСТ 19.503 «ЕСПД. Руководство системного программиста»;
- ГОСТ 19.504 «ЕСПД. Руководство программиста»;
- ГОСТ 19.505 «ЕСПД. Руководство оператора».

Есть также группа стандартов, определяющая требования к фиксации всего набора программ и программной документации (ПД), которые оформляются для передачи ПС. Они порождают лаконичные документы учетного характера и могут быть полезны для упорядочения всего хозяйства программ и ПД (ведь очень часто требуется просто навести элементарный порядок). Есть также стандарты, определяющие правила ведения документов ПС.

ГОСТ 19.301 «ЕСПД. Программа и методика испытаний» [13] (в адаптированном виде) может использоваться при создании документов планирования и проведения испытательных работ по оценке готовности и качества ПС.

ГОСТ 19.701 «ЕСПД. Схемы алгоритмов, программ, данных и систем. Обозначения условные графические и правила выполнения» [3] устанавливает правила выполнения схем, используемых для отображения различных видов задач обработки данных и средств их решения и полностью соответствует стандарту ИСО 5807:1985 «Обработка информации. Символы и условные обозначения блок-схем данных, программ и систем, схем программных сетей и системных ресурсов» [19].

Разработка технического задания

1. Техническое задание оформляют в соответствии с ГОСТ 19.106 [20] на листах формата А4 и А3, как правило, без заполнения полей листа. Номера страниц проставляют в верхней части листа над текстом.

2. Лист утверждения и титульный лист оформляют в соответствии с ГОСТ 19.104 «ЕСПД. Основные надписи» [21]. Информационную часть (аннотацию и содержание), лист регистрации изменений допускается в документ не включать.

3. Для внесения изменений и дополнений в техническое задание на последующих стадиях разработки программы или программного изделия выпускают дополнение к нему. Согласование и утверждение дополнения к техническому заданию проводят в том же порядке, который установлен для технического задания.

4. Техническое задание должно содержать следующие разделы:

1. Название программы и область применения.
2. Основание для разработки.
3. Назначение разработки.
4. Технические требования к программе или программному изделию.
5. Технико-экономические показатели.
6. Стадии и этапы разработки.
7. Порядок контроля и приемки.
8. Приложения.

В зависимости от особенностей программы или программного изделия допускается уточнять содержание разделов, вводить новые разделы или объединять отдельные из них.

5. Содержание разделов.

В разделе «**Название программы и область применения**» указывают наименование, дают краткую характеристику области применения программы или программного изделия и объекта, в котором их используют.

В разделе «**Основание для разработки**» должны быть указаны:

- документ (документы), на основании которых ведется разработка;
- организация, утвердившая этот документ, и дата его утверждения;
- наименование и (или) условное обозначение темы разработки.

В разделе «**Назначение разработки**» – функциональное и эксплуатационное назначение программы или программного изделия.

Раздел «**Технические требования к программе или программному изделию**» должен содержать следующие подразделы:

- требования к функциональным характеристикам;
- требования к надежности;
- условия эксплуатации;
- требования к составу и параметрам технических средств;
- требования к информационной и программной совместимости;
- требования к маркировке и упаковке;
- требования к транспортированию и хранению;
- специальные требования.

В подразделе **«Требования к функциональным характеристикам»** должны быть указаны требования к составу выполняемых функций, организации входных и выходных данных, временным характеристикам и т. п.

В подразделе **«Требования к надежности»** должны быть указаны требования к обеспечению надежного функционирования (обеспечение устойчивого функционирования, контроль входной и выходной информации, время восстановления после отказа и т. п.).

В подразделе **«Условия эксплуатации»** для выбранных типов носителей данных должны быть указаны параметры (температура окружающего воздуха, относительная влажность и пр.), обеспечивающие заданные характеристики, а также вид обслуживания, необходимое количество персонала и его квалификация.

В подразделе **«Требования к составу и параметрам технических средств»** прописывают необходимый состав технических средств с указанием их технических характеристик.

Подраздел **«Требования к информационной и программной совместимости»** должен содержать требования к информационным структурам на входе и выходе, методам решения, исходным кодам и языкам программирования. При необходимости должна обеспечиваться защита информации и программ.

В подразделе **«Требования к маркировке и упаковке»** в общем случае приводятся требования к маркировке программного изделия, варианты и способы упаковки.

В подразделе **«Требования к транспортированию и хранению»** для программного изделия должны быть прописаны условия транспортирования, хранения и складирования, а также обозначено место и сроки хранения в различных условиях.

В разделе **«Технико-экономические показатели»** должны быть указаны: ориентировочная экономическая эффективность, предполагаемая годовая потребность, экономические преимущества разработки по сравнению с лучшими отечественными и зарубежными образцами или аналогами.

В разделе «**Стадии и этапы разработки**» устанавливают необходимые стадии разработки и этапы, прописывают содержание работ (перечень программных документов, которые должны быть разработаны, согласованы и утверждены), включая сроки выполнения, и определяют исполнителей.

В разделе «**Порядок контроля и приемки**» должны быть указаны виды испытаний и общие требования к приемке работы.

6. В приложениях к техническому заданию при необходимости приводят:

- перечень научно-исследовательских и других изысканий, обосновывающих разработку;
- схемы алгоритмов, таблицы, описания, обоснования, расчеты и другие документы, которые могут быть использованы при разработке;
- прочие источники разработки.

Задание для самостоятельного выполнения

1. Ознакомьтесь с ГОСТ 19.102 [6] и ГОСТ 19.201 [14].
2. Выберите тему проекта (прил. А).
3. Разработайте техническое задание к выбранному проекту в соответствии с требованиями, приведенными в ГОСТ 19.201.

Требования к оформлению отчета

В качестве отчета о проделанной работе выступает оформленное согласно ГОСТ 19.201 [12] техническое задание к проекту, включая письменные ответы на контрольные вопросы.

Контрольные вопросы

1. Какой нормативный документ определяет участников работ по стандартизации, правила разработки стандартов и их взаимосвязь с техническими регламентами?
2. Какие положения устанавливают основополагающие стандарты?
3. Какой статус в настоящее время имеют стандарты?
4. Что такое качество документации ПС?
5. Что такое профиль стандартов?

Практическое занятие № 6

СОЗДАНИЕ ТЕХНИЧЕСКОЙ ДОКУМЕНТАЦИИ. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА К ЭСКИЗНОМУ ПРОЕКТУ

Цель занятия

1. Ознакомиться с серией стандартов, являющихся составной частью Комплекса стандартов на автоматизированные системы (ГОСТ 34).
2. Оценить достоинства и недостатки представленных стандартов, актуальность и перспективы.

Основные теоретические сведения

Нормативные документы серии 34, составляющие Комплекс стандартов на автоматизированные системы (КСАС), появились в конце 80-х – начале 90-х годов, заменив или дополнив более ранние стандарты 19-й и 24-й серий, и носят рекомендательный характер. Дело в том, что в соответствии с действующим законодательством стандарты становятся обязательными при ссылке на них в техническом регламенте (ТР) или договоре на разработку (поставку) информационных систем (ИС).

Чтобы понять и оценить логику, содержащуюся в семействе ГОСТ 34, необходимо проанализировать содержание составляющих его стандартов. Особый интерес представляют следующие нормативные документы: ГОСТ 34.201 [15], ГОСТ 34.601 [7], ГОСТ 34.602 [16] и РД50-34.698-90 [17].

ГОСТ 34.201. Виды, комплектность и обозначение документов при создании автоматизированных систем

Настоящий стандарт распространяется на автоматизированные системы (АС), используемые в различных сферах деятельности (управление, исследования, проектирование и т. п.), включая их сочетания, и устанавливает требования к видам, наименованию, комплектности и обозначению документов, разрабатываемых на стадиях создания АС.

В случае отсутствия выделения стадий (или деления на другие стадии) при создании АС перечень разрабатываемой документации и сроки ее представления определяются техническим заданием или совместным решением заказчика и разработчика.

ГОСТ 34.601. Автоматизированные системы. Стадии создания

Во-первых, этот документ является одним из наиболее применяемых до сих пор стандартов, определяет стадии и этапы создания АС. Практически все перечисленные стадии и этапы до сих пор встречаются в практике создания ИС предприятий и организаций. Стандарт не требует знаний в области информационных технологий (ИТ) и, следовательно, понятен обычным управленцам.

Во-вторых, он компактен и прост по структуре, что позволяет человеку, не знакомому с ним, быстро войти в курс дела.

В-третьих, он самодостаточен – практически никаких ссылок на смежные документы в нем нет (за исключением ГОСТ 34.201 [15]).

И наконец, он практичен – сразу понятно, как его применять и как контролировать его применение.

Стадии и этапы создания АС по ГОСТ 34.601 приведены в табл. 6.1.

Таблица 6.1

Стадии и этапы создания АС

Стадии	Этапы
1	2
1. Формирование требований к АС	1.1. Обследование объекта и обоснование необходимости создания АС
	1.2. Формирование требований пользователя к АС
	1.3. Оформление отчета о выполненной работе и заявки на разработку АС (тактико-технического задания)
2. Разработка концепции АС	2.1. Изучение объекта
	2.2. Проведение необходимых научно-исследовательских работ
	2.3. Разработка вариантов концепции АС, удовлетворяющей требованиям пользователя
	2.4. Оформление отчета о выполненной работе
3. Техническое задание	3.1. Разработка и утверждение технического задания на создание АС
4. Эскизный проект	4.1. Разработка предварительных проектных решений по системе и ее частям
	4.2. Разработка документации на АС и ее части
5. Технический проект	5.1. Разработка проектных решений по системе и ее частям
	5.2. Разработка документации на АС и ее части
	5.3. Разработка и оформление документации на поставку изделий для комплектования АС и (или) технических требований (технических заданий) на их разработку
	5.4. Разработка заданий на проектирование в смежных частях проекта объекта автоматизации
6. Рабочая документация	6.1. Разработка рабочей документации на систему и ее части
	6.2. Разработка или адаптация программ

1	2
7. Ввод в действие	7.1. Подготовка объекта автоматизации к вводу АС в действие
	7.2. Подготовка персонала
	7.3. Комплектация АС поставляемыми изделиями (программными и техническими средствами, программно-техническими комплексами, информационными изделиями)
	7.4. Строительно-монтажные работы
	7.5. Пусконаладочные работы
	7.6. Проведение предварительных испытаний
	7.7. Проведение опытной эксплуатации
	7.8. Проведение приемочных испытаний
8. Сопровождение АС	8.1. Выполнение работ в соответствии с гарантийными обязательствами
	8.2. Послегарантийное обслуживание

Практически все перечисленные стадии и этапы до сих пор встречаются в практике создания ИС предприятий и организаций.

Кроме того, ГОСТ 34.601 содержит Приложение 1 (справочное) с поэтапной расшифровкой работ и Приложение 2, в котором приведен перечень организаций, участвующих в работах по созданию АС. Это подсказывает способ адаптации стандарта к конкретным условиям: достаточно переработать эти приложения, и получится вполне разумный корпоративный стандарт на создание ИС.

В начале проектирования разрабатывают документацию, достаточную для утверждения намеченных затрат, доходов, дополнительной численности персонала, дополнительных площадей и организационно-технических решений, а при дальнейшем проектировании – для заказа и комплектации оборудования и материалов, его монтажа и наладки, для организации работы АСУ и ее внедрения.

Проектирование АС должна выполнять специализированная организация-проектировщик АС, соисполнителем может быть генпроектировщик отрасли. Имеют отношение к документированию три стадии:

- 1) эскизный проект (ЭП);
- 2) технический проект (ТП);
- 3) разработка рабочей документации (РД).

В данной практической работе более подробно рассмотрено документирование эскизного проекта.

Эскизный проект согласно ГОСТ 34.601 предназначен для разработки предварительных проектных решений и необходим в крупных проектах, поэтому допускается пропускать эту стадию.

На 4-й стадии «Эскизный проект» на 1-м этапе «Разработка предварительных проектных решений по системе и ее частям» определяют функции АС и ее подсистем, состав решаемых в них задач, концепцию и структуру информационной базы, функции СУБД и основных программных средств, состав вычислительной системы.

На 2-м этапе «Разработка документации на АС и ее части» выполняют разработку, оформление, согласование и утверждение документации, определенной в стандарте и полностью описывающей принятые проектные решения. Работы на стадии, как правило, проводят специализированные организации.

Содержание эскизного проекта задается в техническом задании (ТЗ) на систему. Как правило, на этапе эскизного проектирования определяются:

- функции ИС;
- функции подсистем, их цели и ожидаемый эффект от внедрения;
- задача (комплексы задач);
- концепция информационной базы и ее укрупненная структура;
- функции системы управления базой данных;
- состав вычислительной системы и других технических средств;
- функции и параметры основных ПС.

Результаты работы представляют в виде комплекта проектных документов по ГОСТ 34.201 [15] в объеме, установленном в ТЗ (договоре) на проведение данного этапа.

ГОСТ 34.602. Техническое задание на создание автоматизированной системы

Стандарт описывает правила оформления ТЗ. Так как он тесно связан с другими стандартами, при составлении ТЗ желательно также придерживаться приведенных в них рекомендаций, даже если нет прямых на то указаний. Требование «подготовить ТЗ в соответствии с ГОСТ 34.602» означает, что данный документ должен иметь определенную структуру:

1. Общие сведения.
2. Назначение и цели создания (развития) системы.
3. Характеристика объектов автоматизации.
4. Требования к системе.
5. Состав и содержание работ по созданию системы.
6. Порядок контроля и приемки системы.
7. Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу системы в действие.

8. Требования к документированию.

9. Источники разработки.

Относительно ГОСТ 34.602 можно сказать, что он раскрывает содержание вышеперечисленных девяти разделов. Стоит подчеркнуть, что контролировать соответствие ему ТЗ может обычный управленец, в задачу которого входит, например взаимодействие с субподрядчиками. Это значительно упрощает внедрение и практическое применение стандарта.

РД 50-34.698-90. Требования к содержанию документов

Это нормативный документ, с различной степенью детальности описывающий содержание проектных документов. Данный документ определяет состав и структуру документов, введенных в ГОСТ 34.201 [15], вплоть до форматов приказов о начале опытной эксплуатации и вводе в промышленную эксплуатацию.

Задания для самостоятельного выполнения

1. Ознакомьтесь с ГОСТ 34.601 [7].

2. В соответствии с требованиями данного стандарта разработайте эскизный проект выбранной ИС (прил. А).

Требования к оформлению отчета

В качестве отчета о проделанной работе выступает оформленный согласно ГОСТ 34.601 [16] эскизный проект, а также ответы на контрольные вопросы.

Контрольные вопросы

1. Что такое ГОСТ 34? Какие стандарты входят в ГОСТ 34? Какую деятельность регламентирует ГОСТ 34?

2. В чем заключаются основные достоинства ГОСТ 34?

3. Каковы недостатки ГОСТ 34?

4. Для чего имеет смысл применять ГОСТ 34 сейчас?

Практическое занятие № 7

РАЗРАБОТКА ЭКСПЛУАТАЦИОННОЙ ПРОГРАММНОЙ ДОКУМЕНТАЦИИ

Цель занятия

Ознакомиться с процедурой составления пользовательской (эксплуатационной) документации к ПО.

Основные теоретические сведения

Сведения для обеспечения функционирования и эксплуатации программ (компонентов, комплексов) приводятся в эксплуатационной программной документации (ЭД). Комплектность эксплуатационной документации на программные средства определяется по ГОСТ 19.101 «ЕСПД. Виды программ и программных документов» [2].

Состав комплекта ЭД на программу зависит от её архитектуры, назначения и особенностей целевой аудитории. Необходимость составления того или иного документа определяется на этапе разработки и утверждения технического задания на программу. Ниже в табл. 7.1 приведён перечень ЭД на программы.

Таблица 7.1

Виды эксплуатационной документации

Вид эксплуатационного документа	Код вида документа	Дополнительные указания
1	2	3
Ведомость эксплуатационных документов	20	В документе приводят перечень эксплуатационных документов на программу. Выполняется в соответствии с требованиями ГОСТ 19.507 «Ведомость эксплуатационных документов»
Формуляр	30	В документе указывают основные характеристики программы, комплектность и общие сведения об эксплуатации программы. Выполняется в соответствии с требованиями ГОСТ 19.501 «Формуляр. Требования к содержанию и оформлению»

Окончание табл. 7.1

1	2	3
Описание применения	31	В документе приводят сведения о назначении программы, области применения, применяемых методах, классе решаемых задач, ограничениях для применения, минимальной конфигурации технических средств, входных и выходных данных. Выполняется в соответствии с требованиями ГОСТ 19.502 «Описание применения. Требования к содержанию и оформлению»
Руководство системного программиста	32	В документе приводят сведения для установки, проверки, обеспечения функционирования, интеграции в систему и настройки программы в определённых условиях применения ее, устранения аварийных ситуаций. Требования к содержанию и оформлению – по ГОСТ 19.503 «Руководство системного программиста. Требования к содержанию и оформлению»
Руководство программиста	33	В документе приводят сведения по эксплуатации (сопровождению) программы. Выполняется по ГОСТ 19.504 «Руководство программиста. Требования к содержанию и оформлению»
Руководство оператора	34	Документ содержит сведения о порядке действий оператора при использовании программы. Требования к содержанию и оформлению – по ГОСТ 19.505 «Руководство оператора. Требования к содержанию и оформлению»
Описание языка	35	Документ содержит описание синтаксиса и семантики языка, элементов и конструкций, встроенных функций. Выполняется по ГОСТ 19.506 [27]
Руководство по техническому обслуживанию	46	В документе приводят сведения для применения тестовых и диагностических программ при обслуживании технических средств

Эксплуатационная документация – это вид технической документации, описывающий порядок установки, настройки и использования разработанного программного обеспечения (автоматизированной системы).

Как правило, документы данного вида являются составляют либо дополнение к «ядру» комплекта технической документации (проектной документации), либо полностью независимы и автономны, а в некоторых случаях являются единственной частью документации на программный продукт. Именно эксплуатационную документацию чаще всего подразумевают, когда говорят о программной документации в целом.

Правила оформления программных документов для печатного способа выполнения установлены ГОСТ 19.106 [11].

В стандартах ЕСПД отсутствуют методические указания о том, как разработать документацию, они дают только перечень типов документов со списком разделов первого уровня для каждого и содержат указания о том, какие сведения должны быть в нем изложены. Среди стандартов ИСО/МЭК есть ряд документов, касающихся процессов документирования при разработках в сфере информационных технологий. В отличие от ЕСПД, они содержат минимум требований к составу и структуре документов, при этом в них дано множество указаний, направленных на получение документов высокого качества. Возможно, комплексное применение указанных нормативных документов при разработке эксплуатационной документации на программы позволит повысить качество, информативность и полезность таких документов.

Существуют три комплекса стандартов, по которым разрабатывается эксплуатационная документация: ЕСКД (ГОСТ 2), ЕСПД (ГОСТ 19) и КСАС (ГОСТ 34).

Каждая из вышеперечисленных серий ГОСТ предъявляет свои требования к составу, порядку разработки и ведения комплектов эксплуатационной документации, а также к содержанию и оформлению отдельных документов.

Ниже в табл. 7.2 включены международные стандарты (ИСО/МЭК), содержащие требования к процессу разработки программной и системной документации.

Таблица 7.2

Перечень международных стандартов

Обозначение	Наименование
ГОСТ Р ИСО/МЭК 12207	Информационная технология. Процессы жизненного цикла программных средств
ГОСТ Р ИСО/МЭК ТО 15271	Информационная технология. Руководство по применению ГОСТ Р ИСО/МЭК 12207 (Процессы жизненного цикла программных средств)
ГОСТ Р ИСО/МЭК 9126	Информационная технология. Оценка программной продукции. Характеристики качества и руководство по их применению
ГОСТ Р ИСО/МЭК 15910	Информационная технология. Процесс создания документации пользователя программного средства
ГОСТ Р ИСО/МЭК ТО 9294	Информационная технология. Руководство по управлению документированием программного обеспечения

1	2
ГОСТ Р ИСО/МЭК 15288	Информационная технология. Системная инженерия. Процессы жизненного цикла систем
ISO/IEC 15289	Системная и программная инженерия. Содержание информационных продуктов (документации) процессов жизненного цикла систем и программных средств
ISO/IEC 26514	Системная и программная инженерия. Требования для проектировщиков и разработчиков документации пользователя
ISO/IEC 26513	Системная и программная инженерия. Требования по экспертизе и тестированию документации пользователя
ГОСТ Р 51904	Программное обеспечение встроенных систем. Общие требования к разработке и документированию
ISO/IEC 18019:2004	Программная инженерия. Руководство по разработке и подготовке пользовательской документации на прикладные программные средства
ISO 6592:2000	Обработка информации. Руководство по документации для вычислительных систем
ГОСТ Р ИСО 9127	Системы обработки информации. Документация пользователя и информация на упаковке для потребительских программных пакетов.

Кроме вышперечисленных существуют зарубежные нормативно-правовые документы:

End user – руководства для пользователей, администраторов и обслуживающего персонала (аналоги – Руководство пользователя, Руководство администратора);

Technical – описание кода, алгоритмов и интерфейсов (аналоги – Руководство программиста, Справочник разработчика).

Задания для самостоятельного выполнения

1. Ознакомьтесь с ГОСТ 19.503 [18], ГОСТ 19.504 [19] и ГОСТ 19.505 [20].
2. Разработайте документацию в соответствии с требованиями, указанными в приведенных выше стандартах.

Руководство программиста

Согласно ГОСТ 19.504 [19] «Руководство программиста» должно содержать следующие разделы:

1. Назначение и условия применения программ.
2. Характеристика программы.

3. Обращение к программе.
4. Входные и выходные данные.
5. Сообщения.

В разделе «**Назначение и условия применения программ**» должны быть указаны назначение программы и выполняемые ей функции, объем оперативной памяти, требования к составу и параметрам периферийных устройств, требования к программному обеспечению и т. п.

В разделе «**Характеристика программы**» должно быть приведено описание основных характеристик и особенностей программы (временные характеристики, режим работы, средства контроля правильности выполнения и самовосстанавливаемости программы и т. п.).

В разделе «**Обращение к программе**» – описание процедур вызова программы (способы передачи управления и параметров данных и др.).

В разделе «**Входные и выходные данные**» – описание организации используемой входной и выходной информации и (при необходимости) ее кодирования.

В разделе «**Сообщения**» – тексты сообщений, выдаваемых программисту или оператору в ходе выполнения программы, описание их содержания и действий, которые необходимо предпринять по этим сообщениям.

Руководство оператора

Согласно ГОСТ 19.505 [20] «Руководство оператора» должно содержать следующие разделы:

1. Назначение программы.
2. Условия выполнения программы.
3. Выполнение программы.
4. Сообщения оператору.

В зависимости от особенностей документа допускается объединять отдельные разделы или вводить новые.

В разделе «**Назначение программы**» должны быть указаны сведения о назначении программы и информация, достаточная для понимания функций программы и ее эксплуатации.

В разделе «**Условия выполнения программы**» – условия, необходимые для выполнения программы (минимальный и (или) максимальный состав аппаратурных и программных средств и т. п.).

В разделе «**Выполнение программы**» – последовательность действий оператора, обеспечивающих загрузку, запуск, выполнение и завершение программы. Помимо этого, здесь даются описания функций, формата

и возможных вариантов команд, с помощью которых оператор осуществляет загрузку и управляет выполнением программы, а также ответы программы на эти команды.

В разделе «**Сообщения оператору**» должны быть приведены тексты сообщений, выдаваемых в ходе выполнения программы, описание их содержания и соответствующие действия оператора (действия оператора в случае сбоя, возможности повторного запуска программы и т. п.).

Требования к оформлению отчета

В качестве отчета о проделанной работе выступают оформленные согласно ГОСТ 19.504 и ГОСТ 19.505 «Руководство программиста» и «Руководство оператора».

Контрольные вопросы

1. Какие виды эксплуатационной документации существуют?
2. Для каких целей разрабатывается пользовательская документация?
3. Из каких разделов состоит ГОСТ 19.505 «Руководство оператора»?
4. Опираясь на какой стандарт, возможно разработать «Руководство пользователя»?

Практическое занятие № 8

СЕРТИФИКАЦИЯ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Цель занятия

Ознакомиться с процедурой сертификации информационных систем, технологий, а также средств их обеспечения.

Основные теоретические сведения

Рынок средств и систем информатизации в России сейчас настолько разнообразен, что в подавляющем большинстве случаев потребитель не в состоянии самостоятельно убедиться в соответствии приобретаемой им продукции установленным на государственном уровне нормам и правилам. Положение усугубляется тем обстоятельством, что российский рынок заполнен импортными изделиями. Для этих изделий производители и поставщики в лучшем случае декларируют соответствие отдельным зарубежным стандартам, о содержании которых у вас, как правило, нет никакой информации.

На бытовом уровне логичным путем решения этой проблемы является обращение к некоторому третьему лицу, являющемуся специалистом в данной области и заведомо независимому от поставщика продукции, которое может дать заключение о соответствии продукции установленным требованиям. На государственном уровне аналогичная процедура называется сертификацией.

Сертификация – процедура, выполняемая третьей стороной, независимой от изготовителя (продавца) и потребителя продукции или услуг, по подтверждению соответствия этих продукции или услуг установленным требованиям.

Результатом выполнения процедуры сертификации является так называемый сертификат соответствия.

Сертификат соответствия – документ, выданный по правилам системы сертификации для подтверждения соответствия сертифицированной продукции установленным требованиям.

Общие правовые основы сертификации продукции и услуг в Российской Федерации прописаны в статье 25 Федерального закона «О техническом регулировании» от 27 декабря 2002 г. № 184-ФЗ [21], где определены права и ответственность в области сертификации органов

государственного управления, а также изготовителей (продавцов, исполнителей) и других участников сертификации.

Сертификация средств и систем информатизации является элементом общей системы сертификации продукции в Российской Федерации.

Основными целями сертификации средств информатизации, информационных технологий и услуг являются:

- защита пользователей средств и систем информатизации от приобретения средств и систем, в том числе импортных, которые представляют опасность для жизни, здоровья, имущества, а также для окружающей среды;

- обеспечение разработчиков систем, а также широкого круга пользователей этих систем достоверной информацией о состоянии отечественного и зарубежного рынков средств информатизации, телекоммуникаций, информационных технологий и услуг;

- гарантирование информационного обмена между государственными системами информатизации;

- создание условий для информационного взаимодействия субъектов негосударственной принадлежности с субъектами государственной принадлежности;

- содействие повышению научно-технического уровня и конкурентоспособности отечественных систем информатизации, информационных технологий и услуг;

- помощь в создании условий для вхождения России в мировое информационное пространство.

Необходимо отметить, что сертификация средств информатизации не только обеспечивает удовлетворение интересов потребителя, но и приносит определенные выгоды и изготовителю (поставщику) продукции. Так, в частности, сертификация способствует расширению рынка сбыта (распространению продукции в тех районах, где потребителю неизвестна репутация фирмы) и обеспечивает подтверждение качества продукции фирмы по сравнению с продукцией конкурентов. С точки зрения организации торговых взаимосвязей сертификация способствует созданию доверительных отношений между производителями (поставщиками) и потребителями продукции. Необходимо иметь в виду, что только имеющее место и объективно подтвержденное качество конкретных видов отечественной информационной продукции и средств информатизации может сделать их конкурентоспособными и реально обеспечить спрос на них.

Говоря о сертификации, нельзя не отметить ее тесную взаимосвязь со стандартизацией в сфере информатизации.

Во-первых, как уже говорилось выше, суть процедуры сертификации заключается в подтверждении соответствия средств информатизации установленным требованиям. Документами, содержащими эти требования, являются стандарты, разрабатываемые в процессе стандартизации.

Во-вторых, собственно процедура сертификации регламентируется действующими нормативными документами (стандартами).

Согласно рекомендациям по разработке технических регламентов Р 50.1.046 «Рекомендации по выбору форм и схем обязательного подтверждения соответствия продукции при разработке технических регламентов» [22] предлагаются следующие схемы обязательного подтверждения соответствия.

Подтверждение соответствия продукции требованиям ТР в рамках установленной формы обязательного подтверждения соответствия осуществляется согласно схемам обязательного подтверждения соответствия (далее – схемы), каждая из которых представляет собой полный набор операций и условий их выполнения участниками подтверждения соответствия.

Схемы могут включать одну или несколько операций, результаты которых необходимы для подтверждения соответствия продукции установленным требованиям, в том числе:

- испытания (типовых образцов, партий или единиц продукции);
- сертификацию системы качества (на стадиях проектирования и производства, только производства или при окончательном контроле и испытаниях);
- инспекционный контроль.

Схемы в ТР на конкретные виды продукции рекомендуется выбирать таким образом, чтобы они не были излишне обременительными в сравнении с целями ТР.

В техническом регламенте рекомендуется по возможности устанавливать для одной и той же продукции несколько схем, равнозначных по степени доказательности. Это позволит заявителю выбрать наиболее приемлемую для него схему (см. пункт 1 статьи 28 Федерального закона «О техническом регулировании»).

При этом в ТР на продукцию, подпадающую по наименованию под соответствующую европейскую директиву, желательно брать за основу схемы, близкие к процедурам оценки соответствия, установленным в этой директиве.

Кроме того, в ТР можно устанавливать дополнительные требования по сравнению с соответствующей схемой, приведенной в настоящих рекомендациях, если этого требуют особые условия, например специфика продукции, сектор рынка.

Правила заполнения документов, используемых при сертификации

При проведении сертификации следует использовать установленные в соответствии с ГОСТ Р 57619 «Оценка соответствия. Рекомендации по содержанию и применению форм документов, используемых при добровольной сертификации услуг (работ)» [23] формы документов.

Заявка на проведение сертификации услуг (работ) в общем виде должна содержать следующую информацию:

- наименование заявителя на сертификацию, его адреса и банковские реквизиты;
- наименование услуг (работ) и коды по общероссийским классификаторам, если они предусмотрены системой сертификации;
- перечень нормативных и технических документов, на соответствие которым будет проведена сертификация услуг (работ);
- обязательство заявителя соблюдать требования к сертификации и предоставлять любую информацию, необходимую для оценки;
- дополнительную информацию, необходимую для объективной оценки заявителя на соответствие установленным требованиям.

Заявитель несет ответственность за достоверность информации.

Рекомендуемая форма заявки на проведение сертификации услуг (работ) приведена в прил. Б.

Дополнительная информация, необходимая для объективной оценки заявителя на соответствие установленным требованиям, может быть оформлена в виде анкеты-вопросника, если это предусмотрено документами системы добровольной сертификации.

Формы сертификата соответствия на услуги (работы) и приложения к нему, а также правила их заполнения устанавливаются в конкретной системе добровольной сертификации. Рекомендуемые формы сертификата соответствия и приложения к нему приведены в прил. В и Г.

Задания для самостоятельного выполнения

1. Ознакомьтесь с процедурой разработки и оформления документов сертификации.

2. Проведите сертификацию разработанного Вами программного продукта, используя следующие формы документов:

- заявка (прил. Б);
- сертификат соответствия (прил. В) и приложение к нему (прил. Г).

Требования к оформлению отчета

В качестве отчета о проделанной работе выступают оформленные в соответствии с ГОСТ 57619 [23] заявка, сертификат соответствия и приложение к нему.

Контрольные вопросы

1. Согласно Федеральному закону «О техническом регулировании» от 27.12.2002 № 184-ФЗ [21] с какой целью принимаются технические регламенты?

2. Какие группы документов входят в нормативную базу сертификации средств и систем информатизации?

3. В каких основных направлениях проводится сертификация средств информатизации?

4. Перечислите средства информатизации, которые подлежат обязательной сертификации согласно «Номенклатуре продукции и услуг, подлежащих обязательной сертификации в Российской Федерации».

5. Ознакомьтесь с текстом Федерального закона «Об информации, информационных технологиях и защите информации» от 27 июля 2006 г. № 149-ФЗ. Какие цели защиты информации определяет данный закон?

Практическое занятие № 9

ЛИЦЕНЗИРОВАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Цель занятия

Ознакомиться с основными понятиями и терминами в области лицензирования, процедурой составления лицензионного соглашения конечного пользователя программного продукта.

Основные теоретические сведения

Основным отличием процесса лицензирования от процесса сертификации является состав категорий, по отношению к которым они применяются. В процессе лицензирования фигурируют такие категории, как «деятельность» (подразумеваются виды или направления деятельности) и «субъект» (физическое лицо, предприятие, организация или иное юридическое лицо).

В соответствии с действующим законодательством в Российской Федерации отдельные виды деятельности осуществляются предприятиями, организациями и учреждениями независимо от организационно-правовой формы, а также физическими лицами, осуществляющими предпринимательскую деятельность без образования юридического лица, на основании лицензии – специального разрешения органов, уполномоченных на ведение лицензирования.

Лицензия является официальным документом, который разрешает осуществление указанного в нем вида деятельности в течение установленного срока, а также определяет условия его осуществления.

Основу нормативно-правовой базы лицензирования в сфере информатизации составляют Федеральные законы: «О лицензировании отдельных видов деятельности», «Об информации, информатизации и защите информации».

Общие принципы лицензирования видов деятельности в сфере информатизации России можно сформулировать следующим образом:

– целью лицензирования является защита интересов государства и граждан от неумышленного или сознательного некачественного выполнения работ, соответствующих определенным видам деятельности в сфере информатизации;

– виды деятельности в сфере информатизации, подлежащие лицензированию, а также органы, осуществляющие лицензирование конкретных видов деятельности в различных областях информатизации, определены рядом нормативных документов;

– право на осуществление деятельности, подлежащей лицензированию, может получить субъект, отвечающий определенным критериям, которые заранее определяются правилами проведения лицензирования и являются их неотъемлемой частью требований к предприятию-заявителю. Таким образом, субъектом лицензирования становится лишь то физическое или юридическое лицо, которое представляет все необходимые и правильно оформленные документы и удовлетворяет соответствующим требованиям.

За органом, уполномоченным на проведение лицензионной деятельности, закрепляется право на осуществление контроля за деятельностью лицензиата.

Лицензионное соглашение (*License agreement*, Лицензионный договор) – договор, по которому одна сторона (лицензиар) предоставляет право на использование изобретения или иного технического достижения (лицензию), а другая сторона (лицензиат) выплачивает за это определенное вознаграждение.

Рекомендации по оформлению договора

Договор неисключительной (простой) лицензии заключается, если передаётся в пользование результат интеллектуальной деятельности (программа ЭВМ, изображение, фотография и др.) и автор (правообладатель) оставляет за собой право предоставлять лицензию и иным лицам.

Лицензиар (тот, кто предоставляет лицензию) может заключать другие лицензионные договоры на этот же объект интеллектуальной деятельности с третьими лицами о том же способе использования в отношении той же территории, которые предусмотрены в договоре неисключительной лицензии.

Лицензиар также может сам осуществлять исключительное право на этот же результат интеллектуальной деятельности (РИД), в том числе путем его использования теми же способами и на той же территории, что и лицензиат.

Преамбула договора

В данном разделе (рис. 9.1) необходимо указать место и дату заключения соглашения, а также реквизиты сторон. Сторонами в данном случае будут выступать:

Лицензиар – лицо, предоставляющее другому лицу право на использование объекта лицензии.

Лицензиат – лицо, приобретающее у собственника (лицензиара) право использования результата интеллектуальной деятельности.

При этом сторонами могут выступать как физические лица, так и субъекты предпринимательской деятельности (юридические лица и индивидуальные предприниматели).

Предмет договора

К существенным условиям лицензионного договора можно отнести условия о предмете договора.

Предмет договора – это описание объекта интеллектуальной собственности. Указываются название РИД и его основные характеристики: жанр, тематика, исключительные особенности произведения, язык программирования и функционал программы, особенности построения и назначения базы данных, а также иные сведения, позволяющие идентифицировать результат интеллектуальной деятельности. Если права на предмет зарегистрированы в Роспатенте, то указываются данные свидетельства о регистрации результата интеллектуальной деятельности.

По закону Лицензионный договор признается заключенным на условиях простой (неисключительной) лицензии, если иное не предусмотрено договором. Но в любом случае рекомендуется в самом тексте договора упомянуть о том, что лицензия является неисключительной, чтобы между сторонами не возникало противоречий.

Права и обязанности сторон

Ещё одно обязательное условие, которое нужно прописать в тексте соглашения – это способы использования объекта.

Способы использования РИД определяют конкретные правомочия Лицензиата по использованию объекта. Например, могут быть указаны такие права:

- воспроизведение, т. е. изготовление одного и более экземпляра результата интеллектуальной деятельности или его части в любой материальной форме;

- распространение путем продажи или иного отчуждения его оригинала или экземпляров;

- публичный показ, прокат, публичное исполнение, сообщение в эфир;

– право на внесение изменений, осуществляемых исключительно в целях функционирования, усовершенствования и модификации и др.

Использовать РИД можно только теми способами, которые указаны в договоре.

В договоре можно обозначить территорию использования прав. Если таковая в договоре не указана, Лицензиат вправе осуществлять использование РИД на всей территории Российской Федерации.

Здесь же указывается, на какой срок Лицензиар предоставляет права на использование объекта интеллектуальной деятельности.

Кроме того, можно прописать в соглашении обязанность Лицензиата предоставлять отчёты об использовании РИД ежемесячно, ежеквартально, ежегодно или в иной срок. Можно указать, что отчёты предоставляются по требованию Лицензиара.

Порядок выплаты вознаграждения

Исключительная лицензия обычно предполагается возмездной, поэтому в договоре необходимо определить размер вознаграждения в виде:

- разового платежа с фиксированной суммой;
- периодических платежей с фиксированной суммой;
- процентов от дохода лицензиата.

Обязательно пропишите срок, в течение которого должно быть выплачено вознаграждение. Можно также указать способ выплаты: обычно это происходит путём перечисления денежных средств на расчетный счет Лицензиара.

Ответственность сторон

В этом разделе можно прописать, что в качестве меры ответственности за нарушение прописанных в договоре обязательств, включая расторжение договора в одностороннем порядке, взыскивается неустойка, размер которой определяется нормами Гражданского кодекса РФ (законная неустойка). Кроме того, стороны могут прописать в договоре иной размер неустойки и порядок её исчисления – в виде процента от суммы вознаграждения за каждый день просрочки. Но не рекомендуется указывать слишком большой процент, так как суд вправе уменьшить подлежащую уплате неустойку, если она явно несоразмерна с последствиями нарушения обязательств.

Задание для самостоятельного выполнения

1. Ознакомьтесь с процедурой разработки и оформления лицензионных соглашений.

2. Опираясь на приведенный ниже пример и используя форму из прил. Д, составьте лицензионное соглашение на разработанный Вами программный продукт.

Пример лицензионного договора

Лицензионный договор (неисключительная лицензия) № 3

Москва

8 апреля 2021 г.

Настоящий лицензионный договор (неисключительная лицензия) (далее – **Договор**) заключен между: Ивановым Иваном Ивановичем, паспорт: 3245 № 736876, выдан Отделением УФМС России по городу Москве по району Тверской 05 марта 2013 г., код подразделения 170-453, зарегистрированным по адресу: 456466, г. Москва, ул. Зеленоградская, д. 31А/1 (далее – «**Лицензиар**») и Обществом с ограниченной ответственностью «Пример» в лице директора Петровой М. Н., действующего на основании устава (далее – «**Лицензиат**»). Лицензиар и Лицензиат совместно именуются **Стороны**, а по отдельности **Сторона**.

1. Предмет договора

1.1. **Лицензиар** предоставляет **Лицензиату** неисключительные права на использование **Программы для ЭВМ «Управление базой данных документов»** (свидетельство: номер ук0924РП; дата выдачи: 4 февраля 2021 г.) в обусловленных настоящим **Договором** пределах и на определенный **Договором** срок (неисключительная лицензия), а **Лицензиат** за предоставление этих прав уплачивает вознаграждение **Лицензиару**.

1.3. **Лицензиар** гарантирует наличие у него предоставляемых по настоящему **Договору** исключительных прав на **Программу для ЭВМ «Управление базой данных документов»**. Описание: **Программа для ЭВМ «Управление базой данных документов»**. Язык программирования – JavaScript. Функционал программы: добавление, редактирование и удаление информации. Формирование отчета по выделенным полям, поиск данных по различным критериям.

2. Права и обязанности сторон

2.1. По настоящему **Договору Лицензиар** предоставляет **Лицензиату** следующие права использования **Программы для ЭВМ «Управление базой данных документов»**:

2.1.1. Право использовать **Программу для ЭВМ «Управление базой данных документов»** под своим фирменным наименованием или товарным знаком.

2.1.2. Право на воспроизведение **Программы для ЭВМ «Управление базой данных документов»**. Под воспроизведением понимается изготовление ни более одного экземпляра – результата интеллектуальной деятельности или его части в любой материальной форме, в том числе в форме звуко- или видеозаписи, изготовление в трех измерениях одного и более экземпляра двухмерного произведения и в двух измерениях одного и более экземпляра трехмерного произведения. При этом запись произведения на электронном носителе, в том числе запись в память ЭВМ, также считается воспроизведением, кроме случая, когда такая запись является временной и составляет неотъемлемую и существенную часть технологического процесса, имеющего единственной целью правомерное использование записи или правомерное доведение произведения до всеобщего сведения.

2.1.3. Распространение **Программы для ЭВМ «Управление базой данных документов»** путем продажи или иного отчуждения ее оригинала или экземпляров.

2.1.4. Публичный показ **Программы для ЭВМ «Управление базой данных документов»**.

2.1.5. Публичное исполнение **Программы для ЭВМ «Управление базой данных документов»**.

2.1.6. Право предоставления третьим лицам права пользования **Программой для ЭВМ «Управление базой данных документов»** (заключение sublicензионного Договора).

2.2. Права на использование **Программы для ЭВМ «Управление базой данных документов»**, указанные в п. 2.1 настоящего Договора, передаются **Лицензиаром** **Лицензиату** для использования на всей территории РФ.

2.3. Права на использование **Программы для ЭВМ «Управление базой данных документов»**, указанные в п. 2.1 настоящего Договора, передаются **Лицензиаром** **Лицензиату** для использования на срок в течение 3-х лет с даты подписания настоящего Договора.

2.4. **Лицензиар** вправе предоставлять права использования **Программы для ЭВМ «Управление базой данных документов»**, в том числе в части переданных **Лицензиату** прав посредством заключения лицензионных Договоров с третьими лицами.

2.5. **Лицензиат** обязан представлять **Лицензиару** письменные отчеты об использовании **Программы для ЭВМ «Управление базой данных документов»** ежемесячно.

3. Передача прав и порядок выплаты вознаграждения

3.1. За использование **Программы для ЭВМ «Управление базой данных документов»** по настоящему Договору **Лицензиат** выплачивает **Лицензиару** вознаграждение.

3.2. Обязательства **Лицензиата** по уплате вознаграждения считаются исполненными с момента получения денежных средств **Лицензиаром**.

3.3. Вознаграждение **Лицензиару** составляет 250 000 (двести пятьдесят тысяч) рублей 00 копеек и выплачивается единовременно в течение 5 (пяти) банковских дней с момента передачи **Лицензиату Программы для ЭВМ «Управление базой данных документов»**. Оплата включает в себя стоимость всего срока использования **Лицензиатом** программы.

3.4. **Лицензиат** приобретает права, указанные в п. 2.1. настоящего Договора, с момента его подписания. **Программа для ЭВМ «Управление базой данных документов»** передается **Лицензиату** посредством предоставления доступа в сети Интернет.

3.5. Реквизиты доступа, логины, пароли и иные данные сообщаются дополнительно по электронной почте в течение 3-х дней с момента подписания Договора.

3.6. Все расчеты по настоящему Договору производятся в безналичной форме путем перечисления денежных средств с расчетного счета **Лицензиата** на расчетный счет **Лицензиара**. Датой совершения платежа считается дата зачисления денежных средств на расчетный счет **Лицензиара**.

4. Ответственность сторон

4.1. **Лицензиар** гарантирует, что:

4.1.1. Он является законным правообладателем **Программы для ЭВМ «Управление базой данных документов»**.

4.1.2. На момент вступления в силу настоящего Договора **Лицензиару** ничего не известно о правах третьих лиц, которые могут быть нарушены подписанием настоящего Договора.

4.1.3. На момент заключения настоящего **Договора** права, указанные в п. 2.1 настоящего **Договора**, не отчуждены и не заложены.

4.1.4. На момент заключения настоящего **Договора** права **Лицензиара** не оспорены в суде или иным законным способом.

4.2. При нарушении сторонами своих обязательств по **Договору**, стороны несут ответственность в порядке, предусмотренном действующим законодательством.

4.3. При нарушении **Лицензиаром** п. 3.4. настоящего **Договора**, **Лицензиат** вправе в одностороннем порядке отказаться от настоящего **Договора** и потребовать уплаты неустойки в размере 0,1 % от суммы вознаграждения за каждый день просрочки.

4.4. При нарушении **Лицензиатом** обязанности уплатить **Лицензиару** в установленный срок вознаграждение за предоставление лицензии, **Лицензиар** вправе в одностороннем порядке отказаться от настоящего **Договора** с отзывом права использования **Лицензии** у **Лицензиата**, и потребовать уплаты неустойки в размере 0,1 % от суммы вознаграждения за каждый день просрочки.

Требования к оформлению отчета

Результатом выполнения данной практической работы является оформленное лицензионное соглашение на разработанный программный продукт согласно приведенным выше критериям.

Контрольные вопросы

1. Какие виды деятельности должно ограничивать лицензирование?
2. Ознакомьтесь с лицензированием деятельности в области создания и применения информационных технологий. Какие основные законы регулируют данную деятельность?
3. Ознакомьтесь с лицензированием деятельности в области формирования и ведения информационных ресурсов. Какие основные законы регулируют данную деятельность?
4. Какие нормативно-правовые документы определены в качестве основных государственных органов по лицензированию деятельности в области защиты информации?
5. Перечислите виды деятельности в области защиты информации, подлежащие лицензированию Гостехкомиссией России.
6. Что включает в себя деятельность по международному информационному обмену?

Практическое занятие № 10

ТЕСТИРОВАНИЕ И ОТЛАДКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ. ШАБЛОНЫ СОПРОВОДИТЕЛЬНЫХ ДОКУМЕНТОВ

Цель занятия

Ознакомиться с документацией и процедурой проведения тестирования.

Основные теоретические сведения

Тестирование (*software testing*) – деятельность, выполняемая для оценки и улучшения качества программного обеспечения. Эта деятельность, в общем случае, базируется на обнаружении дефектов и проблем в программных системах. Тестирование программных систем состоит из динамической верификации поведения программ на конечном (ограниченном) наборе тестов (*set of test cases*), выбранных соответствующим образом из обычно выполняемых действий прикладной области и обеспечивающих проверку соответствия ожидаемому поведению системы.

Тестирование – запуск исполняемого кода с тестовыми данными и исследование выходных данных и рабочих характеристик программного продукта для проверки правильности работы системы.

Тестирование – процесс выполнения программы на некотором наборе данных, для которого заранее известны результат применения или правила поведения этих программ. Указанный набор данных называется тестовым или просто тестом.

Таким образом, тестирование есть проверка работы программ с данными, подобными реальным, которые будут обрабатываться в процессе эксплуатации системы. Наличие в программе дефектов и несоответствий требованиям обнаруживается путем исследования выходных данных и выявления среди них аномальных.

Объект тестирования (objectives of testing)

Компонент или система, которые должны быть протестированы.

Цели тестирования (the target of the test)

Тестирование проводится в соответствии с определенными целями, которые могут быть заданы явно или неявно, и различным уровнем точности. Определение цели точным образом, выражаемым количественно,

позволяет обеспечить контроль результатов тестирования. Тестовые сценарии могут разрабатываться как для проверки функциональных требований (известны как функциональные тесты), так и для оценки нефункциональных требований. При этом, существуют такие тесты, когда количественные параметры и результаты тестов могут лишь опосредованно говорить об удовлетворении целям тестирования (например, «*usability*» – легкость, простота использования, в большинстве случаев, не может быть явно описана количественными характеристиками).

Обобщенно цели тестирования можно определить следующим образом:

1) обнаружение несоответствий между ПО и его спецификацией, которые обусловлены ошибками или дефектами, т. е. направлено на выявление скрытых дефектов до момента сдачи ПО заказчику. Тестирование дефектов противоположно аттестации, в ходе которой проверяется соответствие системы своей спецификации. Во время аттестации система должна корректно работать со всеми заданными тестовыми данными. При тестировании дефектов запускается такой тест, который вызывает некорректную работу программы и, следовательно, выявляет дефект. Обратите внимание на эту важную особенность: тестирование дефектов демонстрирует наличие, а не отсутствие дефектов в программе;

2) оценка производительности, надежности и работы ПО в различных режимах эксплуатации. В этом случае тесты разрабатываются так, чтобы имитировать реальную работу системы с реальными входными данными.

Уровни тестирования

Тестирование обычно производится на протяжении всей разработки и сопровождения на разных уровнях. Уровень тестирования определяет «над чем» производятся тесты: над отдельным модулем, группой модулей или системой в целом.

1. Модульное тестирование (*Unit testing*). Этот уровень тестирования позволяет проверить функционирование отдельно взятого элемента системы. Что считать элементом – модулем системы определяется контекстом.

2. Интеграционное тестирование (*Integration testing*). Данный уровень тестирования является процессом проверки взаимодействия между программными компонентами/модулями. Интеграционное тестирование – постоянно проводимая деятельность, предполагающая работу на достаточно высоком уровне абстракции. Наиболее успешная практика интеграционного тестирования базируется на инкрементальном подходе.

3. Системное тестирование (*System testing*). Охватывает целиком всю систему. Большинство функциональных сбоев должно быть идентифицировано еще на уровне модульных и интеграционных тестов. В свою очередь, системное тестирование обычно фокусируется на нефункциональных требованиях – безопасности, производительности, точности, надежности и т. п. Также тестируются интерфейсы к внешним приложениям, аппаратному обеспечению, операционной среде и т. д.

Виды тестирования

Приёмочное тестирование (*Acceptance/qualification testing*) проверяет поведение системы на предмет удовлетворения требований заказчика. Это возможно в том случае, если заказчик берет на себя ответственность, связанную с проведением таких работ.

Установочное тестирование (*Installation testing*) предполагает проверку процедуры инсталляции системы в целевом окружении.

Альфа- и бета-тестирование (*Alpha and beta testing*). Перед выпуском ПО, как минимум, должно проходить стадии альфа (внутреннее пробное использование) и бета (пробное использование с привлечением отобранных внешних пользователей) версий.

Функциональные тесты/тесты соответствия (*Conformance testing/Functional testing/Correctness testing*) – проверка соответствия системы предъявляемым к ней требованиям, описанным на уровне спецификации поведенческих характеристик.

Достижение и оценка надежности (*Reliability achievement and evaluation*). Помогая идентифицировать причины сбоев, тестирование подразумевает и повышение надежности программных систем. Случайно генерируемые сценарии тестирования могут применяться для статистической оценки надежности.

Регрессионное тестирование (*Regression testing*) – тестирование ПО, направленное на обнаружение ошибок в уже протестированных участках исходного кода. Такие ошибки, связанные с тем, что после внесения изменений в программу перестает работать то, что должно было работать, называют регрессионными (англ. *regression bugs*).

Определение успешности регрессионных тестов (согласно IEEE 610-90 «Standard Glossary of Software Engineering Terminology») гласит, что «повторное выборочное тестирование системы или компонент для проверки сделанных модификаций не должно приводить к непредусмотренным эффектам».

Тестирование производительности (*Performance testing*) – попытка достижения количественных пределов, обусловленных характеристиками самой системы и ее операционного окружения.

Нагрузочное тестирование (*Stress testing*) проводится с целью достижения максимальных возможностей производительности (вплоть до запланированных характеристик и далее) с отслеживанием поведения на всем протяжении повышения загрузки системы.

Сравнительное тестирование (*Back-to-back testing*) – единичный набор тестов, позволяющих сравнить две версии системы.

Восстановительные тесты (*Recovery testing*). Цель – проверка возможностей рестарта системы в случае непредусмотренной катастрофы, влияющей на функционирование операционной среды, в которой выполняется система.

Конфигурационное тестирование (*Configuration testing*). В случаях, если программное обеспечение создается для использования различными пользователями (в терминах «ролей»), данный вид тестирования направлен на проверку поведения и работоспособности системы в различных конфигурациях.

Тестирование удобства и простоты использования (*Usability testing*) позволяет проверить, насколько легко конечный пользователь системы может ее освоить, включая не только функциональную составляющую – саму систему, но и ее документацию; насколько эффективно пользователь может выполнять задачи, автоматизация которых осуществляется с применением данной системы; наконец, насколько хорошо система застрахована (с точки зрения потенциальных сбоев) от ошибок пользователя.

Разработка, управляемая тестированием (*Test-driven development*). По сути, это не столько техника тестирования, сколько стиль организации процесса разработки, жизненного цикла, когда тесты являются неотъемлемой частью требований (и соответствующих спецификаций) вместо того, чтобы рассматриваться независимой деятельностью по проверке удовлетворения требований программной системой.

Структурное тестирование предполагает создание тестов на основе структуры системы и ее реализации. Такой подход иногда называют тестированием методом «белого ящика» (*White box*), «стеклянного ящика» или «прозрачного ящика», который как бы позволяет заглянуть внутрь программного обеспечения. Как правило, структурное тестирование применяется к относительно небольшим программным элементам, например к подпрограммам или методам, ассоциированным с объектами. Например, из анализа кода можно определить, сколько контрольных тестов нужно

выполнить для того, чтобы в процессе тестирования все операторы были задействованы по крайней мере один раз.

Функциональное тестирование базируется на том, что все тесты основываются на проверке не самого ПО, а только выполняемых им функций. ПО представляется как «черный ящик» (*Black-box*), поведение которого можно определить только посредством изучения ее входных и соответствующих выходных данных.

Комбинированное тестирования сочетает проверку как структуры ПО, так и его функционирования. Такой подход называют тестирование методом «серого ящика» (*Gray-box Testing*).

Деятельность по тестированию (Test Activities)

Тестовые работы – сложные процессы проверки соответствия предъявляемым требованиям. Они представляют собой одну или несколько проверок, эксперименты или исследования для получения информации о качественном атрибуте объекта тестирования. Ниже перечислены этапы, которые рекомендуется соблюдать при организации тестирования ПО.

1. **Планирование** (*Planning*). Необходимо разработать план тестирования для дальнейшего генерирования тест-кейсов.

2. **Генерация сценариев тестирования** (*Test-case generation*) – создание тестовых сценариев основывается на уровне и конкретных техниках тестирования.

3. **Разработка тестового окружения** (*Test environment development*). Окружение должно обеспечивать разработку и контроль тестовых сценариев, ведение журнала тестирования и возможности восстановления ожидаемых и отслеживаемых его результатов, самих сценариев, а также других активов тестирования.

4. **Выполнение тестов** (*Execution*). Выполнение тестов должно содержать основные принципы ведения научного эксперимента:

– должны фиксироваться все работы и результаты процесса тестирования;

– форма журналирования таких работ и их результатов должна быть такой, чтобы соответствующее содержание было понятно, однозначно интерпретируемо и повторяемо другими лицами;

– тестирование должно проводиться в соответствии с заданными и документированными процедурами;

– тестирование должно производиться над однозначно идентифицируемой версией и конфигурацией программной системы.

5. **Анализ результатов тестирования** (*Test results evaluation*) может быть использован для демонстрации особенностей функционирования системы, поиска и устранения недостатков, определения направлений развития и повышения эффективности работы системы.

6. **Отчёты о проблемах/журнал тестирования** (*Problem reporting/Test log*). Во многих случаях, в процессе тестовой деятельности ведётся журнал тестирования, фиксирующий информацию о соответствующих работах: когда проводится тест, какой тест, кем проводится, для какой конфигурации программной системы (в терминах параметров и в терминах идентифицируемой версии контекста конфигурационного управления) и т. п. Неожиданные или некорректные результаты тестов могут записываться в специальной подсистеме ведения отчетности по сбоям (*problem-reporting system*), обеспечивая формирование базы данных, используемой для отладки, устранения проблем и дальнейшего тестирования. Кроме того, аномалии (помехи), которые нельзя идентифицировать как сбои, также могут фиксироваться в журнале и/или системе ведения отчетности по сбоям. В любом случае документирование таких аномалий снижает риски процесса тестирования и помогает решать вопросы повышения надежности самой тестируемой системы.

7. **Отслеживание дефектов** (*Defect tracking*). Дефекты могут (и, чаще всего, должны) анализироваться для определения момента и места первого их появления в системе с указанием соответствующих причин.

Документы тестирования

План тестирования – организационный документ, содержащий требования к тому, как должно выполняться тестирование в данном конкретном проекте. Создается менеджером тестирования (*test manager*).

Тест-требования – документы, в которых подробно описано то, какие аспекты поведения системы должны быть протестированы. На основании описания архитектуры создаются низкоуровневые тест-требования, где раскрываются аспекты поведения конкретной программной реализации системы, которые необходимо протестировать. Создаются разработчиками тестов (*test procedure developers*).

Тест-планы (тестовые спецификации) – документы, которые содержат подробное пошаговое описание того, как должны быть протестированы тест-требования. На основании тест-требований и проектной документации разработчиков также создается тестовое окружение, необходимое для

корректного выполнения тестов на тестовых стендах – драйверы, заглушки, настроечные файлы и т. п. Создаются разработчиками тестов (*test developers*).

Структура тест-плана может соответствовать структуре тест-требований или следовать логике внешнего поведения системы. Каждый пункт тест-плана описывает, как производится проверка правильности функционирования программной реализации, и содержит:

- ссылку на требование(я), которое проверяется этим пунктом;
- конкретное входное воздействие на программу (значения входных данных);
- ожидаемую реакцию программы (тексты сообщений, значения результатов);
- описание последовательности действий, необходимых для выполнения пунктов тест-плана.

Возможные формы подготовки тест-планов:

а) в виде текстовых документов, в которых отдельные разделы представляют собой описания тестовых примеров, каждый пример включает в себя перечисление последовательности действий, которые необходимо выполнить, – сценария теста, а также ожидаемые отклики системы на эти действия;

б) для автоматизированного тестирования сценарий может записываться на формальном языке;

в) таблица используется при четко и формально определенных входных потоках данных системы. Например, каждый столбец таблицы может представлять собой тестовый пример, каждая строка – описание входного потока данных, тогда в ячейку таблицы заносится передаваемое в конкретном тестовом примере в соответствующий поток значение. Ожидаемые значения для данного теста записываются в аналогичной таблице, где в строках перечисляются выходные потоки данных;

г) в виде конечного автомата используется при тестировании протоколов связи или программных модулей, взаимодействие которых с внешним миром производится при помощи обмена сообщениями по заранее заданному интерфейсу.

Отчеты о выполнении тестирования (они могут создаваться либо автоматически, либо вручную), которые содержат информацию о том, какие несоответствия требованиям были выявлены в результате тестирования. Создаются тестировщиками (*testers*).

Отчеты о проблемах – документы, которые направляются на анализ в группу разработчиков с целью определения причины возникновения несоответствия.

Задание для самостоятельного выполнения

1. Сформируйте сценарий системного тестирования разработанного Вами ПО.
2. Уточните спецификации для тестирования отдельных компонентов (функций).
3. Опишите состояние окружения (входные данные) и ожидаемую последовательность событий в системе (ожидаемый результат) в спецификации для каждого тестового случая.
4. Разработайте соответствующие тесты.
5. Протестируйте общую работоспособность и отдельные функциональные возможности разработанного приложения.
6. Выполните верификацию функциональных возможностей разработанного приложения с имеющимся перечнем функциональных требований.

Требования к оформлению отчета

Результатом выполнения данной работы является оформленный отчет о выполнении тестирования.

Контрольные вопросы

1. Что представляет собой сценарий тестирования?
2. Какие виды тестирования обязательно применять в процессе разработки сложного ПО?
3. Расскажите о модульном тестировании. Раскройте понятие модуля.
4. Что представляет собой V-образная модель? Расскажите о статическом и динамическом тестированиях.
5. В чем состоит принципиальное отличие верификации от валидации?
6. Раскройте понятия «тестовый случай» и «тестовое покрытие».

Практическое занятие № 11

ПАСПОРТ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Цель занятия

Ознакомиться с основными понятиями, структурой и правилами составления паспорта ПО.

Основные теоретические сведения

Законодательные нормы диктуют особые требования к процессу разработки и выпуску лицензированного ПО. Согласно действующим законодательным нормам, каждая программа должна иметь индивидуальный паспорт продукта.

Паспорт программного обеспечения – это технический документ, который детально описывает лицензируемый продукт, его параметры, стоимость, системные требования, патенты и др.

Паспорт ПО необходим, чтобы законно и свободно тиражировать лицензионную версию программы на территории РФ и стран СНГ. Помимо этого, в некоторых государственных учреждениях запрещено использовать программы на компьютерных устройствах, которые не имеют технического паспорт изделия на оборудование и паспорт программного продукта.

Следует обратить внимание, что паспорт проекта разработки (ТЗ на разработку) отличается от стандартного паспорта изделия или программы. Первый составляется исключительно на период работы над разработкой. Паспорт необходим для утверждения в государственных органах (если заказчиком выступает государственная структура), для патента, лицензирования, легального тиражирования. Сразу после завершения процедуры разработки программы составляется паспорт ПО.

Технический паспорт должен содержать следующие разделы:

1. Общие сведения о программном продукте.
 - 1.1. Наименование программного продукта.
 - 1.2. Назначение программного продукта.
2. Перечень модулей и слоев, входящих в состав программного продукта.
3. Функциональные возможности модулей и слоев.
4. Требования к аппаратно-программному обеспечению.

Задания для самостоятельного выполнения

1. Опираясь на приведенный ниже пример, разработайте технический паспорт на Ваш программный продукт.
2. Ответьте на контрольные вопросы.

Пример технического паспорта ИС «Yamed» для нужд медицинских организаций

1. Общие сведения о программном продукте

1.1. Информационная система «Yamed» (ИС «Yamed»).

1.2. ИС «Yamed» представляет собой программный продукт, предназначенный для персонифицированного учёта медицинской помощи, оказанной застрахованным лицам в сфере обязательного медицинского страхования. В его задачи входит:

- формирование, обработка, передача и хранение сведений о медицинской помощи, оказанной медицинской организацией застрахованным лицам;
- формирование и отправка сообщений в Региональный сегмент Единого регистра застрахованных лиц (РС ЕРЗ) с запросами на идентификацию застрахованных лиц;
- формирование и отправка реестров счетов за медицинскую помощь, оказанную лицам, застрахованным на территории РФ;
- формирование и отправка в Территориальный фонд обязательного медицинского страхования (ТФОМС) реестров счетов на медицинскую помощь, оказанную застрахованным лицам за пределами субъекта РФ, на территории которого выдан полис обязательного медицинского страхования (ОМС);
- формирование файла, содержащего сведения о случаях оказания медицинской помощи в формате *.OMS. Структура файла соответствует структуре, описанной в прил. Д приказа Федерального фонда обязательного медицинского страхования (ФОМС) от 23 марта 2018 г. № 54;
- формирование в ИС «Yamed» следующих отчетных форм:
 - а) счетов на оплату медицинской помощи за отчетный период;
 - б) актов выполненных работ;
 - в) реестров счетов;
 - г) основных показателей в разрезе видов и профилей оказанной медицинской помощи;
 - д) отчетных форм в соответствии с нормативной базой.

2. Перечень модулей и слоев, входящих в состав программного продукта

Модули программного продукта:

- Администрирование;
- Настройки;
- Авторизация;
- Стартовая страница;
- Реестр;
- Автоматический медико-экономический контроль;
- Отчеты;
- Поиск.

Слои приложения:

- Базовая библиотека (*Core*);
- Ядро (*Core Layer*);
- Доступ к данным (*Database*);
- Обработка данных (*Data*);
- Прикладной слой (*App Layer*);
- Оболочка (*Medicine Next*);
- *Workbench*.

Каждый из модулей может использоваться совместно с любым из слоев.

3. Функциональные возможности модулей и слоев

Модуль «Администрирование» предназначен для проведения общих настроек и сопровождения системы в целом. Он позволяет решать следующие задачи:

- регистрацию организации для обеспечения работы с данными компании;
- регистрацию пользователей системы;
- регистрацию версий слоев (словарей) для эффективного использования зарегистрированных данных
- обслуживание таблиц базы данных (сбор статистики, формирование индексов вторичных ключей) для оптимизации скорости работы системы.

Модуль «Настройки» предназначен для настройки соединения с базой данных (рис. 11.1).

Модуль «Авторизация» необходим для заполнения полей в отчетах (возможно добавление следующего функционала: записи в журнале операций; получение разрешенных определенной роли списка действий). Авторизация не защищает от неправомерного доступа к базе данных, так как используется двухуровневая модель приложения (приложение \Rightarrow база данных) и носит чисто функциональный характер.



Рис. 11.1

Процесс авторизации происходит путем заполнения соответствующих полей в одноименном окне (рис. 11.2).

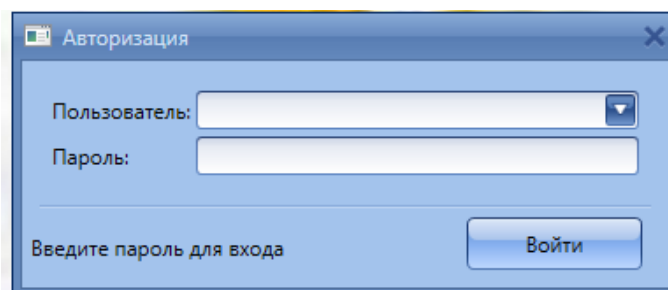


Рис. 11.2

Модуль «Стартовая страница» состоит из наиболее часто используемых при работе с программой элементов (рис. 11.3): реестр, регистратура, справочники НСИ, обновление.

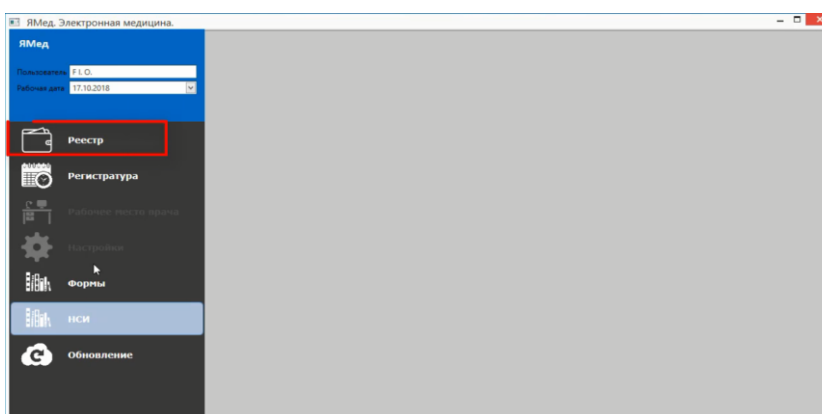


Рис. 11.3

Модуль «Реестр» (рис. 11.4) предоставляет возможность формирования реестра на оплату медицинской помощи с учетом отчетного месяца, типа реестра.

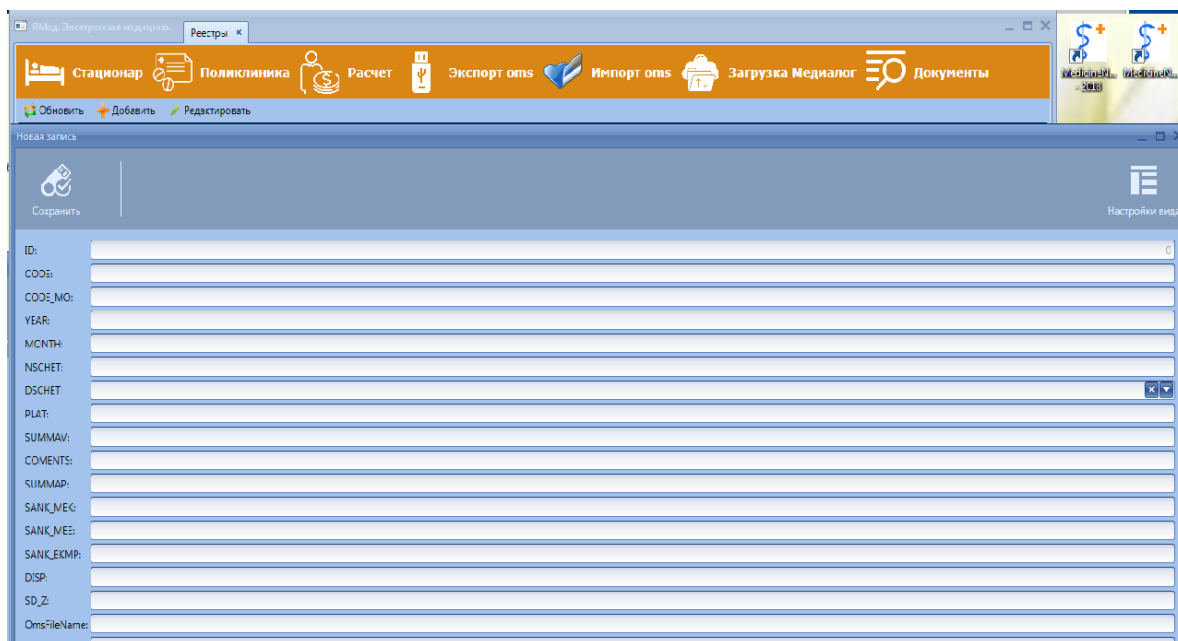


Рис. 11.4

Модуль «Стационар» (рис. 11.5) используется для формирования реестров оказанной медицинской помощи в условиях дневного и круглосуточного стационара, в соответствии Приложением Д приказа ФОМС от 23 марта 2018 г. № 54.

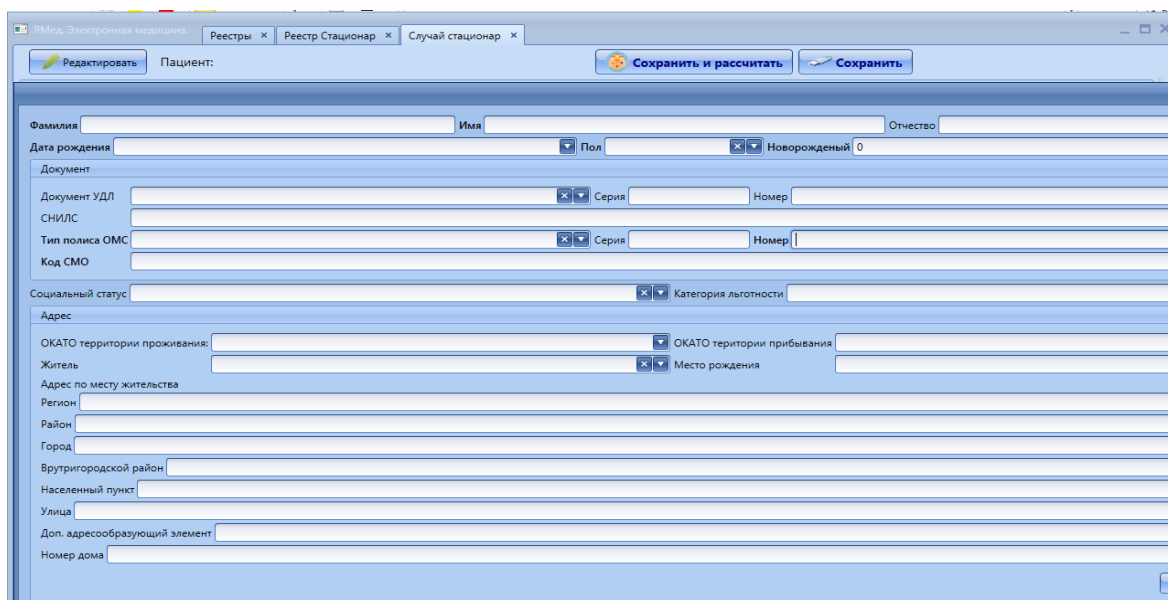


Рис. 11.5

Модуль «Поликлиника» (рис. 11.6) предназначен для формирования реестров оказанной медицинской помощи в условиях поликлиники в соответствии с прил. Д приказа ФОМС от 23 марта 2018 г. № 54.

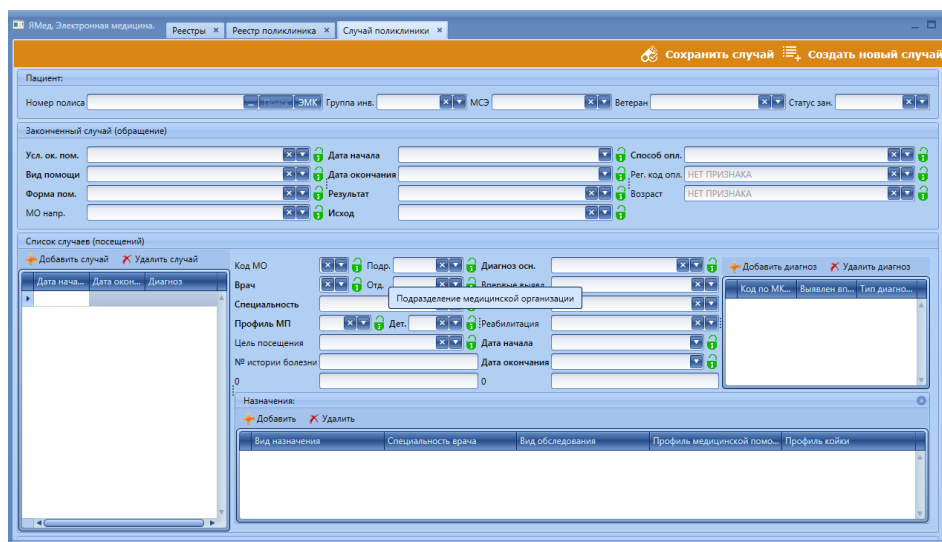


Рис. 11.6

Модуль «Расчет» (рис. 11.7) производит автоматический расчет стоимости лечения с учетом нормативно-правовой базы, порядков оплаты, действующих на территории оказания медицинской помощи.

Модуль «Экспорт OMS» (рис. 11.7) формирует файл на оплату в ТФОМС и Систему массового обслуживания (СМО) в соответствии с действующим форматом электронного взаимодействия.

Модуль «Импорт OMS» (рис. 11.7) используется для загрузки проведенных ТФОМС- и СМО-файлов, прошедших форматно-логический (ФЛК) и медико-экономический контроль (МЭК), медико-экономическую экспертизу (МЭЭ), а также экспертизу качества оказания медицинской помощи (ЭКМП).

Модуль «Документы» (рис. 11.7) формирует счета на оплату медицинской помощи за отчетный период, актов выполненных работ, реестров счетов, основных показателей деятельности медицинской организации и т. д.

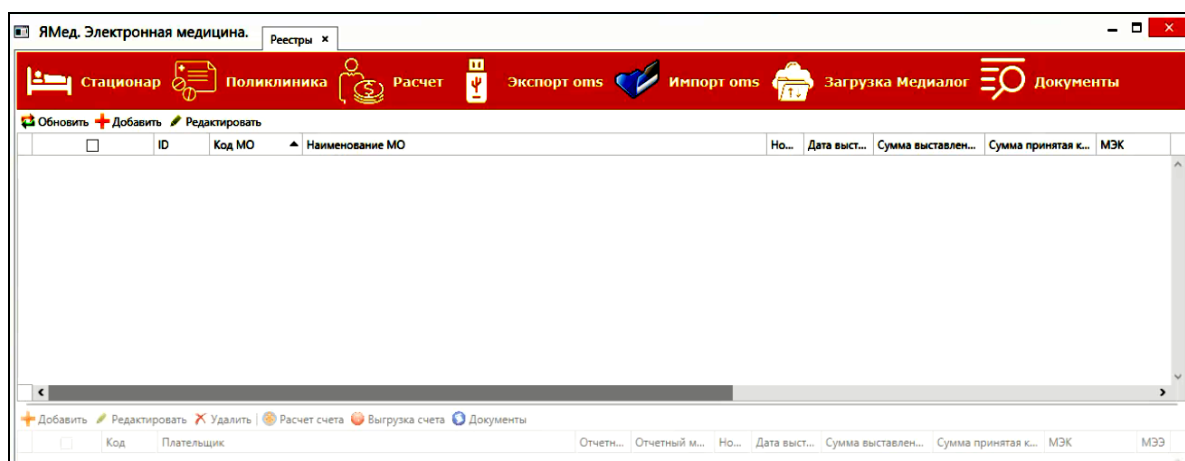


Рис. 11.7

Модуль-справочник «Регистратура» (рис. 11.8) служит для формирования записи на прием.

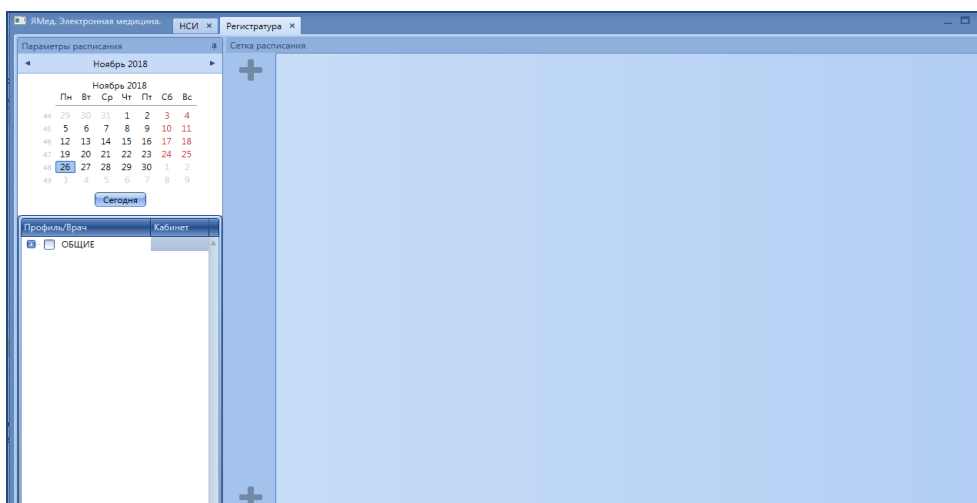


Рис. 11.8

Модуль «НСИ» (рис. 11.9) позволяет пользователю видеть справочники (федеральные, региональные) в удобном для прочтения виде, а также удалять, редактировать и добавлять данные. Предусмотрена возможность импорта и экспорта данных в формате *.XML.

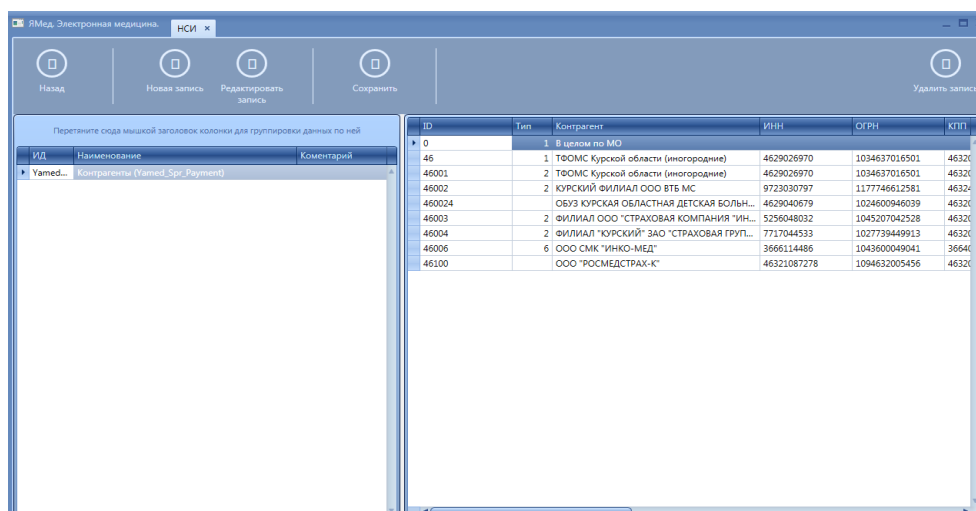


Рис. 11.9

Модуль «Автоматический медико-экономический контроль» предполагает проведение проверки на заполнение обязательных полей, дублирование случаев в одном реестре еще на этапе формирования и расчета реестров счетов, что уменьшает количество ошибок при проведении СМО- и ТФОМС-файлов, прошедших ФЛК и МЭК.

Примерный перечень отчетов, содержащихся в одноименном модуле, представлен на рис. 11.10.

ID	Наименование документа	Комментарий
11	ТАП	
25	Основные показатели ВД	
28	Акт ПВ	
30	Основные показатели ПВ	
35	Основные показатели ПД	
57	ОП диспансеризации детей-сирот	
58	ОП диспансеризации детей под опекой	
63	Отчет по профилю	
68	Коррекция ошибок ввода данных	
74	ОП по врачам	
75	ОП по специальности	
81	ОП диспансеризация 1 раз в 2 года	
84	Акт выполненных работ ВД 2-этап	д.1
85	Реестр диспансеризация детей под опекой	г.9
86	Реестр диспансеризация взрослых 2 этап	г.8
87	Реестр диспансеризация взрослых 1 этап	г.7
89	Акт ПВ	г.5
90	Акт диспансеризация 1 раз в 2 года	г.4
91	Акт диспансеризации детей-сирот	г.3
92	Акт диспансеризации детей под опекой	г.2
97	Акт ВМП	г.1
98	Основные показатели ВМП	9
99	ОП по специальности	8
100	ОП по врачам	7
101	Основные показатели (АПП, СП, КС, ДС)	6
102	Счет (АПП, СП, КС, ДС)	5
103	Акт выполненных работ (АПП, СП, КС, ДС)	4
104	Отчет по врачу	3
105	Акт выполненных работ ВД 1-этап	19
107	Счет ВМП	17
108	Реестр стационар-дн стац	16
109	Реестр стационар-дн стац	15

Рис. 11.10

Кроме того, предусмотрена возможность добавления новых, а также редактирование существующих отчетов по мере необходимости.

Модуль «Поиск» позволяет осуществлять поиск записей в реестрах счетов по заданным параметрам. Возможен также отдельный поиск по каждой из граф (рис. 11.11) и персональным данным (рис. 11.12).

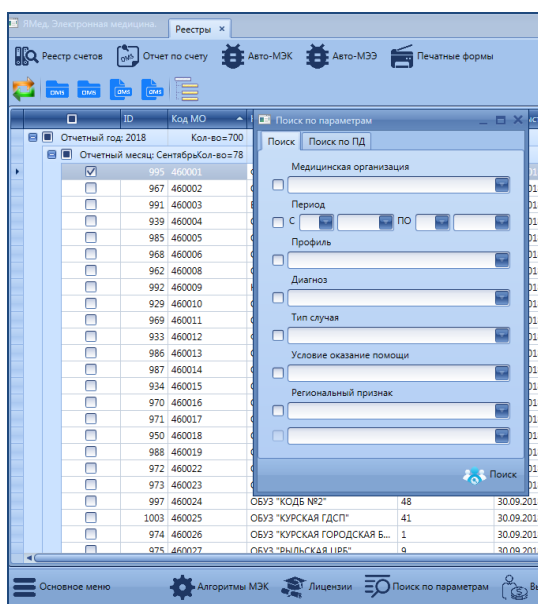


Рис. 11.11

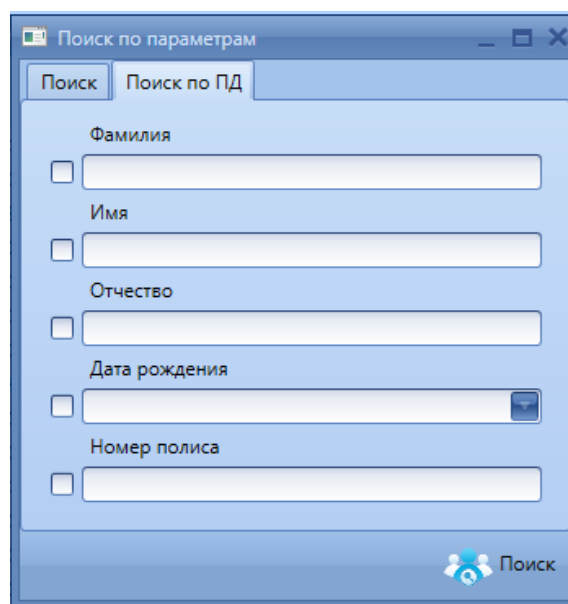


Рис. 11.12

4. Требования к аппаратно-программному обеспечению

Требования к ПО:

- минимальные: ОС Windows не ниже XP SP3, наличие .NET Framework 3.5 и выше. Для хранения баз данных используется СУБД MS SQL 2008 R2;
- рекомендуемые: ОС Windows 7 и выше, наличие .NET Framework 4 и выше.

Требования к аппаратному обеспечению:

- процессор двухъядерный, с тактовой частотой не менее 2,5 ГГц;
- не менее 4 ГБ ОЗУ (DDR3);
- 500 МБ свободного дискового пространства (без учета размера баз данных);
- наличие сетевого соединения (при работе с удаленным сервером БД и/или при работе с сервисом онлайн-проверки корректности ввода номера страхового полиса).

Требования к оформлению отчета

В качестве отчета по практической работе необходимо предоставить паспорт ПО.

Контрольные вопросы

1. Для каких целей разрабатывается паспорт программного обеспечения?
2. В чем разница между техническим заданием (паспортом на разработку) и непосредственно паспортом готового программного обеспечения?
3. Кто непосредственно принимает участие в разработке технического паспорта программного продукта или изделия?
4. Когда необходимо заниматься паспортизацией информационных систем?

СПИСОК РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ

1. ГОСТ 19.402-78. Описание программы : межгосударственный стандарт : введен Постановлением Государственного комитета СССР по стандартам от 18 декабря 1978 г. № 3350 : дата введения 1980-01-01. – Москва : Стандартиформ, 2010. – 2 с.

2. ГОСТ 19.101-77. Виды программ и программных документов : межгосударственный стандарт : введен Постановлением Государственного комитета стандартов Совета Министров СССР от 20 мая 1977 г. № 1268 : дата введения 1980-01-01. – Москва : Стандартиформ, 2010. – 3 с.

3. ГОСТ 19.701-90 (ИСО 5807-85). Схемы алгоритмов, программ, данных и систем. Обозначения условные и правила выполнения : межгосударственный стандарт : утвержден и введен в действие Постановлением Государственного комитета СССР по управлению качеством продукции и стандартам от 26 декабря 1990 г. № 3294 : дата введения 1992-01-01. – Москва : Стандартиформ, 2010. – 22 с.

4. ГОСТ Р ИСО/МЭК ТО 9294-93. Информационная технология. Руководство по управлению документированием программного обеспечения : утвержден и введен в действие Постановлением Госстандарта России от 20 декабря 1993 г. № 260 : дата введения 1994-07-01. – Москва : ИПК Издательство стандартов, 2003. – 8 с.

5. ГОСТ Р ИСО 9127-94. Системы обработки информации. Документация пользователя и информация на упаковке для потребительских программных пакетов : принят и введен в действие Постановлением Госстандарта России от 10 октября 1994 г. № 242 : дата введения 1995-07-01. – Москва : ИПК «Издательство стандартов», 2000. – 8 с.

6. ГОСТ 19.102-77. Стадии разработки программ и программной документации : межгосударственный стандарт : принят и введен в действие Постановлением Государственного комитета стандартов Совета Министров СССР от 20 мая 1977 г. № 1268 : дата введения 1980-01-01. – Москва : Стандартиформ, 2010. – 2 с.

7. ГОСТ 34.601-90. Информационная технология. Автоматизированные системы. Стадии создания : межгосударственный стандарт : утвержден и введен в действие Постановлением Государственного комитета СССР по управлению качеством продукции и стандартам от 29 декабря 1990 г. № 3469 : дата введения 1992-01-01. – Москва : Стандартиформ, 2009. – 5 с.

8. ГОСТ Р ИСО/МЭК 12207-2010. Информационная технология. Процессы жизненного цикла программных средств : национальный стандарт Российской Федерации : утвержден и введен в действие Приказом

Федерального агентства по техническому регулированию и метрологии от 30 ноября 2010 г. № 631-ст : дата введения 2012-03-01. – Москва : Стандартинформ, 2011. – 100 с.

9. ГОСТ 28195-89. Оценка качества программных средств. Общие положения : утвержден и введен в действие Постановлением Государственного комитета СССР по стандартам от 28 июля 1989 г. № 2507 : дата введения 1990-07-01. – Москва : ИПК «Издательство стандартов», 2000. – 30 с.

10. ГОСТ Р ИСО/МЭК 9126-93. Информационная технология. Оценка программной продукции. Характеристика качества и руководства по их применению : утвержден и введен в действие Постановлением Госстандарта России от 28 декабря 1993 г. № 267 : дата введения 1994-07-01. – Москва : ИПК «Издательство стандартов», 2000. – 10 с.

11. ГОСТ 19.106-78. Требования к программным документам, выполненным печатным способом : межгосударственный стандарт : введен Постановлением Государственного комитета СССР по стандартам от 18 декабря 1978 г. № 3350 : дата введения 1980-01-01. – Москва : Стандартинформ, 2010. – 10 с.

12. ГОСТ 19.105-78. Общие требования к программным документам : межгосударственный стандарт : введен Постановлением Государственного комитета СССР по стандартам от 18 декабря 1978 г. № 3350 : дата введения 1980-01-01. – Москва : Стандартинформ, 2010. – 2 с.

13. ГОСТ 19.301-79. Программа и методика испытаний. Требования к содержанию и оформлению : межгосударственный стандарт : введен Постановлением Государственного комитета СССР по стандартам от 11 декабря 1979 г. № 4753 : дата введения 1981-01-01. – Москва : Стандартинформ, 2010. – 2 с.

14. ГОСТ 19.201-78. Техническое задание. Требования к содержанию и оформлению : межгосударственный стандарт : введен Постановлением Государственного комитета СССР по стандартам от 18 декабря 1978 г. № 3351 : дата введения 1980-01-01. – Москва : Стандартинформ, 2010. – 2 с.

15. ГОСТ 34.201-2020. Виды, комплектность и обозначение документов при создании автоматизированных систем : межгосударственный стандарт : введен в действие Приказом Федерального агентства по техническому регулированию и метрологии от 19 ноября 2021 г. № 1521-ст : дата введения 2022-01-01. – Москва : Российский институт стандартизации, 2022. – 10 с.

16. ГОСТ 34.602-2020. Техническое задание на создание автоматизированной системы : межгосударственный стандарт : введен в действие Приказом Федерального агентства по техническому регулированию и метрологии от 19 ноября 2021 г. № 1522-ст : дата введения 2022-01-01. – Москва : Российский институт стандартизации, 2022. – 9 с.

17. РД 50-34.698-90. Методические указания. Информационная технология. Автоматизированные системы. Требования к содержанию документов : утверждены Постановлением Госстандарта СССР от 27 декабря 1990 г. № 3380. – URL: <http://a-podkidyshev.ru/GOST/RD-50-34-698-90-AS.pdf> (дата обращения: 11.04.2023). – Текст : электронный.

18. ГОСТ 19.503-79. Руководство системного программиста. Требования к содержанию и оформлению : межгосударственный стандарт : введен Постановлением Государственного комитета СССР по стандартам от 12 января 1979 г. № 74 : дата введения 1980-01-01. – Москва : Стандартинформ, 2010. – 2 с.

19. ГОСТ 19.504-79. Руководство программиста. Требования к содержанию и оформлению : межгосударственный стандарт : введен Постановлением Государственного комитета СССР по стандартам от 12 января 1979 г. № 74 : дата введения 1980-01-01. – Москва : Стандартинформ, 2010. – 1 с.

20. ГОСТ 19.505-79. Руководство оператора. Требования к содержанию и оформлению : межгосударственный стандарт : введен Постановлением Государственного комитета СССР по стандартам от 12 января 1979 г. № 74 : дата введения 1980-01-01. – Москва : Стандартинформ, 2010. – 2 с.

21. О техническом регулировании : Федеральный закон от 27 декабря 2002 г. № 184-ФЗ : принят Государственной Думой 15 декабря 2002 г. : одобрен Советом Федерации 18 декабря 2002 г. – URL: <https://www.rst.gov.ru/portal/gost/home/activity/technicalregulation> (дата обращения: 11.04.2023). – Текст : электронный.

22. Р 50.1.046. Рекомендации по выбору форм и схем обязательного подтверждения соответствия продукции при разработке технических регламентов : приняты и введены в действие Постановлением Госстандарта России от 23 июня 2003 г. № 201а-ст. – URL: <https://base.garant.ru/71722698/> (дата обращения: 11.04.2023). – Текст : электронный.

23. ГОСТ Р 57619-2017. Оценка соответствия. Рекомендации по содержанию и применению форм документов, используемых при добровольной сертификации услуг (работ) : национальный стандарт Российской Федерации : утвержден и введен в действие Приказом Федерального агентства по техническому регулированию и метрологии от 17 августа 2017 г. № 921-ст : дата введения 2018-01-01. – Москва : Стандартинформ, 2020. – 20 с.

24. Белов, В. В. Метрология, стандартизация, сертификация и контроль качества : учебное пособие / В. В. Белов, В. Б. Петропавловская. – Москва : КноРус, 2018. – 272 с.

25. Гвоздева, Т. В. Проектирование информационных систем. Стандартизация : учебное пособие / Т. В. Гвоздева, Б. А. Баллод. – 2-е изд., стер. – Санкт-Петербург : Лань, 2021. – 252 с.

26. Ехлаков, Ю. П. Управление программными проектами. Стандарты, модели : учебное пособие для вузов / Ю. П. Ехлаков. – 3-е изд., стер. – Санкт-Петербург : Лань, 2021. – 244 с.

27. Крылов, Е. В. Технология, надежность и качество программного обеспечения : учебник для вузов : в 2 книгах / Е. В. Крылов, В. А. Острейковский, Н. Г. Типикин. – Москва : Высшая школа, 2008. – 2 кн.

28. Ларин, М. В. Управление документами и на основе международного стандарта ИСО 15489-2001 : методическое пособие / М. В. Ларин, О. И. Рысков. – Москва : ОЦПК ВНИИДАД, 2005. – 107 с.

29. Остроух, А. В. Проектирование информационных систем : монография / А. В. Остроух, Н. Е. Суркова. – 2-е изд., стер. – Санкт-Петербург : Лань, 2021. – 164 с.

30. Тугов, В. В. Проектирование автоматизированных систем управления в TRACE MODE : учебное пособие / В. В. Тугов, А. И. Сергеев, Н. С. Шаров. – Оренбург : ОГУ, 2017. – 203 с.

ПРИЛОЖЕНИЯ

Приложение А

Темы проектов

1. Разработка информационной системы аэропорта.
2. Разработка информационной системы диспетчера такси.
3. Разработка информационной системы учета и анализа деятельности сети автомоек.
4. Разработка информационной системы больницы.
5. Разработка информационной системы детского сада.
6. Разработка информационной системы для учета посещаемости школы.
7. Разработка информационной системы для учета успеваемости школы.
8. Разработка информационной системы университета – учет заработной платы сотрудников.
9. Разработка информационной системы университета – учет посещаемости студентов.
10. Разработка информационной системы банка (различных направлений: ипотека, кредиты, вклады и др.).
11. Разработка информационной системы статистического центра по обработке звонков.
12. Разработка информационной системы налогового управления
13. Разработка мобильного приложения «...» для операционной системы «...».
14. Разработка электронного учебника по дисциплине «...».
15. Разработка обучающего приложения по дисциплине «...».
16. Разработка интерактивной системы для поддержки курса «...».
17. Разработка программного обеспечения для автоматизации работы отдела кадров.
18. Разработка корпоративного мессенджера организации.
19. Разработка программного обеспечения для сервиса по подбору специалистов.
20. Разработка программного обеспечения для автоматизации рабочего места отдела кадров предприятия.
21. Разработка единой базы знаний для внутреннего использования в организации.
22. Разработка корпоративной социальной сети организации.
23. Мобильное приложение для автоматизации какой-либо деятельности (бронирование заявок; составление отчетов, документов; доставки и др.).
24. Разработка веб-сервиса для размещения электронных объявлений.

Форма заявки на проведение сертификации услуг (работ)

В орган по сертификации

наименование органа

**ЗАЯВКА
НА ПРОВЕДЕНИЕ СЕРТИФИКАЦИИ УСЛУГ (РАБОТ)
В СИСТЕМЕ ДОБРОВОЛЬНОЙ СЕРТИФИКАЦИИ**

наименование Системы добровольной сертификации

№ _____ от _____

наименование организации-исполнителя, индивидуального предпринимателя (далее – Заявителя)

код ОКПО или ИНН

Юридический (фактический) адрес: _____

Телефон _____ Факс _____ E-mail _____

Банковские реквизиты: _____

в лице _____

фамилия, имя, отчество руководителя

просит провести добровольную сертификацию услуг (работ) _____

наименование группы (подгруппы, вида) услуга (работы), код ОК

оказываемой по _____

наименование и обозначение документов исполнителя

(национальные стандарты, стандарты организации, технические документы и др.)

на соответствие требованиям _____

наименование и обозначение документов

по схеме _____

номер схемы сертификации

Заявитель обязуется выполнять правила сертификации: соблюдать требования к сертификации и предоставлять любую информацию, необходимую для оценки.

Дополнительные сведения:

Руководитель организации _____

подпись

инициалы, фамилия

Главный бухгалтер _____

подпись

инициалы, фамилия

М.П.

Дата

Форма сертификата соответствия

наименование лица, создавшего Систему добровольной сертификации

Знак Системы добровольной сертификации	<hr/> наименование Системы добровольной сертификации <hr/>
--	--

СЕРТИФИКАТ СООТВЕТСТВИЯ

- (1) N _____
 - (2) Срок действия с _____ по _____
 - (3) Орган по сертификации _____
 - (4) Сертификат выдан _____
 - (5) Услуги (работы) _____
 - (6) Соответствуют требованиям _____
 - (7) Сертификат выдан на основании _____
 - (8) Дополнительная информация _____
 - (9) Руководитель ОС _____
- подпись
- инициалы, фамилия
- Эксперт _____
- подпись
- инициалы, фамилия

М.П.

Форма приложения к сертификату соответствия

наименование лица, создавшего Систему добровольной сертификации

Знак Системы добровольной сертификации	<hr/> наименование Системы добровольной сертификации <hr/>
--	--

ПРИЛОЖЕНИЕ

к сертификату соответствия № _____

Перечень видов услуг (работ), на которые распространяется действие сертификата соответствия

Код ОК	Наименование услуг (работ)	Обозначение нормативной и технической документации, регламентирующие характеристики, подтверждаемые при сертификации	Дополнительная информация
--------	----------------------------	--	---------------------------

Руководитель
органа по сертификации _____
подпись
инициалы, фамилия

Эксперт _____
подпись
инициалы, фамилия

М.П.

Зарегистрирован в Государственном реестре

Лицензионный договор (неисключительная лицензия) № _____

_____ Г.

Настоящий лицензионный договор (неисключительная лицензия) (далее – «**Договор**») заключен между: _____, паспорт: _____ № _____, выдан _____ Г., код подразделения _____ - _____, зарегистрированным по адресу: _____

_____ (далее – «**Лицензиар**») и _____, паспорт: _____ № _____, выдан _____ Г., код подразделения _____ - _____, зарегистрированным по адресу: _____ (далее – «**Лицензиат**»).

Лицензиар и **Лицензиат** совместно именуются **Стороны**, а по отдельности – **Сторона**.

1. Предмет договора

1.1. **Лицензиар** предоставляет **Лицензиату** неисключительные права на использование _____, в обусловленных настоящим **Договором** пределах и на определенный **Договором** срок (неисключительная лицензия), а **Лицензиат** за предоставление этих прав уплачивает вознаграждение **Лицензиару**.

1.2. **Лицензиар** гарантирует наличие у него предоставляемых по настоящему **Договору** исключительных прав на _____.
Описание _____:

2. Права и обязанности сторон

2.1. По настоящему **Договору** **Лицензиар** предоставляет **Лицензиату** следующие права использования _____:

2.2. Права на использование _____, указанные в п. 2.1. настоящего **Договора**, передаются **Лицензиаром** **Лицензиату** для использования на территории всего мира.

2.3. Права на использование _____, указанные в п. 2.1 настоящего **Договора**, передаются **Лицензиаром** **Лицензиату** для использования на весь срок действия исключительного права.

2.4. **Лицензиар** вправе предоставлять права использования _____, в том числе в части переданных **Лицензиату** прав посредством заключения лицензионных **Договоров** с третьими лицами.

2.5. **Лицензиат** не обязан представлять **Лицензиару** письменные отчеты об использовании _____.

3. Передача прав и порядок выплаты вознаграждения

3.1. За использование _____ по настоящему Договору Лицензиат выплачивает Лицензиару вознаграждение.

3.2. Обязательства Лицензиата по уплате вознаграждения считаются выполненными с момента получения денежных средств Лицензиаром.

3.3. Вознаграждение Лицензиару составляет _____ и выплачивается единовременно в течение _____ банковских дней с момента передачи Лицензиату _____. Оплата включает в себя стоимость всего срока использования Лицензиатом _____.

3.4. Лицензиат приобретает права, указанные в п. 2.1. настоящего Договора, в соответствии с условиями настоящего Договора с момента его подписания _____ передается Лицензиату посредством _____.

3.5. _____ и иные данные сообщаются дополнительно по электронной почте в течение _____ дней с момента подписания договора.

3.6. Все расчеты по настоящему Договору производятся в безналичной форме путем перечисления денежных средств с расчетного счета Лицензиата на расчетный счет Лицензиара. Датой совершения платежа считается дата зачисления денежных средств на расчетный счет Лицензиара.

4. Ответственность сторон

4.1. Лицензиар гарантирует следующее.

4.1.1. Лицензиар является законным правообладателем _____.

4.1.2. На момент вступления в силу настоящего Договора Лицензиару ничего не известно о правах третьих лиц, которые могут быть нарушены подписанием настоящего Договора.

4.1.3. На момент заключения настоящего Договора права, указанные в п. 2.1 настоящего Договора, не отчуждены и не заложены.

4.1.4. На момент заключения настоящего Договора права Лицензиара не оспорены в суде или иным законным способом.

4.2. При нарушении сторонами своих обязательств по Договору, стороны несут ответственность в порядке, предусмотренном действующим законодательством.

5. Обстоятельства непреодолимой силы

5.1. Стороны освобождаются от ответственности за неисполнение или ненадлежащее исполнение своих обязательств по настоящему Договору, если неисполнение или ненадлежащее исполнение обязательств произошли вследствие наступления обстоятельств непреодолимой силы (форс-мажорных обстоятельств), т. е. чрезвычайных и непредотвратимых при данных условиях обстоятельств, возникших помимо воли и желания Сторон, которых они не могли предвидеть и избежать. К таким обстоятельствам Стороны относят: пожары, наводнения, землетрясения и другие стихийные бедствия, технологические катастрофы, эпидемии, военные действия, а также непредвиденные и неотвратимые Сторонами события чрезвычайного характера, если эти обстоятельства не являлись следствием виновных действий Стороны, и непосредственно повлияли на исполнение настоящего Договора.

Сторона, затронутая обстоятельствами непреодолимой силы, обязана в кратчайшие сроки известить об этом другую **Сторону**, при необходимости предоставив соответствующие подтверждения наличия таких обстоятельств.

Неуведомление или несвоевременное уведомление (позднее 10 (десяти) рабочих дней после возникновения форс-мажорных обстоятельств) о наступлении форс-мажорных обстоятельств лишает соответствующую **Сторону** права ссылаться в дальнейшем на указанные выше обстоятельства как основание, освобождающее от ответственности за невыполнение или ненадлежащее исполнение обязательств.

Если обстоятельства непреодолимой силы будут продолжаться более 3 (трех) месяцев, то **Стороны** должны согласовать свои дальнейшие действия или могут расторгнуть настоящий **Договор**. Настоящий **Договор** может быть расторгнут любой из **Сторон** в одностороннем внесудебном порядке с обязательным письменным уведомлением другой **Стороны**. При этом настоящий **Договор** будет считаться расторгнутым с даты доставки уведомления о его расторжении настоящего.

6. Срок действия договора и порядок его прекращения

6.1. Настоящий **Договор** действует со дня его подписания **Сторонами** и до исполнения ими своих обязательств.

6.2. **Стороны** вправе досрочно расторгнуть настоящий **Договор** по письменному соглашению и в случаях, предусмотренных законодательством РФ.

6.3. При нарушении **Лицензиатом** п. 3.4 настоящего **Договора**, **Лицензиат** вправе в одностороннем порядке отказаться от настоящего **Договора** и потребовать уплаты неустойки в размере ____% от суммы вознаграждения за каждый день просрочки.

6.4. В случае нарушения срока выплаты вознаграждения, предусмотренного разд. 3 настоящего **Договора**, **Лицензиат** уплачивает **Лицензиару** пеню в размере ____% от суммы долга за каждый день просрочки. Пеня начисляется с момента направления **Лицензиатом** **Лицензиату** соответствующей претензии.

6.5. Использование **Лицензиатом** _____ способом, не предусмотренным **Договором**, либо по прекращении действия такого **Договора**, либо иным образом за пределами прав, предоставленных лицензиату по **Договору**, влечет ответственность за нарушение исключительного права на результат интеллектуальной деятельности или на средство индивидуализации, установленную законодательством.

6.6. Уведомление об отказе от **Договора** должно быть направлено **Стороне** в произвольной форме с подтверждением его получения (заказного или ценного письма, курьером и т. п.).

6.7. С момента получения **Стороной** такого уведомления об отказе настоящий **Договор** считается расторгнутым.

7. Порядок разрешения споров

7.1. В случае возникновения споров между **Сторонами** по вопросам, предусмотренным настоящим **Договором** или в связи с ним, **Стороны** принимают все меры к их разрешению путем переговоров.

7.2. При неурегулировании в процессе переговоров спорных вопросов споры разрешаются в судебном порядке в соответствии с действующим законодательством.

7.3. До передачи спора на разрешение суда **Стороны** принимают меры по досудебному урегулированию спора.

8. Заключительные положения

8.1. Любые изменения и дополнения к настоящему **Договору** действительны при условии, если они совершены в письменной форме и подписаны сторонами или надлежаще уполномоченными на то представителями **Сторон**. Все уведомления и сообщения должны направляться **Сторонами** друг другу в письменной форме.

8.2. Настоящий **Договор** составлен в двух экземплярах, имеющих одинаковую юридическую силу, один из которых находится у **Лицензиара**, второй – у **Лицензиата**.

8.3. В случае изменения наименования, местонахождения, банковских реквизитов и других данных каждая из **Сторон** обязана в пятидневный срок в письменной форме сообщить другой **Стороне** о произошедших изменениях.

8.4. Во всем ином, что не предусмотрено в настоящем **Договоре**, стороны руководствуются законодательством РФ.

9. Реквизиты сторон

Лицензиар

Лицензиат

Адрес регистрации:

Адрес регистрации:

Почтовый адрес:

Почтовый адрес:

Документ, удостоверяющий личность:

Документ, удостоверяющий личность:

Серия: _____ Номер: _____

Серия: _____ Номер: _____

Выдан: _____

Выдан: _____

Дата выдачи: _____

Дата выдачи: _____

Код подразделения: _____

Код подразделения: _____

подпись

расшифровка

подпись

расшифровка

СОДЕРЖАНИЕ

Введение	3
Практическое занятие № 1. Документирование программного кода	4
Практическое занятие № 2. Жизненный цикл программного обеспечения: модели и стадии	10
Практическое занятие № 3. Качество программных средств	16
Практическое занятие № 4. Технологии проектирования	23
Практическое занятие № 5. Создание технической документации. Техническое задание	31
Практическое занятие № 6. Создание технической документации. Пояснительная записка к эскизному проекту	38
Практическое занятие № 7. Разработка эксплуатационной программной документации	43
Практическое занятие № 8. Сертификация программного обеспечения ...	49
Практическое занятие № 9. Лицензирование программного обеспечения	54
Практическое занятие № 10. Тестирование и отладка программного обеспечения. Шаблоны сопроводительных документов	62
Практическое занятие № 11. Паспорт программного обеспечения	70
Список рекомендуемой литературы	79
Приложения	83

Учебное издание

**ДОКУМЕНТИРОВАНИЕ И СЕРТИФИКАЦИЯ
ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

Практикум

**Составители: Хабибрахманова Алсу Ильгамовна,
Бикеева Наталья Геннадьевна**

Кафедра информационных технологий и интеллектуальных систем КГЭУ

Редактор *М. С. Беркутова*
Технический редактор *И. В. Краснова*
Компьютерная верстка *Т. И. Лунченковой*

Подписано в печать 29.11.2023.
Формат 60×84 1/16. Усл. печ. л. 5,35. Уч.-изд. л. 3,90.
Заказ 485/эл.

Редакционно-издательский отдел КГЭУ,
420066, г. Казань, ул. Красносельская, 51