|  |  |
| --- | --- |
|  |  |
| МИНОБРНАУКИ РОССИИ | | |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** | | |

**Р.Г. БОЛБАКОВ, М.Ю. ВОЛКОВ, В.Т. МАТЧИН, В.А. МОРДВИНОВ**

**ЗАДАНИЯ И МЕТОДИЧЕСКИЕ УКАЗАНИЯ К ВЫПОЛНЕНИЮ ПРАКТИЧЕСКИХ РАБОТ ПО ДИСЦИПЛИНЕ «ИНФОРМАЦИОННЫЙ МЕНЕДЖМЕНТ ПРОГРАММНЫХ ПРОДУКТОВ И СИСТЕМ»**

**УЧЕБНО-МЕТОДИЧЕСКОЕ ПОСОБИЕ**

**для студентов бакалавриата РТУ МИРЭА**

**по образовательной программе «Разработка программных продуктов и проектирование информационных систем» направления подготовки 09.03.04 «Программная инженерия»**

Москва – 2021

УДК 004.92

**Задания и методические указания к выполнению практических работ по дисциплине «Информационный менеджмент программных продуктов и систем»** / Учебно-методическое пособие для студентов бакалавриата РТУ МИРЭА по образовательной программе «Разработка программных продуктов и проектирование информационных систем» направления подготовки 09.03.04 «Программная инженерия» [Электронный ресурс]: учебно-методическое пособие / Болбаков Р.Г., Волков М.Ю., Матчин В.Т., Мордвинов В.А. – Москва: МИРЭА - Российский технологический университет (РТУ МИРЭА), 2021. — 1 электрон. опт. диск (CDROM), 80 с. – Текст: электронный.

Рассматриваются вопросы и даются установки и рекомендации по осуществлению в рамках практических занятий на выпускающей кафедре Инструментального и прикладного программного обеспечения (ИиППО) Института ИТ РТУ МИРЭА по дисциплине «Информационный менеджмент программных продуктов и систем» для образовательной программы (профиля) бакалавриата «Разработка программных продуктов и систем» направления подготовки 09.03.04 «Программная инженерия», в основу которых закладывается освоение сущности и методологии разностороннего менеджмента в проектах по созданию и сопровождению программных продуктов и систем.

Материалы адресованы студентам бакалавриата, выполняющим практические работы по указанной дисциплине сообразно требованиям и составу Учебного плана и Рабочей программы указанной дисциплины (уточняются по годам). Эти работы поддерживаются соответствующим информационным обеспечением, в том числе в среде дистанционного обучения.

В помощь иностранным студентам представленные здесь установочные и методические материалы выпускаются также на английском языке.

Учебное пособие отображает реальный опыт работы и парадигматику развития учебно-научного процесса выпускающей кафедры Инструментального и прикладного программного обеспечения (ИиППО) Института ИТ РТУ МИРЭА.

Учебно-методическое пособие представлено и издаётся в авторской редакции.

Авторский коллектив: Болбаков Роман Геннадьевич (к.т.н., доц., зав. каф. ИиППО),

Волков Михаил Юрьевич, (преп., каф. ИиППО),

Матчин Василий Тимофеевич, (ст. преп., каф. ИиППО, редактор уч. мет. пособия),

Мордвинов Владимир Александрович (к.т.н., проф. каф. ИиППО, отв. за нормоконтроль ВКР)

Рецензент:

Холопов Владимир Анатольевич, кандидат технических наук, доцент, заведующий кафедрой промышленной информатики Института ИТ РТУ МИРЭА

Минимальные системные требования:

Поддерживаемые ОС: Windows 10 и выше.

Память: ОЗУ 8 Гб.

Жесткий диск: не менее 128 Гб.

Устройства ввода: клавиатура, мышь, желателен сенсорный экран.

Дополнительные программные средства: Программа Adobe Acrobat Reader

Информационно-технологическое сопровождение настоящего продукта – тьюторы-модераторы Беликов И.В, Санников Д.И. и другие.

Рекомендовано к использованию по решению Учёного совета …

\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_ 2021 г.

Объем: \_\_\_ Мб

ISBN \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

© Р.Г. Болбаков, М.Ю. Волков, В.Т. Матчин, В.А. Мордвинов, 2021

© МИРЭА – Российский технологический университет, 2021

**Оглавление**

[Введение 5](#_Toc61087047)

[Раздел 01. Теоретический базис и справочные материалы к практическим занятиям по дисциплине «Информационный менеджмент…» 12](#_Toc61087048)

[01.01. Методология, модели разработки. Перечни и понятия. Управленческое соглашение проекта 12](#_Toc61087049)

[01.02. Жизненный цикл программного обеспечения. Методология, перечни и понятия. Управленческое соглашение проекта 19](#_Toc61087050)

[01.03. Национальный стандарт Российской Федерации ГОСТ Р 53622-2009 «Информационные технологии. Информационно-вычислительные системы. Стадии и этапы жизненного цикла, виды и комплектность документов. Information technologies. Information-computing systems. Life cycle stages and steps, kinds and completeness of the documents. ОКС 03.120.10» в управленческом соглашении проекта ПО 23](#_Toc61087051)

[01.04. Интегрированные среды разработки в менеджменте создания ПО. Языковое и управленческое соглашения проекта 28](#_Toc61087052)

[01.05. Проектный менеджмент и применение в нём (в формате учебной деятельности) положений ГОСТ Р МЭК 61160-2015 «Проектный менеджмент. Документальный анализ проекта» 38](#_Toc61087053)

[01.06. Глоссарий (к формированию онтологического соглашения проекта) 50](#_Toc61087054)

[Раздел 02. Задания на выполнение практических работ и указания по их выполнению 57](#_Toc61087055)

[02.01. Задания: состав, структура, последовательность выполнения 57](#_Toc61087056)

[02.02. Структура отчёта по практической работе и его титульная часть 62](#_Toc61087057)

[02.03. Краткое изложение правил оформления и редакторской обработки текста отчёта практической работы 65](#_Toc61087058)

[02.04. Краткое описание выбранного в качестве примера проекта – прототипа из числа ВКР бакалавриата 72](#_Toc61087059)

[02.05. Технологическая среда и ПО администрирования и модерирования образовательными ресурсами в дистанционном обучении (ДО) и дистанционном информационно-методическом обеспечении работ и процедур, связанных с выполнением практических работ по дисциплине «Информационный менеджмент…» 75](#_Toc61087060)

[Заключение 77](#_Toc61087061)

[Источники 81](#_Toc61087062)

# Введение

Настоящие теоретические пояснения, справочные материалы, задания и методические указания (МУ) определяют состав, структуру и содержание выполняемых на четвёртом курсе бакалавриата так называемых «Практических работ» по изучаемой на выпускающей кафедре ИиППО дисциплине «Информационный менеджмент программных продуктов и систем» студентами бакалавриата образовательной программы «Разработка программных продуктов и проектирование информационных систем» в РТУ МИРЭА по направлению подготовки 09.03.04. «Программная инженерия».

Настоящая дисциплина (Б1.В.ДВ.04.1) вводится в действие для студентов бакалавриата, начиная с 2019-го года поступления в вуз, то есть она сама и её Рабочая программа отвечают требованиям Учебного плана 09.03.04\_РПППИС\_ИИТ\_2019.plx, одобренного Учёным советом вуза 24.04.2019 (Протокол № 9). В составе дисциплины присутствуют обязательные для студентов практические работы (практические занятия, относимые к разряду семинарских), вводные разъясняющие учебные материалы к которым в сопровождении мини презентационного пакета и глоссария, а также вариативный перечень заданий и методические указания по выполнению этих заданий представлены в настоящей публикации.

Согласно утверждённому Учебному плану в процессе изучения указанной дисциплины (Б1.В.ДВ.04.1) должны быть освоены следующие компетенции в составе ФГОС ВО 09.03.04 (образовательный стандарт – ФГОС ВО 09.03.04 за № 920 от 19.09.2017): ПК-2.2; ПК-2.6; ПК-2.8; ПК-2.10; ПК-2.11; ПК-2.16; ПК-1.17.

Описание приведённых здесь компетенций с расшифровкой индикаторов освоения и действий с ними связанных излагаются в конспекте лекций по дисциплине, которые предоставляются студентам в пользование.

Из выше приведённого следует, что обсуждаемая здесь дисциплина включается в реальный учебный процесс на 7-ом семестре, завершающем теоретический курс обучения в бакалавриате, начиная с 2022/23 уч. г.

До этого, то есть в 2021/22 учебном году (и ранее) также перед самым дипломным проектированием на 4-ом курсе бакалавриата осваивалась созвучная дисциплина «Информационный менеджмент систем», что, по-видимому, означает её более широкое тематическое толкование, куда как меньшую конкретизацию в части менеджмента разработки, создания и сопровождения программных продуктов. Но, тем не менее, некоторая наследственность в переходе от прежней дисциплины к настоящей налицо, тем более, что и в том и в другом случаях лекционный материал строился как 8-ми модульный вариативный, что позволило внести соответствующие обновления заменой части прежних модулей (№ 06 и 07) на новые, не ломая установившихся структуры курса и средств его информационно-методического и технологического обеспечения.

Поэтому в переходной период накануне введения в строй обновлённой дисциплины в части практических работ доживающей свой век прежней дисциплины можно с успехом опираться на материалы настоящего пособия.

Однако, при этом надо иметь ввиду, что прежняя версия дисциплины закладывалась в Учебные планы в соответствии с требованиями ФГОС ВО 09.03.04 предыдущей редакции от 12.03.2015 за № 229.

Отвечающий этой версии Учебный план 09.03.04\_РПППИС\_ИИТ\_2018. Plx, одобренный Учёным советом вуза 28.03.2018 (Протокол № 8), и ещё более ранние версии определяют к освоению несколько иной перечень компетенций, из числа прописанных в ФГОС ВО 09.03.04 (2015-го года), а именно: ПК-14.

Эта профессиональная компетенция по состоянию на начало 2021 года в названном учебном плане представлена следующим образом: «обладать готовностью обосновывать принимаемые проектные решения, осуществлять постановку и выполнение эксперимента по проверке их корректности и эффективности». В дисциплине «информационный менеджмент…» такого рода компетентность является краеугольным камнем в становлении специалиста по программной инженерии, а потому она и все указанные выше компетенции действительно важны и «обыгрываются» как в теоретической лекционной части курса, так и в описываемых здесь практических занятиях (практических работах) по дисциплине. Эти обстоятельства должны быть достаточно выпукло и однозначно показаны в отчётах студентов по практическим занятиям как в параметрах индикаций освоения, так и в связанных с этим действиях, для чего, собственно, и проводятся практические занятия. При всём указанном следует обратить особое внимание на то, что год от года в процессе совершенствования Учебных планов, Рабочих программ по дисциплинам могут произойти некоторые изменения как в составе компетенций, так и в раскрытии конкретики их требований. Так что, прежде чем приступать к реализации практических работ, и, тем более, к составлению отчёта по ним, совершенно необходимо изначально вместе с преподавателем уточнить, какие компетенции и их описания актуализированы на текущий учебный год. Это принципиально важно!

Не менее существенно и то, что дисциплина «информационный менеджмент…» конфигурирована на самой оконцовке теоретической подготовки специалиста перед выходом на дипломное проектирование, к тому же по жанру дисциплина широка и универсальна. Дисциплина обобщает разделы некоторых ранее или параллельно изучаемых дисциплин, например, «Проектирование информационных систем», «Программные средства зашиты информации», «Системная и программная инженерия» и т.п.

В связи с указанным как лекционные занятия, так и практические работы подчинены цели особенно упрочить знания и навыки студентов, положительно влияющие в ближайшем последствии на качественное выполнение выпускной квалификационной работы (ВКР), разумеется, прежде всего в тех частях её, которые прямо или косвенно связаны с менеджментом проекта, с его устойчивостью, с обеспечением жизненного цикла (ЖЦ) создаваемых программных продуктов и систем (здесь и далее: «ППиС»). Для этого практические занятия по дисциплине подчиняются логике реинжиниринга, как созидательного жанра вполне конкретной деятельности студента, выполняющего практические работы.

Конкретно это воплощается в том, что обучающемуся из специфического выделенного информационного консиста базы знаний репозитория кафедры предлагается к анализу и улучшению некий обезличенный студенческий проект (каждому студенту индивидуальный) создания и/или сопровождения программных продуктов и систем. Одновременно предоставляется рекомендуемый кафедрой к изучению и апробациям «на ниве» указанного проекта некий чётко сформулированный инвариантный перечень моделей, методов, средств менеджмента.

Step by step в процессе практических занятий студент выполняет следующие действия:

01. Анализирует состав и содержание исследуемого проекта с позиций рациональности и качества применения его анонимным автором избранных в проекте метода, методики и результатов менеджмента, обращая особое внимание на следующие свойства:

* ясность и однозначность средств управления (управленческое соглашение проекта);
* качество, эргономический и технико-эстетический уровни проектирования интерфейсов, юзабилити (онтологическое соглашение проекта);
* обеспечение ЖЦ (трёхзвенное соглашение проекта);
* обеспечение устойчивости проекта (трёхзвенное соглашение проекта);
* удачность выбора языковой среды (языковое соглашение проекта);
* чёткость формирования онтологии проекта, атрибутов межагентного взаимодействия (онтологическое соглашение проекта);
* эффективность и полнота использования выбранного автором менеджмента (управленческое соглашение проекта).

**Мотивированные оценка и предложения** оформляются в форме: «Экспертное заключение по состоянию и качеству менеджмента проекта – прототипа».

(Форма и структура представления достаточно вольные, но с обязательным полным охватом приведённого выше перечня. Разумеется, этот учебный документ, как и все последующие, имеет титульный лист, подтитульный лист с выходными данными, использованные источники, подписывается с раскрытием Ф.И.О. и указанием даты).

02. Один за другим перебираются порученные для рассмотрения варианты осовремененного менеджмента, что нацеливается на глубокий или частичный реинжиниринг проекта-прототипа. Оцениваются возможности и прогнозируемые результаты такого рода изменений, что оформляется по каждой из версий авторизированной запиской обучающегося в двух её разделах: «Технические предложения (ТП) по реинжинирингу проекта «имя рек» в части менеджмента» и «Экспертное заключение по состоянию и качеству менеджмента в реинжиниринге проекта – выбираемого прототипа проекта».

Рассмотрению предлагаются: Эволюционные и Инкрементальные модели менеджмента [1]; методология разработки Agile (в том числе v. 4.0) [2]; методология разработки Scrum [3]; методология разработки FDD (Feature Driven Development) [4]; методология разработки RAD (Rapid Application Development) [5]; методология разработки Spiral [6]; методология разработки LD (Lean Development) [7] и другие.

В сопоставлении могут участвовать также более старые версии традиционные для прежних времён: методы спирали (старых версий); метод водопада; метод наибыстрейшего спуска; каскадный метод; веерный метод; а также их обобщения в виде классического метода конвергенции/дивергенции; методы планируемого эксперимента (например, с использованием математического аппарата Монте-Карло); матрица Александера и ряд других известных решений.

03. На заключительном этапе практических занятий по дисциплине обучающийся формирует наукоёмкую авторизированную аналитическую записку, в которой мотивированно выбирает и описывает лучшее на его взгляд решение применительно к рассматриваемому проектному прототипу, сопровождая свой выбор кратким каталожным описанием (КО) окончательного решения (реинжиниринга проекта-прототипа).

Центральное место в записке занимает формулировка обучающегося по формированию так называемого **трёхзвенного проектного соглашения** (востребованного в последствии в работе над ВКР бакалавра), а именно:

* языковое / платформенное (кроссплатформенное) соглашение проекта;
* онтологическое соглашение проекта;
* управленческое соглашение проекта.

**Напоминание:** при всём этом не следует забывать корректно указывать источники и приводить посылы о соответствующем освоении требований компетенций. Весь пакет предоставляется преподавателю, возможно, с необходимостью отвечать на его вопросы и замечания, что должно заканчиваться благополучным кафедральным зачётом по выполненным практическим работам.

Всё это предоставляется исключительно в электронном виде в установленном для традиционных форм обучения или в формате дистанционного обучения (ДО) распорядках.

Задания и исходные материалы также выдаются студенту либо в процессе очного общения на кафедре ИиППО средствами открытия QR-кодовых окон доступа к упомянутому выше консисту, либо в условиях ДО именной рассылкой или открытием доступа к специально созданному для этого облаку в Интернете.

Все приведённые здесь материалы обрели наряду с русскоязычной версией англоязычную, что сделано в интересах иностранных студентов, обучающихся в Университете. Прилагается также двуязычный глоссарий.

(Прим.: англоязычная версия настоящих материалов издаётся отдельно и является автономным учебно-методическим документом).

В завершение введения рекомендуется знакомиться с приводимым ниже пояснительным тематическим материалом опираясь прежде всего на конспект лекций, рекомендованное в лекциях основное учебное пособие, а также внимательно изучить положения Национального стандарта РФ ГОСТ Р 53622-2009 «Информационные технологии. Информационно-вычислительные системы. Стадии и этапы жизненного цикла, виды и комплектность документов» // Information technologies. Information-computing systems. Life cycle stages and steps, kinds and completeness of the documents. ОКС 03.120.10. Дата введения 2011-01-01. Активно используйте материалы этого стандарта в Вашей работе.

**Основные источники в качестве учебников / учебных пособий** согласно утверждённой Рабочей программы (РП) по дисциплине и конспекту лекций к ней: Петрова, Е. А. Информационный менеджмент: учебник / Е. А. Петрова, Е. А. Фокина. — Санкт-Петербург: Лань, 2019. — 144 с. — ISBN 978-5-8114-3923-2. — Текст: электронный // Лань: электронно-библиотечная система. — URL: https://e.lanbook.com/book/125740 (дата обращения: 29.11.2020). — Режим доступа: свободный (последнее обращение 03.09.20), а также конспект лекций кафедры ИиППО по дисциплине, сопровождаемый презентационным пакетом в PowerPoint.

С пожеланием успеха! Составители и переводчики на английский язык настоящего учебно-методического материала.

# Раздел 01. Теоретический базис и справочные материалы к практическим занятиям по дисциплине «Информационный менеджмент…»

## 01.01. Методология, модели разработки. Перечни и понятия. Управленческое соглашение проекта

Методология разработки программного обеспечения — это прежде всего организация труда, включающая идеологические принципы, план, контроль над процессами, подход к сотрудникам.

Здесь представлены 12 современных моделей разработки:

**Waterfall** — модель быстрой разработки.

**RUP (Rational Unified Process)** — «рациональная модель».

**Agile** — общая методология гибкой разработки.

**Crystal Clear** — подход с уравниванием разработчиков в коллективе.

**Spiral** — спиральный метод (в современных редакциях, начиная с 2017 г.).

**RAD (Rapid Application Development)** — модель быстрой разработки.

**DSDM (Dynamic Systems Development Model)** — динамическая модель.

**FDD (Feature Driven Development)** — методология, рассматривающая будущие изменения.

**JAD (Joint Application Development)** — ориентированный на пользователя подход.

**Scrum** — концепция работы в условиях сорванных сроков и идеологического кризиса.

**XP (Extreme Programming)** — экстремальная разработка в динамической среде.

**LD (Lean Development)** — метод, предполагающий бережное отношение ко всем участникам процесса.

Раскроем приведённый выше перечень на реферативном уровне описания его составляющих.

**Waterfall**

Модель Waterfall относится к классическому пониманию разработки программного обеспечения (ПО). Весь процесс является жестким и линейным, имеет четкие цели для каждого этапа. Новая фаза начинается по завершению предыдущей; нет возврата назад. Преимущества водопадной методологии — децентрализация и строгий контроль над сроками и качеством исполнения.

На практике Waterfall часто не оправдывает ожиданий, поскольку игнорирует динамические изменения. Так, после тестирования очень сложно откатить процесс и заложить функции, не учтенные на стадии разработки. Waterfall не всегда эффективен ещё и потому, что предполагает временные простои сотрудников в рамках одного проекта. Тестирование проводится только в конце разработки, хотя проблемы, найденные на завершающем этапе — это дорогостоящие и времяёмкие исправления и корректировки.

**RUP**

RUP — итеративный подход, который успешно решает проблемы, которые есть у Waterfall.

Чем замечателен RUP:

* Учитывает изменяющиеся требования. Как бы ни был хорош руководитель проекта, учесть всё в начале невозможно.
* Интеграция функций происходит постепенно, то есть каждая «деталь» проходит цикл разработки, проверки и внедрения в проект. Как следствие, снижаются риски и стоимость производства.
* Ранний выпуск продукта. ПО выходит с уменьшенной функциональностью, чтобы занять нишу на рынке и противостоять конкурентам, после чего обрастает добавлениями.
* Повторное использование. При наращивании функциональности проще выделить типовые решения, которые сократят разработку.
* Постоянное обучение. Из-за частых итераций разработчики не имеют больших пауз между доработкой кода, поэтому профессиональный рост происходит плавно и безболезненно.
* Постоянное улучшение продукта. Итерации позволяют оценить проект не только с точки зрения соответствия плану и ТЗ, но и найти пути увеличения эффективности и качества продукта.

**Agile**

Agile — метод гибкой разработки программного обеспечения, предполагающий большое количество итераций. Документ Agile Manifesto описывает 4 идеи и 12 принципов гибкого подхода. Кратко его можно описать всего двумя пунктами:

* Неформальные отношения важнее задокументированных. То есть устные договоренности между сотрудниками, между заказчиком и исполнителем важнее всего, что отражено в планах, договорах и техническом задании. Иначе говоря, клиент всегда прав.
* Работающий продукт — главная оценка прогресса. Важны не инструменты, решения, производительность и изящество, а тот факт, что все запланированные возможности реализованы.

Несмотря на недостатки, Agile стала фундаментальной концепцией для разработки ПО и нашла отражение в других методологиях, речь о которых пойдет далее. К тому же Agile постоянно развивается и совершенствуется. Например, в последнее время освоена и широко применяется улучшенная версия v.04.

**Crystal Clear**

Методология, созданная для небольших коллективов из 6−10 сотрудников. Также поддерживает принципы гибкой разработки, но имеет чуть больше конкретики. Основная идея, которая и заключена в названии — каждая команда является набором людей с разным уровнем знаний, разными умениями и опытом.

Именно поэтому нет универсального подхода для разработки ПО. Он должен определяться в процессе общения внутри группы. Там же назначаются роли, инструменты, стандарты. Затем группа принимается за единицу и те же самые вопросы решаются на уровень выше, пока иерархия не дойдет до заказчика.

**Spiral**

Модель спирального жизненного цикла — это сложная организация жизненного цикла ПО, которая фокусируется на раннем выявлении и уменьшении проектных рисков. Разработка начинается в небольшом масштабе, решаются локальные задачи, оцениваются риски и пути их уменьшения. Следующий шаг охватывает более комплексные задачи — следующий виток спирали.

Преимущество подхода не в увеличении скорости разработки, а в снижении уровня возникновения рисков. Успешность спирального метода зависит от добросовестного, внимательного и компетентного управления, а размер проекта не имеет принципиального значения.

**RAD**

RAD — методология, которая во главу угла ставит скорость и удобство разработки. Одно из главных условий — использование языка быстрой разработки. Это название абстрактного языка программирования, с помощью которого программист способен решать задачи быстрее, чем с представителями третьего поколения (C / C ++, и т.д.). Вот ещё несколько пунктов концепции:

* Использование фокус-групп для сбора требований.
* Прототипирование и пользовательское тестирование конструкций.
* Повторное использование программных компонентов.
* Использование плана, не включающего переработку, или дизайн следующей версии продукта.
* Проведение неформальных совещаний по запросу одной из сторон.
* RAD предполагает использование целого комплекса инструментов помимо языка быстрой разработки: системы сбора требований, среды разработки, фреймворки, программы для группового общения, ПО для тестирования.

**DSDM**

Модель развития динамических систем была разработана в Великобритании в середине 1990-х годов и является эволюционным развитием быстрой разработки приложений (RAD). Основная идея стандартная: при планировании в самом начале невозможно понимать всех тонкостей разработки, поэтому весь процесс — исследовательская работа.

В DSDM тоже присутствует деление на команды, в каждой из которых есть уполномоченный для принятия стратегических решений. В процессе могут участвовать все заинтересованные стороны: пользователи, разработчики, заказчики, руководители. Тестирование проводится на протяжении всего жизненного цикла.

**FDD**

FDD — процесс для обеспечения масштабируемости и повторяемости, при этом поощряющий творчество и инновации. Основные принципы FDD:

* Разработка каждого крупного проекта должна иметь системность.
* Процессы должны быть простыми и проработанными.
* Ценность и логичность процесса должна быть ясна каждому члену команды.
* Предпочтение отдаётся коротким итеративным циклам разработки. Это уменьшает количество ошибок и позволяет быстрее наращивать функциональность.
* FDD регламентирует время, которое должно затрачиваться на каждый из процессов. Организационной деятельности в цикле должна занимать не более 23−25%, в то время как на непосредственную разработку, сборку и тестирование функций необходимо тратить 75−77% времени.

**JAD**

JAD — это методология, нацеленная на максимальную занятость в разработке конечного пользователя. Происходит это посредством встреч и проведения совместных семинаров. JAD была придумана в 1970-х годах сотрудниками IBM и нацелена на бизнес в целом. Однако со временем данная концепция стала успешно применяться и для разработки программного обеспечения.

В отличие от подхода Waterfall, JAD приводит к сокращению времени разработки, большей удовлетворенности клиентов и экономии средств на изучении рынка. С другой стороны, это требует большой клиентской выборки и необходимости разработчиков работать не со строгими требованиями ТЗ, а с постоянно меняющимся мнением.

**Scrum**

Scrum — гибкий метод управления проектами, целью которого является повышение производительности труда в командах, ранее парализованных более тяжелыми методологическими процессами. В основе концепции лежат «спринты». Спринт — короткая итерация, строго ограниченная по времени (обычно 2−4 недели). В это время минимизируется длительность совещаний, но увеличивается их частота (они называются «схватками»).

Благодаря этому контроль за выполнением становится более гибким, а разработчики быстрее реагируют на возникающие проблемы. Традиционное планирование отходит на второй план, его место занимает журнал спринтов.

**XP**

Экстремальное программирование — возможность вести разработку в условиях постоянно меняющихся требований. Вот несколько признаков:

Игра в планирование. В начале проекта есть только приблизительный план, после каждой итерации его чёткость возрастает.

Высокая частота релизов. Новая версия продукта имеет незначительные изменения по сравнению с предыдущей, но время на выпуск при этом минимально.

Контакт с клиентом. Для удовлетворения требований конечной аудитории необходимо оперативное реагирование на замечания и пожелания.

Рефакторинг. Улучшение качества кода без уменьшения функциональности.

Стандарт выполнения кода. Или применяются общие правила, или разногласия в оформлении не подлежат обсуждению и критике.

Коллективная ответственность. Несмотря на то, что каждый член команды выполняет свой участок работ, за код в целом отвечает весь коллектив.

**LD**

Бережливая разработка ПО — ещё одно ответвление гибкой методологии, предполагающее сохранение высокого морально-функционального состояния разработчиков. Это выражается в:

Поощрении сотрудников за успешную работу.

Изменении текущих задач только по мере необходимости или по запросу заказчика.

Строгое выполнение плана: всё, что сверх — считается потерями времени и ресурсов.

Внедрении общей концепции «Мыслить широко, делать мало, ошибаться быстро, учиться стремительно».

## 01.02. Жизненный цикл программного обеспечения. Методология, перечни и понятия. Управленческое соглашение проекта

В основе любой отрасли промышленного производства, к которой, несомненно, относится и создание программных продуктов, лежит технологический процесс. Большинство характеристик ПО, как, например, стоимость, актуальность, функциональность, качество, определяются непосредственно технологией разработки и точностью ее соблюдения.

Ключевое при этом понятие качество положено в основу международных стандартов ISO серии 9000. Основополагающая идея ISO 9000 состоит в том, что система качества предполагает построение такой структуры управления производством, которая гарантирует выпуск качественного продукта (в нашем случае программного обеспечения) в любой момент, пока система действует. Требования стандартов ISO 9000 распространяются на такие качества продукта, как управление документацией и проектированием, контроль и проведение испытаний, корректирующие и упреждающие действия, техническая поддержка и многие другие.

Теперь непосредственно о понятии жизненного цикла ПО. Под жизненным циклом (ЖЦ) ПО понимается совокупность процессов, связанных с последовательным изменением состояния ПО от формирования исходных требований к нему до окончания его эксплуатации. Жизненный цикл состоит из стадий логически завершенных частей ЖЦ. Стадии характеризуются определенным состоянием ПО, видом предусмотренных работ и их результатом. Выделяют несколько основных моделей ЖЦ, различающихся порядком следования стадий и взаимосвязями между ними, которые мы обсудим несколько позже.

При создании заказного ПО выделяют несколько основных стадий ЖЦ анализ и формализация требований заказчика, проектирование, реализация, тестирование, внедрение и эксплуатация.

Среди моделей ЖЦ можно выделить две основных группы: последовательные модели и эволюционные модели.

**Последовательные модели** предполагают, что каждая следующая стадия может быть начата только после завершения работ на предыдущей стадии. Он применим в тех случаях, когда требования к программному продукту четко определены и не меняются со временем, имеется достаточно большой опыт реализации подобного рода программного обеспечения и время реализации проекта составляет от нескольких месяцев до года. Реализация проекта по данному типу ЖЦ легко планируется и контролируется. Однако для этого необходимо заранее иметь все требования к ПО. Данные модели ЖЦ не приспособлены к изменениям требований к ПО. Разрабатываемый продукт не может использоваться, пока проект не будет близок к завершению. В реальности в последнее время эти модели жизненного цикла практически никогда не применяются.

**Эволюционные модели** формируются в предположении, что наращивание возможностей программного обеспечения происходит постепенно. В процессе разработки основные стадии ЖЦ (проектирование, реализация, тестирование) проходятся несколько раз применительно к очередной добавляемой порции возможностей ПО. Данные модели ЖЦ не требуют заранее наличия всех требований к ПО и позволяют заказчику активно участвовать в процессе разработки, что является безусловным плюсом. Однако среди недостатков эволюционных моделей можно отметить сложность в управлении и контроле выполнения проекта, сложность оценки затрат на разработку и риск бесконечного улучшения ПО. Данные модели применимы, когда требования к ПО плохо определены или будут изменяться, отсутствует достаточный опыт в разработке подобного ПО, используются новые технологии и (или) инструментальные средства, в ходе разработки ПО необходимо иметь промежуточные версии продукта.

Выбор конкретной модели ЖЦ в каждом конкретном случае зависит от ряда субъективных и объективных причин, сопутствующих проекту, то есть наличия четких и подробных требований к ПО; ресурсов, имеющихся в наличии для проведения работ по проекту; наличия и доступности заказчика в процессе разработки и т.д. Как правило, в реальности применяются смешанные методы жизненного цикла.

Прежде чем перейти к более подробному рассмотрению этих моделей, необходимо дать понятие **системной архитектуры**. Под архитектурой программной системы понимается совокупность решений относительно организации программной системы, выбора структурных элементов, составляющих систему и их интерфейсов, составления из этих структурных и поведенческих элементов все более и более крупных подсистем и т. д. Архитектура программной системы охватывает не только ее структурные и поведенческие аспекты, но и использование, функциональность, производительность, гибкость, возможности повторного применения, полноту, экономические и технологические ограничения и компромиссы, а также эстетические вопросы.

На сегодняшний день считается, что архитектура программной системы наиболее оптимально может быть описана с помощью пяти взаимосвязанных видов или представлений, каждый из которых является одной из возможных проекций организации и структуры системы и заостряет внимание на определенном аспекте ее функционирования:

* Вид с точки зрения **прецедентов (use case view)** охватывает прецеденты, которые описывают поведение системы, наблюдаемое конечными пользователями, аналитиками и тестерами. Этот вид специфицирует не истинную организацию программной системы, а те движущие силы, от которых зависит формирование системной архитектуры.
* Вид с точки зрения **проектирования (design view)** охватывает классы, интерфейсы и кооперации, формирующие словарь задачи и ее решения. Этот вид подчеркивает, прежде всего, функциональные требования, предъявляемые к системе, то есть те сервисы, которые она должна предоставлять конечным пользователям.
* Вид с точки зрения **процессов (process view)** охватывает нити и процессы, формирующие механизмы параллелизма и синхронизации в системе. Этот вид описывает главным образом производительность, масштабируемость и пропускную способность системы.
* Вид с точки зрения **реализации (implementation view)** охватывает компоненты и файлы, используемые для сборки и выпуска конечного программного продукта. Этот вид предназначен в первую очередь для управления конфигурацией версий системы, составляемых из независимых (до некоторой степени) компонентов и файлов, которые могут по-разному объединяться между собой.
* Вид с точки зрения **развертывания (deployment view)** охватывает узлы, формирующие топологию аппаратных средств системы, на которой она выполняется. В первую очередь он связан с распределением, поставкой и установкой частей, составляющих физическую систему.

Каждый из перечисленных видов может считаться вполне самостоятельным, так что члены группы проекта могут сосредоточиться на изучении только тех аспектов архитектуры, которые касаются непосредственно их. Правда, нельзя забывать о том, что эти виды взаимодействуют друг с другом, что неудивительно, так как они представляют разные взгляды на одну систему. Например, узлы вида с точки зрения развертывания содержат компоненты, описанные для вида с точки зрения реализации, а те, в свою очередь, представляют физическое воплощение классов, интерфейсов, коопераций и активных классов из видов с точки зрения проектирования и процессов.

## 01.03. Национальный стандарт Российской Федерации ГОСТ Р 53622-2009 «Информационные технологии. Информационно-вычислительные системы. Стадии и этапы жизненного цикла, виды и комплектность документов. Information technologies. Information-computing systems. Life cycle stages and steps, kinds and completeness of the documents. ОКС 03.120.10» в управленческом соглашении проекта ПО

Стандарт входит в перечень нормативных документов, используемых при создании и использовании информационно-вычислительных систем.

Настоящий документ устанавливает требования к последовательности

создания и использования информационно-вычислительных систем, включающей стадии и этапы, состав видов документов, разрабатываемых на

стадиях и этапах жизненного цикла информационно-вычислительных систем

от формирования требований до снятия с эксплуатации.

Стандарт предполагает разработку стандартов на требования к составу и содержанию документов, разрабатываемых на каждой стадии жизненного цикла (ЖЦ) информационно-вычислительной системы, прежде всего технического задания (ТЗ) на создание информационно-вычислительной системы, конструкторских документов, эксплуатационных документов, организационно-распорядительных документов. До принятия таких стандартов предписано руководствоваться стандартами систем ЕСПД и ИТ.

**Термины и определения в стандарте Г****ОСТ Р 53622-2009:**

**Аппаратно-программная платформа:** Единый комплекс средств вычислительной техники и системных программ.

**Данные:** Информация, представленная на электронном носителе в цифровой форме, пригодной для обработки программами вычислительной техники.

**База данных:** Совокупность данных, организованных по определенным

правилам, предусматривающим общие принципы описания, хранения и манипулирования данными, независимая от прикладных программ.

**Вычислительные средства (средства вычислительной техники):** Технические средства, непосредственно осуществляющие обработку данных.

**Информационно-вычислительная система (программно-технический комплекс):** Совокупность данных (баз данных) и программ, функционирующих на вычислительных средствах как единое целое для решения определенных задач.

**Информационные технологии:** Программы вычислительных средств, выполняющие сбор, накопление, хранение, поиск, обработку и выдачу данных

(информации).

**Конструкторский документ:** Документ, описывающий состав, структуру, алгоритмы обработки данных и методы их реализации, правила функционирования и применения информационно-вычислительной системы и

ее составных частей, предназначенный для разработчика на всех стадиях жизненного цикла.

**Программное обеспечение (программа, программное средство):** Упорядоченная последовательность инструкций (кодов) для вычислительного

средства, находящаяся в памяти этого средства и представляющая собой описание алгоритма управления вычислительными средствами и действий с данными.

**Программно-информационный продукт:** Программы, базы данных, электронные издания, мультимедийные приложения и им соответствующие эксплуатационные документы, предназначенные для поставки потребителю (пользователю).

**Система управления базами данных:** Программа, обеспечивающая процессы описания, хранения и манипулирования данными в базах данных.

**Стадия:** Часть процесса создания ИВС, характеризующаяся однородными по содержанию и результатам видам деятельности с заданным завершенным результатом.

**Технические средства:** Аппаратные средства, используемые для сбора, обработки, хранения, манипуляции и выдачи данных.

**Техническое задание:** Организационно-распорядительный документ, содержащий технические требования к информационно-вычислительной системе и порядку ее создания.

**Цифровые информационные ресурсы:** Переведенная в цифровой код информация в форме данных, баз данных и программно-информационных продуктов, которая обрабатывается с использованием средств вычислительной техники.

**Эксплуатационный документ:** Документ, описывающий состав, структуру и правила применения информационно-вычислительной системы и ее компонент, предназначенный для пользователей на стадии эксплуатации.

**Основные положения стандарта ГОСТ Р 53622-2009:**

Средства вычислительной техники разрабатываются как продукция производственно-технического назначения по государственным стандартам системы "Разработка и постановка продукции на производство".

Программы разрабатываются как продукция производственно-технического назначения по государственным стандартам "Единой системы программной документации".

Системные программы (операционные системы и средства их расширения) разрабатываются, как правило, под конкретные типы средств вычислительной техники.

Программы общего назначения (системы управления базами данных, редакторы и пр.) разрабатываются под созданные аппаратно-программные платформы и в совокупности с ними представляют собой интегрированную среду для разработки и реализации прикладных программ.

Вычислительная техника, системные программы и программы общего назначения при создании и использовании конкретной информационно-вычислительной системы (ИВС) рассматриваются как готовые (стандартные) изделия.

Информационно-вычислительные системы разрабатываются в соответствии с настоящим стандартом и техническим заданием на создание ИВС.

Исходными данными для создания ИВС является перечень решаемых задач и им соответствующих типов и объемов данных.

Объектом разработки при создании ИВС являются прикладные программы различного назначения, в соответствии с которыми обосновываются и выбираются конкретные вычислительные средства, системные программы и программы общего назначения.

**Виды документов, разрабатываемых при создании и использовании ИВС**

Основным документом при создании ИВС является ТЗ на создание (разработку) ИВС, в котором устанавливаются структура ИВС, составные части ИВС и перечень подлежащих разработке ТЗ на создание (разработку) составных частей ИВС.

ТЗ на создание (разработку) составных частей ИВС разрабатываются на

основе ТЗ на создание (разработку) ИВС.

ТЗ на создание (разработку) ИВС и ТЗ на создание (разработку) составных частей ИВС должны являться, как правило, результатами выполнения соответствующих научно-исследовательских работ.

**Основные стадии и этапы жизненного цикла ИВС:**

**Жизненный цикл** создания (разработки) и использования ИВС представляет собой последовательность стадий работ, включающих однородные по содержанию и результатам этапы работ.

**Проведение научно-исследовательских работ** - обоснование состава решаемых задач, структуры и состава ИВС и подготовка проекта ТЗ на создание (разработку) ИВС включает:

* обоснование требований к ИВС и ее составным частям;
* подготовку проекта ТЗ на создание (разработку) ИВС (составной части
* ИВС);
* согласование и утверждение ТЗ на создание (разработку) ИВС

**Проектирование** (эскизное, техническое) ИВС включает:

* обоснование состава вычислительных средств;
* обоснование состава системных программ;
* обоснование состава программ общего назначения;
* обоснование и описание технических решений по прикладным программам.

Результаты проектирования оформляются в виде конструкторских документов - описаний проектов ИВС.

**Реализация проекта** (рабочее проектирование опытного образца

ИВС), включающая:

* уточненный состав вычислительных средств;
* уточненный состав системных программ и программ общего назначения;
* прикладные программы в исполняемом коде;
* откорректированные конструкторские документы;
* эксплуатационные документы;
* технические условия (для тиражирования ИВС).

**Внедрение** (адаптация) опытного образца ИВС в конкретных условиях

применения, включающая:

* уточненный состав вычислительных средств;
* уточненный состав системных программ и программ общего назначения;
* прикладные программы в исполняемом коде;
* откорректированные конструкторские и эксплуатационные документы;
* откорректированные технические условия (для тиражирования).

**Эксплуатация ИВС и сопровождение ИВС** включают:

* ТЗ на внесение изменений в прикладные программы;
* проекты реализации изменений;
* доработанные прикладные программы;
* уточненный состав вычислительных средств, системных программ и
* программ общего назначения;
* откорректированные конструкторские и эксплуатационные документы.

**Снятие с эксплуатации**, включающее:

* архивирование программ;
* утилизацию аппаратных средств.

## 01.04. Интегрированные среды разработки в менеджменте создания ПО. Языковое и управленческое соглашения проекта

**Интегрированная среда разработки (IDE)**

IDE - Integrated Development Environment, Интегрированная среда разработки - система программных средств, используемая программистами для разработки программного обеспечения.

В работе любого современного разработчика программного обеспечения существуют инструменты, ставшие практически обязательными для каждодневного применения. Если говорить о программировании на языке C++, таким инструментом, в первую очередь, будет, конечно, интегрированная среда разработки (IDE). Конечно, опытный программист может обойтись и без использования IDE, но это существенно скажется на таких моментах как скорость разработки, качество кода и, конечно же, на удобстве самого процесса написания этого кода.

Независимо от того, являетесь ли вы опытным разработчиком или только учитесь программировать, важно знать обо всех новых и уже существующих интегрированных средах разработки.

Для начала напомним разницу между текстовым редактором, применяемым для написания программ и полноценной IDE. Если упростить, текстовый редактор – это программа, с помощью которой разработчик взаимодействует непосредственно с кодом как c текстом. Используя клавиатуру, с помощью текстового редактора разработчик вводит символы и ключевые слова языка программирования, описывая этими словами некий алгоритм, понятный в дальнейшем компьютеру. Код, написанный на C++ с помощью текстового редактора кода, нужно будет в дальнейшем отправить на компилирование в другую программу – компилятор, который не входит в состав текстового редактора.

Интегрированная среда разработки, в свою очередь, это некий комплекс отдельных инструментов разработчика, позволяющий не только написать код программы, но и скомпилировать этот код в готовое приложение, провести тестирование этой программы, поиск ошибок, рефакторинг и другие действия. В состав современных IDE, как отдельные элементы уже входят:

* текстовый редактор;
* компилятор (или интерпретатор – для интерпретируемых языков программирования);
* отладчик;
* система управления версиями;
* другие инструменты.

Современные IDE могут существенно расширять свой функционал благодаря различным плагинам. Стоит отметить, что некоторые текстовые редакторы (например, Vim, Visual Studio Code, Emacs), благодаря доступным плагинам, также расширяют свой функционал, доводя его до практически возможностей IDE. Однако, это все же два разных инструмента разработчика. Текстовый редактор предназначен скорее для внесения мелких изменений в код отдельных файлов будущей программы. IDE, в свою очередь, работает с программой комплексно, рассматривая ее в целом как «проект» или набор «проектов», а не как отдельные текстовые файлы с кодом. Отсюда и разница в доступных возможностях этих инструментов.

**Visual Studio**

Это широко известная полнофункциональная среда разработки от компании Microsoft, которая позволяет работать с такими платформами, как Windows, Интернет, облако и Android. Возможности IDE Visual Studio позволяют правильно и эффективно писать код, реорганизовывать, анализировать и исправлять проблемы с кодом. Система IntelliSense предсказывает API-интерфейсы по мере их написания и выполняет автоматическое завершение для повышения скорости и эффективности разработки. Эта линейка программного обеспечения включает в себя множество инструментов для тестирования совместимости. Благодаря своей гибкости Visual Studio является отличным инструментом для студентов и профессионалов. Именно эта IDE позволяет новичку легче всего войти в процесс разработки.

Официальный сайт - https://visualstudio.microsoft.com/ru/

Поддерживаемые языки: Ajax, ASP.NET, DHTML, JavaScript, JScript, Visual Basic, Visual C#, Visual C++, Visual F#, XAML и другие.

**Особенности:**

* Свой компилятор – MSVC.
* Бесплатно распространяется Visual Studio Community, с достаточным набором возможностей.
* Платные версии могут предоставляться учебным заведениям и студентам бесплатно.
* Огромная библиотека расширений, которая постоянно увеличивается;
* IntelliSense.
* Настраиваемая панель и закрепляемые окна.
* Простой рабочий процесс и файловая иерархия.
* Статистика мониторинга производительности в режиме реального времени.
* Инструменты автоматизации.
* Легкий рефакторинг и вставка фрагментов кода.
* Поддержка разделенного экрана.
* Список ошибок, который упрощает отладку.
* Проверка утверждения при развертывании приложений с помощью ClickOnce, Windows Installer или Publish Wizard.
* Большое количество настроек среды разработки под «себя», благодаря встроенным механизмам и доступным дополнениям.

**Минусы IDE Visual Studio:**

* Версии Visual Studio Professional и Visual Studio Enterprise являются платными.
* Тяжеловесная IDE, достаточно требовательная к железу.
* Ряд разработчиков считают ее избыточной для создания мелких проектов.
* Достаточно маленькая скорость запуска проектов и приложений.
* Не работает на Linux.
* Платные версии с расширенным функционалом распространяются по подписке: Professional – от 45$ в месяц, Enterprise – от 250$ в месяц.

Поскольку Visual Studio является очень тяжелой IDE, для открытия и запуска приложений требуются значительные ресурсы. Поэтому на некоторых устройствах внесение простых изменений может занять много времени. Для простых задач целесообразно использовать компактную версию.

**QT Creator**

QT Creator - еще одна достаточно удобная и быстрая интегрированная среда разработки C++, предлагающая интересные возможности для разработчика. Эта кроссплатформенная IDE имеет современный редактор кода C++, встроенный удобный графический интерфейс, дизайнер форм, инструменты навигации и многое другое. Среди разработчиков многие выбирают его за его API и библиотеки, поскольку они последовательны, подробны, удобны и хорошо документированы.

QT Creator поставляется с полным набором инструментов разработчика, которые предназначены для одновременного создания приложений и пользовательских интерфейсов, а затем для дальнейшего их развертывания в различных мобильных ОС и настольных ПК. Часто отмечают простоту и отзывчивость интерфейса. В себя он включает удобный редактор кода с возможностью подсветки синтаксиса и автозаполнением, возможность создания интерфейса пользователя используя перетаскивания, инструмент визуальной отладки и др.

Официальный сайт: https://www.qt.io

**Особенности IDE QT Creator:**

* Наличие бесплатной версии.
* Доступны версии под Linux, macOS, Windows.
* Поддерживаемые компиляторы:

GCC / G++;

MinGW;

Linux ICC;

Clang;

Nim;

QCC;

MSVC.

* Доступны расширения, позволяющие работать с такими языками как Python, Ruby, Java, PHP и другими.
* Интеллектуальная система подсказок.
* Возможность компиляции проектов для различных систем.

**Минусы IDE QT Creator:**

* Платная версия от 459$ в месяц;
* Бесплатная версия имеет ряд функциональных ограничений.

**Eclipse**

IDE Eclipse, основанный на платформе Eclipse, является одной из ведущих IDE для разработчиков на C ++ и C. Eclipse предлагает поддержку разработки проектов и управляемую среду разработки с различными наборами инструментов. Он также поддерживает различные инструменты, сортировку, навигацию по гиперссылкам, редактор кода с функцией подсветки синтаксиса, инструменты визуальной отладки и многое другое.

Он может оказаться полезным, как для новичков, так и для профессионалов. Первоначально создаваемый как среда для Java-разработки сегодня Eclipse имеет широкий диапазон возможностей благодаря большому количеству плагинов и расширений. Помимо средств отладки и поддержки Git / CVS, стандартная версия Eclipse поставляется с инструментами Java и Plugin Development Tooling. Если вам этого недостаточно, доступно много других пакетов: инструменты для построения диаграмм, моделирования, составления отчетов, тестирования и создания графических интерфейсов. Клиент Marketplace Eclipse открывает пользователям доступ к хранилищу плагинов и информации.

Это бесплатное ПО с открытым исходным кодом, работающее со всеми основными ОС, включая Windows, Mac OS X и Linux. Eclipse может похвастаться удобным интерфейсом.

**Особенности IDE Eclipse:**

* Работает на таких ОС - Windows, macOS, Linux.
* Работает с такими языками - C, C++, Java, Perl, PHP, Python, Ruby и другие.
* Работает с такими компиляторами, как:

GCC / G++;

MinGW;

Cygwin.

* Простота как установки, так и применения.
* Удобный механизм разработки графических интерфейсов.
* Инструмент с открытым исходным кодом.
* Улучшения Java IDE, такие как иерархические представления вложенных проектов.
* Интерфейс, ориентированный на задачи, включая уведомления в системном трее.
* Автоматическое создание отчетов об ошибках.
* Параметры инструментария для проектов JEE.
* Интеграция с Junit.
* Распространяется бесплатно.
* Большое количество доступных модулей расширения – многие разработчики отмечают, что возможности модулей этой IDE значительно опережают таковые у других сред разработки.

**Минусы IDE Eclipse:**

Ряд разработчиков отмечают неудобство использования механизмов автодополнения при написании кода.

**NetBeans**

Бесплатная среда разработки с открытым исходным кодом. Подходит для редактирования существующих проектов или создания новых. Спонсируемый компанией Oracle, проект IDE NetBeans был передан в 2016 году в руки фонда Apache. На сегодня NetBeans поставляется с открытым исходным кодом. Написанная на Java, эта IDE имеет возможность работать на всех системах, которые поддерживают Java - Windows, Linux, macOS и Solaris. NetBeans IDE выглядит как интерфейс с функцией перетаскивания, совместно со списком шаблонов проектов. Среда в основном используется для разработки Java приложений, но можно устанавливать пакеты, поддерживающие другие языки.

Официальный сайт NetBeans - https://netbeans.apache.org

**Особенности IDE NetBeans:**

* Находится под управлением Apache Software Foundation.
* Поддерживает такие языки - Java, HTML, HTML 5, C, C++, PHP и другие.
* Работает на Windows, Linux, macOS и Solaris.
* Поддерживается работа с такими компиляторами:

Oracle Solaris Studio;

GCC/G++;

CLang/LLVM;

Cygwin;

MinGW.

* Удобный интуитивный инструмент построения графического интерфейса GUI Builder.
* Поддержка Qt Toolkit;
* Поддержка Fortan и Assembler;
* Большое количество расширений.
* Кроссплатформенность.
* Интеллектуальное автозавершение кода.
* Большое сообщество.

**Минусы IDE NetBeans:**

* Пользователи отмечают долгий запуск приложения.
* Ряд пользователей отмечают неудобство работы с дебаггером.

**IntelliJ IDEA**

Еще одна IDE, разработанная Jet Brains. Она предлагает пользователям бесплатную версию Community Edition, 30-дневную бесплатную ознакомительную версию Ultimate Edition и годовую подписку на версию Ultimate Edition за $533 — $693. IntelliJ IDEA поддерживает Java 8 и Java EE 7, обладает обширным инструментарием для разработки мобильных приложений и корпоративных технологий для различных платформ. Если говорить о цене, IntelliJ является прекрасным вариантом из-за огромного списка функций.

Поддерживаемые языки программирования: AngularJS, CoffeeScript, HTML, JavaScript, LESS, Node JS, PHP, Python, Ruby, Sass, TypeScript и другие.

**Особенности IDE IntelliJ IDEA:**

* Расширенный редактор баз данных и дизайнер UML.
* Поддержка нескольких систем сборки.
* Пользовательский интерфейс тестового запуска приложений.
* Интеграция с Git.
* Поддержка Google App Engine, Grails, GWT, Hibernate, Java EE, OSGi, Play, Spring, Struts и других.
* Встроенные средства развертывания и отладки для большинства серверов приложений.
* Интеллектуальные текстовые редакторы для HTML, CSS и Java.
* Интегрированный контроль версий.
* AIR Mobile с поддержкой Android и iOS.

**Недостатки:**

* Эта среда разработки JavaScript требует времени и усилий на изучение, поэтому может оказаться не лучшим вариантом для начинающих.
* Много сочетаний горячих клавиш, которые нужно просто запомнить.
* Некоторые пользователи жалуются на неуклюжий интерфейс.

**RubyMine**

Премиум IDE, разработанная компанией Jet Brains. Предлагается 30-дневная бесплатная ознакомительная версия, полная версия стоит $210 — $687 в год. Удобная навигация, логичная организация рабочего процесса и совместимость с большинством платформ делают RubyMine одним из популярных инструментов для разработчиков.

**Поддерживаемые языки:** CoffeeScript, CSS, HAML, HTML, JavaScript, LESS, Ruby и Rails, Ruby и SASS.

**Особенности:**

* Сниппеты кода, автоматическое заполнение и автоматический рефакторинг.
* Дерево проектов, которое позволяет быстро анализировать код.
* Схема модели Rails.
* Просмотр проекта Rails.
* RubyMotion поддерживает разработку под iOS.
* Поддержка стека включает в себя Bundler, pik, rbenv, RVM и другие.
* Отладчики JavaScript, CoffeeScript и Ruby.
* Интеграция с CVS, Git, Mercurial, Perforce и Subversion.

**Недостатки среды разработки:**

* Чтобы RubyMine работала бесперебойно, компьютеру требуется не менее 4 ГБ оперативной памяти.
* Некоторые пользователи также жалуются на отсутствие опций настройки GUI.

**Вместо заключения**

Существует достаточно большое количество интегрированных сред разработки. Для выбора IDE, как и для любого другого инструмента разработчика, справедлив индивидуальный подход. Отталкиваясь от текущих задач, базовой ОС, используемых языков, необходимых плагинов, требований команды и других, сугубо индивидуальных факторов разработки, выбор лучшей среды программирования будет различным.

В сравнительно несложных быстрых в исполнении проектов может оказаться вполне продуктивным использование под Windows пакета Visual Studio в бесплатной версии Community. В качестве лучшей бесплатной среды разработки C++под Linux можно использовать NetBeans.

## 01.05. Проектный менеджмент и применение в нём (в формате учебной деятельности) положений ГОСТ Р МЭК 61160-2015 «Проектный менеджмент. Документальный анализ проекта»

Не следует полагать, что к экспертизам учебного проекта – прототипа программного продукта и его альтернативных решений в статусе предполагаемого реинжиниринга уместно в полном объёме применять всю совокупность требований, относящихся к категории полноценного профессионального проектного менеджмента в соответствии с требованиями упомянутого в заголовке Национального стандарта РФ «ГОСТ Р МЭК 61160-2015 Проектный менеджмент. Документальный анализ проекта». [8]

Однако, имея ввиду наличие экспертных действий по адресу проектов ПО в составе практических работ по дисциплине, представляется весьма полезным как ознакомление с указанным стандартом, так и непосредственное использование его важнейших положений, не относящихся, впрочем, к организации профессиональной деятельности специальных экспертных комиссий и групп. Здесь речь идёт исключительно об эскизном варианте описания деятельности по экспертизе проектов ПО (к тому же учебных студенческих), выполняемой в весьма ограниченный временной период одиночным студентом (максимум, бригадой студентов от двух, до четырёх человек – по указанию преподавателя).

Именно в таком ракурсе в помощь выполнению практических работ по настоящей дисциплине далее приводятся отдельные выдержки из указанного стандарта, которыми рекомендуется воспользоваться в учебной работе.

Некоторые основные положения Национального стандарта РФ «ГОСТ Р МЭК 61160-2015 Проектный менеджмент. Документальный анализ проекта. Project management. Documentary analysis of the project. ОКС 03.100.40» обозначают следующее:

«Анализ проекта - консультативная деятельность, предназначенная прежде всего для обеспечения проверки работы группы проектирования, разработки рекомендаций (если это возможно) по улучшению продукции или процесса и их реализации. Таким образом, анализ проекта следует рассматривать как процедуру проверки соответствия результатов проектирования и разработки установленным требованиям.

Анализ проекта независимо от частоты или глубины его проведения не может заменить этапы определения параметров продукции, требований к ее конструкции и процессу управления проектированием и разработкой. Анализ проекта, используемый для контроля, может обеспечить необходимую проверку успешного завершения конструкторской разработки в установленный срок.

Анализ проекта не следует путать с оперативным управлением проектом. Руководитель проекта несет ответственность за конструкцию и окончательные решения в качестве ответной редакции на результаты анализа проекта и предложенные рекомендации. Анализ проекта, выполненный должным образом, повышает доверие к тому, что проектирование и разработка выполнены в соответствии с установленными требованиями на всех стадиях жизненного цикла продукции.

Применение настоящего стандарта должно быть адаптировано в соответствии с требованиями проектирования и разработки продукции или рассматриваемой задачи и организации, разрабатывающей проект».

**Целями анализа проекта** являются:

* проверка соответствия предложенного решения входным требованиям проектирования и разработки, включающим установленные общие требования эксплуатации, требования безотказности, предусмотренные затраты, требования безопасности, износостойкости, окружающей среды, электромагнитной совместимости, требования, связанные с человеческим фактором;
* проверка соответствия того, что предложенные решения наиболее устойчивы, эффективны и результативны для выполнения требований к продукции;
* обеспечение рекомендаций для выполнения исходных требований к конструкции;
* определение статуса конструкции с точки зрения полноты и завершенности чертежей и спецификаций;
* анализ объективных свидетельств, обеспечивающих верификацию выполнения проектирования и разработки;
* предложения по улучшениям.

**Анализ проекта** облегчает определение статуса проекта в соответствии с выполнением входных требований, идентификацию возможностей улучшения и побуждает к выполнению соответствующих действий по улучшению проекта. Анализ проекта ускоряет подготовку производства продукции за счет сокращения времени установления деталей конструкции и позволяет выпускать продукцию без частых прерываний. Анализ проекта может также стимулировать действия по улучшению продукции на более ранних стадиях».

**Применение**

Стадии жизненного цикла, на которых должен быть выполнен анализ проекта, необходимо определить на стадии планирования проектирования и разработки или выполнения задач проектирования. Факторами, влияющими на проектирование и разработку, являются требования потребителя, обязательные требования, особенности продукции и ее использования, последствия отказа и т.д.».

**«Нормативные ссылки», «термины и определения»** (выдержки для выполнения практических работ):

«МЭК 60050-191:1990 Международный электротехнический словарь. Глава 191. Надежность и качество обслуживания (IEC 60050-191:1990, International electrotechnical vocabulary; chapter 191: dependability and quality of service)

МЭК 62198 Менеджмент риска при проектировании. Руководство по применению (IEC 62198, Managing risk in projects - Application guidelines)».

«Анализ (review): Деятельность, предпринимаемая для установления пригодности, адекватности и результативности объекта для достижения установленных целей.

Проектирование и разработка (design and development): Совокупность процессов, переводящих требования в установленные характеристики или спецификации на продукцию, процесс или систему.

(Примечание 1 - Термины "проектирование" и "разработка" иногда используют как синонимы, а иногда для определения различных стадий процесса проектирования и разработки в целом).

(Примечание 2 - Для обозначения объекта проектирования и разработки могут быть использованы определяющие слова (например, разработка и проектирование продукции или разработка и проектирование процесса)).

**Анализ проекта (design review):** Запланированный, документированный независимый анализ существующего или предполагаемого проекта.

(Примечание 1 - Цели включают анализ способности проекта соответствовать установленным требованиям, идентификацию фактических или возможных недостатков для предполагаемого улучшения).

(Примечание 2 - Анализа проекта самого по себе недостаточно для обеспечения качества проекта).

(Примечание 3 - Проект может относиться к продукции или к процессу)».

«**Верификация (verification):** Подтверждение, посредством представления объективных свидетельств того, что установленные требования выполнены».

«**Валидация (validation):** Подтверждение посредством представления объективных свидетельств того, что требования для конкретного использования или применения выполнены».

**Управление процессом анализа проекта:**

«Система менеджмента анализа проекта должна включать:

a) идентификацию функций и действий, необходимых для анализа проекта;

b) установление целей и плана анализа проекта для достижения этих целей с помощью планирования;

c) обеспечение своевременного выполнения действий анализа проекта на всех применимых стадиях проектирования;

d) анализ критериев и методов анализа выполнения, проверки и приемки продукции;

e) обеспечение ресурсов и информации, необходимой для выполнения необходимого анализа проекта;

f) мониторинг действий анализа проекта, измерение и анализ результатов для непрерывного улучшения»;

а также «отчеты, являющиеся результатом анализа проекта» (в учебном процессе по дисциплине «Информационный менеджмент» - отчёты обучающихся о выполненных практических работах).

**Процесс и содержание анализа проекта:**

«Главными целями анализа проекта являются обеспечение уверенности в том, что требования к продукции идентифицированы, выполнены эффективно и результативно и ведут к созданию надежной и полезной продукции. В частности, цели анализа проекта включают (и могут быть дополнены) проверку того, что:

1. продукция соответствует установленным требованиям и ее производство рентабельно;
2. план проектирования и разработки включает стратегию подтверждения и демонстрации того, что функциональные требования выполнены;
3. конструкция соответствует требованиям интерфейса и потребителя;
4. продукция и ее элементы, при необходимости, могут быть безопасно и экономно ликвидированы;
5. использованы необходимые методы проектирования, производства и монтажа;
6. затраты оптимизированы после рассмотрения всех требований к продукции;
7. принят во внимание график выполнения работ;
8. обеспечены рекомендации по улучшению конструкции;
9. компоненты использованы в соответствии с их установленными функциями и нагрузками;
10. учтены более ранние рекомендации;
11. чертежи, программное обеспечение и спецификации являются полными и завершенными».

(*В практических работах по дисциплине «Информационный менеджмент» в анализ включаются только те пункты и в такой мере, которая соответствует уровню и характеру обследования учебного проекта – прототипа*).

**Развивающие указания в частях первоначального и концептуального анализа проекта**

(*для учебных проектов* *применять выборочно*)

Цели **первоначального анализа проекта** включают:

* проверку идентификации и разъяснение, при необходимости, требований потребителя;
* проверку идентификации применимых обязательных требований;
* проверку включения во входные данные проектирования других необходимых технических требований;
* проверку идентификации организационных и технических интерфейсов;
* проверку проведения оценки риска, соответствующего конструкции.

Целями **концептуального анализа** проекта являются:

1. интерпретация, определение и согласование требований потребителя и требований к продукции;
2. установление обязательных, желательных и дополнительных характеристик;
3. анализ взаимодействий членов группы проектирования, ответственных за проектирование, продажи, изготовление, испытания, контроль, установку, эксплуатацию и техническое обслуживание и т.д.;
4. анализ условий и технологий, доступных в настоящее время для удовлетворения требований потребителя, рынка и требований безопасности;
5. анализ функционирования, безопасности и информации об ответственности на основе отчетов об эксплуатации аналогичной продукции или аналогичного производства;
6. анализ планов и графиков разработки и проектирования;
7. анализ предложенной концепции проектирования».

**Жизненные циклы продукции и затраты на их обеспечение**

«В процессе проектирования и разработки на стадии жизненного цикла проводят анализ данных жизненного цикла, целей и вариантов конструкции. Анализ может охватывать такие вопросы, как выбор материалов, данные о производительности, установленная стоимость, срок службы, затраты на эксплуатацию, вывод из эксплуатации и распоряжение продукцией.

Факторы технико-экономического анализа, критичные для продукции, должны быть исследованы на стадии предварительного анализа проекта. Анализ проекта, начиная с начальной стадии проекта, может способствовать экономии средств путем определения требований к затратам».

**Готовность спроектированного изделия, коэффициент готовности**

«Вопросы, относящиеся к готовности, включают:

1. наиболее вероятные причины отказов, например, наиболее частых отказов, выявленных с помощью анализа видов и последствий отказа (FMEA) и/или анализа дерева отказов (FTA);
2. требования к коэффициенту готовности, например, средний коэффициент готовности (мгновенной), коэффициент готовности, ресурс;
3. сопоставление прогнозируемого коэффициента готовности продукции или процесса с требованиями, включая предположения, модели и базы данных;
4. действия по улучшению коэффициента готовности, например, деление на модули, введение резервирования, заменяемых элементов, ограничений рабочих характеристик, контроль окружающей среды, использование быстрых отключений;
5. историю эксплуатации аналогичной и конкурирующей продукции;
6. влияние рабочей среды и технического обслуживания в условиях эксплуатации на коэффициент готовности;
7. специальное оборудование и инструмент для эксплуатации и обслуживания;
8. влияние пользователя на коэффициент готовности продукции, например, обучения персонала, неправильного использования оборудования, незаконного присвоения инструмента и компонентов;
9. намеченный план определения коэффициента готовности и критериев согласия, например, количество объектов испытаний, период, условия, стадия жизненного цикла для проведения испытаний».

«Вопросы, относящиеся к обеспечению качества, включают (но не ограничены):

1. элементы, имеющие отношение к требованиям потребителя и их удовлетворению;
2. сопоставление требований потребителя с техническими спецификациями на материалы, продукцию и процессы;
3. валидацию конструкции на основе испытаний опытного образца;
4. способность продукции функционировать в ожидаемых условиях использования и окружающей среды;
5. непреднамеренное и неправильное использование продукции;
6. соответствие обязательным требованиям, требованиям национальных и международных стандартов практикам организации;
7. сопоставление продукции с конкурирующей продукцией;
8. сопоставление с аналогичными конструкциями и анализ истории внутренних и внешних проблем для исключения их повторения;
9. элементы, имеющие отношение к требованиям на продукцию и требованиям к ее обслуживанию;
10. допустимые отклонения и сравнение их с возможностями процесса;
11. критерии приемки/отклонения;
12. простоту сборки и установки, требования к хранению, сроку годности и распоряжению (утилизации);
13. характеристики неопасного отказа;
14. эстетические требования и критерий приемки;
15. способность диагностирования и решения проблем;
16. маркировку, предупреждающие знаки, идентификацию, требования прослеживаемости, инструкции пользователя и контроль документации;
17. анализ и использование типовых деталей;
18. элементы, относящиеся к требованиям процесса;
19. технологичность конструкции, включая установленные требования процесса, механизацию, автоматизацию, сборку и установку компонентов;
20. возможность контроля и испытаний продукции, включая специальные требования контроля и испытаний;
21. требования проверки калибровки средств измерений;
22. требования к материалам, компонентам и сборочным единицам, включая одобренных поставщиков, а также доставку потребителю;
23. требования к упаковке, обработке, хранению и сроку годности, особенно к факторам безопасности входящих и выходящих объектов;
24. элементы, относящиеся к верификации конструкции;
25. альтернативные вычисления, выполняемые для проверки правильности сделанных вычислений и исследований;
26. испытания, например, модели или опытного образца, определение программы испытаний и документирование результатов;
27. независимую проверку правильности выполненных вычислений и других действий проектирования и разработки;
28. контроль конфигурации, адекватность системы идентификации;
29. серийное или разовое изготовление, анализ и верификацию информации о продукции, местоположении отчетов и ссылок, регистрацию кодированной или не кодированной информации».

«Вопросы, связанные с человеческим фактором, включают:

a) особенности и сложность информации, используемой оператором при обработке, контроле и регулировке продукции или процесса;

b) эффективность информации, включая сообщения СМИ, используемой оператором для управления продукцией или процессом;

c) соответствие продукции ожиданиям человека, его нормальные реакции и реакции в экстремальной ситуации;

d) адекватность и пригодность применяемых предупреждений и инструкций для эксплуатации, установки, технического обслуживания, сборки и распоряжения;

e) анализ интерфейса пользователя на соответствие требованиям технического обслуживания в ожидаемых и возможных условиях окружающей среды (техническое обслуживание, маркировка, рабочее место);

f) анализ комфортности работы оператора при использовании продукции;

g) доступность инструмента, используемого с продукцией;

h) доступность инструмента, используемого при выполнении действий с продукцией;

i) анализ степени учета вероятных квалификации и навыков персонала, участвующего в монтаже, эксплуатации, техническом обслуживании продукции, при проектировании продукции или процесса;

j) анализ требований к отображению информации (формат, объем, возможность прокрутки);

k) простоту использования программного обеспечения и документации;

l) необходимость и вид звуковых и визуальных сигналов тревоги для индикации функционирования, состояния и появления отказов;

m) анализ критериев улучшения работы человека и простоты взаимодействия человека с продукцией;

n) адекватность, точность, однозначность, простоту использования и понимание всей документации и иллюстраций, необходимых для эксплуатации и технического обслуживания продукции;

o) квалификацию операторов, необходимую для эксплуатации продукции, исследуемую при разработке потребностей в обучении;

p) возможные трудности операторов с недостаточной квалификацией при работе с продукцией;

q) возможные трудности, вызванные действиями операторов, неверно использующих продукцию;

у) анализ технических требований, рекламной информации, инструкций по монтажу, предупреждений, маркировки и другой документации для соответствующего эффективного функционирования продукции».

**Библиография Национального стандарта РФ «ГОСТ Р МЭК 61160-2015 Проектный менеджмент. Документальный анализ проекта. Project management. Documentary analysis of the project. ОКС 03.100.40»:**

IEC 60300-1:2003 Dependability management - Part 1: Dependability management systems.

IEC 60300-2:2004 Dependability management - Part 2: Guidelines for dependability management.

IEC 60721-2 (all parts) Classification of environmental conditions - Part 2: Environmental conditions appearing in nature.

IEC 60721-3 (all parts) Classification of environmental conditions - Part 3: Classification of groups of environmental parameters and their severities.

IEC 61078:1991 Analysis techniques for dependability - Reliability block diagram method.

ISO 9000:2005 Quality management systems - Fundamentals and vocabulary.

ISO 9001:2008 Quality management system – Requirements.

## 01.06. Глоссарий (к формированию онтологического соглашения проекта)

**Класс (class)** - это описание совокупности объектов с общими атрибутами, операциями, отношениями и семантикой. Класс реализует один или несколько интерфейсов.

**Интерфейс (interface)** - это совокупность операций, которые определяют определенную службу (сервис, набор услуг), которые предоставляет класс или компонент. Интерфейс отвечает за видимое извне поведение элемента. С помощью интерфейса поведение класса или компонента может быть представлено полностью или частично, он определяет только спецификации операций (сигнатуры), но никогда их практическую реализацию.

**Кооперация (collaboration)** определяет взаимодействие, она представляет собой совокупность ролей и других элементов, которые, работая вместе, производят некоторый кооперативный эффект, не сводящийся обычно к сумме слагаемых. Кооперация, таким образом, имеет как структурный, так и поведенческий аспект. Один и тот же класс может принимать участие в нескольких кооперациях, которые представляют собой реализацию образцов поведения, определяющих систему.

**Прецедент (use case)** это описание последовательности выполняемых системой действий, которая производит наблюдаемый результат, значимый для какого-то определенного актера (actor). Основная цель применения прецедентов структурировать поведенческие сущности системы. Реализуются прецеденты посредством кооперации.

**Три другие сущности:** активные классы, компоненты и узлы подобны классам, они описывают совокупности объектов с общими атрибутами, операциями, отношениями и семантикой. Тем не менее, они в достаточной степени отличаются друг от друга и от классических классов, и, учитывая их важность при моделировании определенных аспектов объектно-ориентированных систем, заслуживают персонального рассмотрения.

**Активным классом (active class)** называется класс, объекты которого вовлечены в один или несколько процессов или нитей (threads), и поэтому могут инициировать управляющее воздействие. Активный класс практически полная копия обычного, за исключением того, что его объекты представляют собой элементы, деятельность которых осуществляется одновременно с деятельностью других элементов.

Два последних элемента компоненты и узлы также имеют свои особенности. Они соответствуют физическим сущностям системы, а предыдущие пять концептуальным и логическим сущностям.

**Компонент (component)** это физическая заменяемая часть системы, которая соответствует некоторому набору интерфейсов и обеспечивает его реализацию. Классический пример компонентов компоненты COM+. Компонент обычно представляет собой физическую упаковку логических объектов, таких как классы, интерфейсы и кооперации.

**Узел (node)** это элемент реальной (физической) системы, который существует во время функционирования программного продукта и представляет собой некоторый вычислительный ресурс, обычно обладающий, как минимум, некоторым объемом памяти, а часто еще и возможностью обработки. Совокупность компонентов может размещаться на узле, а также мигрировать с одного узла на другой.

**Дополнительно**

**Каталожное описание продукта (КО)** – документ Федеральной системы каталогизации продукции, содержащий сведения о характеристиках конкретной продукции, заполняемый согласно требованиям соответствующего стандартного формата описания. Одним из ключевых факторов, влияющих на эффективность процессов послепродажного обслуживания техники, является качество их информационного обеспечения.

Это относится как к информации о собственно образцах техники (электронные каталоги, перечни запасных частей, прейскуранты, электронная эксплуатационная документация и т.п.), так и к информации об организации бизнес-процессов (прохождение заявок на запасные части и материалы, подготовка и обмен в электронном виде счетами и другими документами).

Эффективное формирование и использование этой информации, как показывает передовой отечественный и зарубежный опыт, возможно только на основе применения каталогизации предметов снабжения, поставляемых в сферу материально-технического обеспечения эксплуатации и ремонта сложной наукоемкой продукции.

В определенном смысле каталогизация представляет собой процесс формирования нормативно-справочной информации, то есть условно неизменной составляющей корпоративной информации, обеспечивающей создание единого информационного пространства. [9, 10]

**Заключение к разделу 01**

Ещё раз о сфере информационного менеджмента [11, 12].

«В узком понимании — это круг задач управления производственного и технологического характера в сфере основной деятельности организации, в той или иной мере использующих информационную систему и реализованные в ней информационные технологии.

В широком смысле — это совокупность задач управления на всех этапах жизненного цикла организации, включающая в себя действия и операции как с информацией в различных ее формах и состояниях, так и с организацией в целом на основе информации.

Информационный менеджмент охватывает все аспекты проблемы менеджмента в сфере создания и использования информационных ресурсов.

Цель информационного менеджмента — повышение эффективности деятельности предприятия на основе использования информационных систем и технологий.

Предмет информационного менеджмента — процессы создания, эксплуатации и развития информационной системы предприятия.

Область профессиональной деятельности менеджера — обеспечение эффективного управления информационными ресурсами и информационными системами на уровне организации, обеспечение использования информации как стратегического ресурса, организация систем управления в отрасли информационного бизнеса, совершенствование управления в соответствии с тенденциями социально-эконо­мического развития.

Объектами профессиональной деятельности менеджера являются различные информационные ресурсы и информационные системы организаций экономической, производственной и социальной сфер, информационные ресурсы и информационные системы подразделений систем управления государственных предприятий, акционерных обществ и частных фирм, а также различные организации в сфере информационного бизнеса. Профессиональная деятельность менеджера распространяется также на информационные ресурсы и информационные системы научно-производственных объединений, научных, конструкторских и проектных организаций, органов государственного управления и социальной инфраструктуры народного хозяйства.

Сферу информационного менеджмента составляет совокупность всех задач управления, связанных с формированием и использованием информации во всех ее формах и состояниях для достижения поставленных перед предприятием це­лей. При этом должны решаться задачи определения ценности и эффективности использования не только собственно информации (данных и знаний), но и других ресурсов предприятия, в той или иной мере входящих в контакт с информацией: технологических, кадровых, финансовых и т. д. В этих задачах управления в той или иной мере используются информационные системы и реализованные в них информационные технологии.

**Концепция информационного менеджмента** **объединяет следующие** **подходы**:

* экономический — вопросы привлечения новой документированной информации рассматриваются исходя из соображений полезности и финансовых затрат;
* аналитический — основывается на анализе потребностей пользователей в информации и коммуникациях;
* организационный — информационные технологии рассматриваются в их влиянии на организационные аспекты;
* системный — обработка информации рассматривается на основе целостного, системно ориентированного, всеохватывающего процесса обработки инфор­мации в организации, при этом особое внимание уделяется оптимизации ком­муникационных каналов, информации, материальных средств и других затрат, методов работы.

Основные направления информационного менеджмента:

* управление информационной системой на всех этапах ее жизненного цикла;
* стратегическое развитие ИС;
* ИС-маркетинг.

**Задачи информационного менеджмента:**

* формирование технологической среды информационной системы;
* развитие информационной системы и обеспечение ее обслуживания;
* планирование в среде информационной системы;
* формирование организационной структуры в области информатизации;
* использование и эксплуатация информационных систем;
* формирование инновационной политики и осуществление инновационных программ;
* управление персоналом в сфере информатизации;
* управление капиталовложениями в сфере информатизации;
* формирование и обеспечение комплексной защищенности информационных ресурсов».

Решение задач информационного менеджмента прежде всего опирается на применение соответствующих стандартов, где ведущее место отведено стандарту ITIL.

**Назначение ITIL**

«Библиотека инфраструктуры информационных технологий (Information Tech­nology Infrastructure Library, ITIL) в настоящее время фактически стала между­народным стандартом в сфере организации и управления информационными технологиями.

Библиотека ITIL изначально разрабатывалась Центральным агентством по вычислительной технике и телекоммуникациям Central Computer and Telecom­munications Agency, CCTA) при правительстве Великобритании. В настоящее время владельцем библиотеки является британская правительственная организация OGC (The Office of Government Commerce), одно из направлений деятельности которой — повышение эффективности и рентабельности использования информа­ционных технологий в государственных учреждениях Великобритании.

Значительная роль в развитии и популяризации ITIL принадлежит также неком­мерческому профессиональному сообществу ITSMF (IT Service Management Forum), объединяющему крупные организации, в том числе компании-производители (Microsoft, SUN, HP и IBM), а также частных лиц, работающих в области ИТ.

ITIL включает в себя описание различных видов деятельности в сфере ИТ, таких как управление проектами, управление закупками, ИТ-сервис, и состоит из ряда практических руководств, предоставляющих информацию об эффективном и рациональном использовании различных ИТ-сервисов и предоставлении их потребителям.

Философия библиотеки ITIL основана на общих схемах обеспечения качества (total quality frameworks), предлагаемых Европейской организацией управления качеством (European Foundation of Quality Management, EFQM) и стандартами серии ISO-9000. Эти системы качества поддерживаются путем предоставле­ния стандартизированного описания процессов с учетом передового опыта ИТ- менеджмента.

Библиотека ITIL предлагает структурированное описание наиболее часто ис­пользуемых ИТ-процессов, их целей и параметров, а также связей между отдель­ными ИТ-процессами, однако целью библиотеки ITIL не является предоставление описаний конкретных способов внедрения этих процессов».

(Прим.: В проектах, связанных с разработкой, улучшением и использованием информационных ресурсов необходимо дополнительно к выше приведённому учитывать требования Национального стандарта РФ ГОСТ Р 43.0.9-2017 «Информационное обеспечение техники и операторской деятельности. Информационные ресурсы». [13]. На основе этих требований формируются Паспорт информационного ресурса и целостное описание документографических систем в формате Каталожных описаний (КО)).

«Информационный менеджмент в организации решает стратегические, оператив­ные и административные задачи. К числу стратегических задач относятся: создание информационной инфраструктуры организации и управление информационными технологиями. Оперативные и административные задачи носят более узкий характер, ориентированный на выполнение работ по конкретным направлениям [11, 12].

# Раздел 02. Задания на выполнение практических работ и указания по их выполнению

## 02.01. Задания: состав, структура, последовательность выполнения

Задания, их структура и последовательность выполнения выстроены по вариативно-модульной системе в расчёте на то, что число практических занятий, их расчасовка по тем или иным причинам могут год от года видоизменяться.

За основу принята четырёхмодульная архитектура сопровождения дисциплины практическими занятиями – симметрично, видимо, по 4 – 8 часов учебной работы по каждому из модулей, чередующихся друг за другом в установленном здесь порядке:

**Практическая работа № 01. Выбор проекта – прототипа, составление и согласование перечня показателей анализа прототипа**

Изначально, готовясь к настоящему занятию в формате СРС (самостоятельной работы студента) обучающиеся внимательно знакомятся с вышеприведёнными в разделе 01 установочными и справочными материалами.

Далее, уже на первом занятии, обучающиеся единолично или в составе небольшой по численности участников бригады студентов (по указанию преподавателя) от 2-х до 4-х человек выбирают в качестве прототипа для анализа, экспертизы и последующего реинжиниринга некий проект- прототип по созданию программного продукта (ПП) или информационной системы, применительно к разработке которой ПП был создан и описан её автором.

Проект – прототип преимущественно учебный уровня бакалавриата, выполненный в качестве курсового проекта (курсовой работы) или, скорее всего, в качестве выпускной квалификационной работы бакалавра (ВКР) в предшествующие годы. С позиций освоения настоящей дисциплины, естественно, интерес представляют только те проекты, где представлен их информационный менеджмент.

Поэтому кафедра ИиППО предлагает на выбор студента и преподавателя набор такого рода проектов в количестве 15 - 30, хранящихся в информационном репозитории кафедры, причём, совершенно не обязательно, чтобы это были проекты, выпущенные непосредственно кафедрой ИиППО. Могут привлекаться к исследовательской работе и иные проекты (как, в частности, транслируемый далее пример проекта - прототипа), даже иностранные проекты, созданные на языках - оригиналах, например, на английском языке. Последний вариант может оказаться наиболее комфортным для иностранных студентов, обучающихся в РТУ МИРЭА.

Студент может представить на согласование преподавателю **собственный вариант проекта – прототипа**, принимаемый им в качестве прототипа выполняемой в самом ближайшем будущем выпускной квалификационной работы (ВКР). Такая постановка задачи всячески приветствуется - лишь бы в прототипе содержался материал, относимый к информационному менеджменту, в частности, к обеспечению ЖЦ и, желательно, к оценкам **устойчивости проекта** и **коэффициента запаса** изделия.

Первое занятие завершается тем, что после первоначального ознакомления с избранным проектом – прототипом и сопутствующих этому консультаций с преподавателем, обсуждениям в группе обучающиеся формируют и **согласовывают с преподавателем не менее 15 показателей** анализа (экспертизы) оценки качества и результативности информационного менеджмента показанного в прототипе, выбирая их из всего перечня, приведённого в разделе 01 настоящих указаний.

В их составе должны или могут присутствовать такие показатели как: устойчивость проекта, коэффициент запаса, обеспечение ЖЦ, ресурсоёмкость, кадровая востребованность и обеспечение лиц сопровождения проекта и его изделия на всём ЖЦ соответствующими инструкциями, коэффициент готовности, оценка и информационное обеспечение мер по диагностике, профилактикам, аварийно-восстановительных мер на всём ЖЦ, оценка проектных рисков, надёжность и нагруженность изделия, наличие и качество каталогизации изделия проекта, эргономические, технико-эстетические, дидактические и иные специфические составляющие в зависимости от жанра проекта, его юзабилити и т.п. Настоящий перечень не более, чем установочный – в окончательном виде он формируется студентом по согласованию с преподавателем и принимается за основу для последующей реализации.

Первая практическая работа считается благополучно выполненной, если в установленный расписанием учебных занятий срок с преподавателем согласован выбор прототипа и перечень позиций экспертизы прототипа.

**Практическая работа № 02. Экспертный анализ проекта – прототипа**

Готовясь к настоящему занятию в формате СРС (самостоятельной работы студента) обучающиеся повторно знакомятся с выше приведёнными в разделе 01 установочными / справочными материалами и изучают в предверии занятия избранный ими прототип, особое внимание уделяя наличию и правомерности использования в проекте – прототипе положений и требований соответствующих стандартов, в том числе обсуждаемых в разделе 01 настоящих материалов.

Непосредственно на занятиях (по расписанию занятий), консультируясь с преподавателем, опираясь на справочные и разъяснительные материалы раздела 01, обучающиеся step by step формируют (в табличной форме Word или сплошным текстом – перечнем) экспертные оценки по всем избранным ранее показателям экспертизы прототипа в контексте информационного менеджмента. В ситуациях, где какие-то показатели не могут быть оценены с позиций информационного менеджмента (например, потому, что в проекте отсутствуют материалы, соотносимые к эти позициям) просто делается запись об объективных причинах (например, отсутствии материалов) невыполнения оценки. Разумеется, это никак не плюс в адрес проекта – прототипа. Однако, в число «невыполнений» нельзя включать позиции, отвечающие достижению требуемого качества, работоспособности изделия на всём ЖЦ – такого рода проект должен быть отвергнут в качестве прототипа и вместо него следует избрать другой, более удачный, проект – прототип.

В итоге проделанного многопараметрического анализа делается обобщающий вывод о полноценности проекта – прототипа с позиций его обеспечения методами и мерами информационного менеджмента (особенно в частях, относящихся к созданию и сопровождению ПО), соответствия стандартам и пригодности к реинжинирингу путём дальнейших улучшений и замен в частях информационного менеджмента проекта (без изменений в технической сущности проектных решений). Принятие преподавателем сформулированных выводов в установленные расписанием занятий сроки свидетельствует о благополучном завершении практической работы № 02.

**Практическая работа № 03. Реинжиниринг информационного менеджмента проекта – прототипа и экспертная оценка модифицированной версии**

Изначально, готовясь к настоящему занятию в формате СРС (самостоятельной работы студента) обучающиеся внимательно знакомятся с выше приведёнными в разделе 01 установочными и справочными материалами.

Далее, по итогам проделанной в предшествующей работе экспертизы выдвигается альтернативный подход к мотивированно обновлённому обустройству информационного менеджмента проекта с опорой на позиции, изложенные в подразделе «01.04. Интегрированные среды разработки в менеджменте создания ПО. Языковое и управленческое соглашения проекта». В проект – прототип вводятся соответствующие изменения (желательно выделить их синим цветом шрифта на фоне чёрного цвета написания в оригинале).

Точно также, как в предшествующей практической работе по тем же самым признакам и свойствам производится экспертиза, но уже с обновлённым описанием информационного менеджмента. Производится сравнение результативности произведённого реинжиниринга менеджмента, что оформляется в форме заключения к практической работе №3.

Материалы работы, а главное, заключение по ней, являются основанием для преподавателя считать выполнение практической работы № 3 успешно завершённым.

**Практическая работа № 04. Разработка и представление итогового отчёта по совокупности выполненных заданий №№ 01, 02, 03**

Готовясь к предстоящей последней по дисциплине практической работе № 04 обучающимся следует заблаговременно внимательно изучить требования по составу и оформлению учебной отчётной документации (см. выше).

На самом практическом занятии, консультируясь в меру необходимости с преподавателем, следует составить и оформить, согласно установленным правилам и форматам, указанный отчёт в электронной форме.

В отчёт наряду с ранее сформированными текстами должны быть включены три дополнительных фрагмента, формируемых непосредственно на четвёртом занятии, а именно:

* Каталожное (или Техническое) описание модифицированного объекта с формулировкой в нём состава трёхзвенного проектного соглашения (языкового/платформенного, онтологического, управленческого).
* Справка, фрагмент или сообщение о выполнении закреплённых за дисциплиной требований компетенций (как в описании их индикаторов, так и связанных с этими индикаторами трудовых действий).
* Список использованных в работе источников, обязательно включающий адресную ссылку на обслуживающий менеджментом прототип, перечень стандартов, а также основное учебное пособие по дисциплине (см. выше), конспект лекций с презентационным пакетом к ней и настоящее учебно-методическое пособие.

Сформированный таким образом отчёт проверяется студентом на меру самостоятельности программой «Антиплаг», скрин акта проверки по которой приобщается к отчёту приложением. В приложениях также размещается файл исходного проекта - прототипа. После всего этого отчёт в эл. виде сдаётся преподавателю в установленные расписанием сроки, что является основанием для оценивания практических работ по дисциплине как благополучно завершённых.

Качество, полнота, соблюдение учебного графика в указанном виде учебной деятельности, а также выполнение НИРс по тематике дисциплины, могут быть исходной позицией для экзаменатора по дисциплине в части введения в оценку добавочных баллов (до 25 из 100 всего) за «активность» в учебном процессе.

## 02.02. Структура отчёта по практической работе и его титульная часть

Отчёт по практическим работам в электронном виде выполняется в текстовом редакторе, совместимом по формату с Microsoft Word, шрифт текста Times New Roman, размер шрифта 14; междустрочный интервал 1,5 строки, выравнивание текста по ширине, с установленной опцией нумерации страниц, при этом нумерация начинается с титульного листа, первая страница не нумеруется, но в последующей нумерации страниц учитывается. Границы полей страницы: верхнее, нижнее поле – 2-2,5 см, боковое левое поле – не менее 2,5 – 3 см; правое – не менее 1,5 см.

Пример оформления титульного листа приводится ниже. За титульным листом следует подтитульный лист с УДК, выходными данными и аннотацией из трёх фраз (что сделано, как сделано и каков результат).

На этой же странице всё указанное повторяется на английском языке. Страница завершается адресной ссылкой и годом выполнения.

На третьей странице следуют перечень ключевых слов, сокращений, перечень с расшифровкой оценочных индексов и трудовых действий закреплённых за дисциплиной компетенций (актуализированных на конкретный учебный год соответствующими ему Учебным планом).

В случае очной формы работы в аудитории подписывается лист по ТБ и другим мерам безопасности.

Далее следует оглавление, после которого – основной текст отчета.

Указанный текст состоит из краткого постановочного введения, описания работ в порядке их последовательности, каталожного (технического описания) усовершенствованного в частях информационного менеджмента проекта – прототипа (в том числе раскрытия выполнения требований компетенций), обобщающих авторизированных выводов (заключения), сопровождаемого далее списком использованных источников, включая соответствующие стандарты.

В приложениях приводятся скан акта проверки на меру самостоятельности и содержание файла анализируемого проекта – прототипа.

|  |
| --- |
| МИНОБРНАУКИ РОССИИ |
| Федеральное государственное бюджетное образовательное учреждение высшего образования **«МИРЭА − Российский технологический университет»**  **РТУ МИРЭА** |

**Институт информационных технологий (ИИТ)**

**Кафедра инструментального и прикладного программного обеспечения (ИиППО)**

*Пример оформления*

**ОТЧЁТ ПО ПРАКТИЧЕСКИМ РАБОТАМ**

**по дисциплине «Информационный менеджмент систем»**

2021/22 уч.г.

**Наименование проекта-прототипа для проведения анализа и реинжиниринга информационного менеджмента**: «Анализ сообществ социальной сети с целью определения активности подписчиков»

|  |  |  |  |
| --- | --- | --- | --- |
| Отчет представлен к  рассмотрению:  Студент гр. ИКБО-13-19 | «\_\_» \_\_\_\_\_\_\_ 2022 | подпись) | Акимов А.А. |
| Отчёт принят:  Преподаватель каф. ИиППО: | «\_\_» \_\_\_2022 \_\_\_ |  |  |

Отметка сотрудника УВП

О фондировании отчёта: «\_\_\_»\_\_\_\_\_\_2022 \_\_\_\_\_\_\_\_\_\_\_\_\_\_

Москва 2022

## 02.03. Краткое изложение правил оформления и редакторской обработки текста отчёта практической работы

Далее в незначительном сокращении приводятся требования к редакторской обработке текстов выпускных квалификационных работ (ВКР) – именно их, поскольку настоящая дисциплина осваивается в преддверии ВКР и разумнее всего изначально и заблаговременно сформировать устойчивые навыки в этом жанре учебной деятельности.

Итак, предлагается оформлять отчёты по практическим работам согласно требованиям локального нормативного акта РТУ МИРЭА СМКО МИРЭА 7.5.1/03.П.68-18 о правилах оформления и редакторской проработке материалов ВКР.

Указанный в подзаголовке нормативный акт РТУ МИРЭА заявлен как рекомендательный, что, однако, нисколько не снижает уровня требований, предъявляемых к редактуре и оформлению отчётов действующими российскими стандартами, буде то в электронной реализации или на твёрдом, то есть бумажном, носителе, в частности предъявляемых в ЕСКД. Следовательно, основным нормативным источником по вопросам оформления и редактирования отчётов призван быть именно стандарт, к коему и надлежит обращаться авторам (см., в частности, ГОСТ 7.32-2001 «Отчет о научно-исследовательской работе» [14], ГОСТ 2.105-95 «Общие требования к текстовым документам» [15], ГОСТ Р 7.0.12-2011 «Библиографическая запись. Сокращение слов и словосочетаний на русском языке. Общие правила и требования» [16], ГОСТ Р 7.0.5-2008. Библиографическая ссылка. Общие требования и правила составления» [17] и т.п.).

Здесь же приводится некий сокращённый адаптированный вариант (см. Введение к настоящим МУ), отображающий ключевые положения указанного подхода, многолетний опыт кафедры и предупреждающий от достаточно часто встречающихся в отчётах.

Отчёты, как и ВКР в традиционной модели оформляется на листах бумаги стандартного формата А4. Текст размещается на одной стороне листа. Размеры полей: левое – 2 см, правое – 1 см, верхнее – 2 см, нижнее – 2 см. Межстрочный интервал от 1,5 до 1,2. Для основного текста используется шрифт Times New Roman, размер (кегль) 14 пунктов (не исключается 12 пунктов). Для заголовков рекомендуется использовать стиль «Заголовок» кегль 18 или 16 пунктов для автоматического формирования содержания. Абзацный отступ составляет 1.25 см. Абзацы выравниваются по ширине как слева, так и справа\*. Между абзацами не вводятся увеличенные интервалы.

Разумеется, всё это в полной мере относится и к электронным версиям документов, как в реализации Pdf, так и Word, равно как и к текстам в составе слайдов демоверсии к докладу.

(Прим.\*: К сожалению, правила выравнивания строк по ширине студентами нередко нарушаются, несмотря на многочисленные напоминания. Чаще всего нарушается выравнивание справа. Во избежание упомянутого и других эффектов в оформлении текстов ещё до их набора следует установить на компьютере необходимые форматы).

Титульный лист оформляется по установленному образцу. Выпускающая кафедра ИиППО просит также формировать вслед за титульным листом подтитульный лист.

Оглавление работы должно быть автоматическим. Оглавление должно быть не более 2 страниц и иметь не более трех уровней заголовков. Титульный лист, задание, график выполнения и аннотации не приводятся в оглавлении.

При написании аббревиатур, сокращений и собственных имен необходимо учесть следующее.

В тексте работы, кроме общепринятых буквенных аббревиатур, могут быть использованы вводимые лично авторами аббревиатуры, которые при первом обращении к ним по тексту должно быть полностью раскрыто, расшифровано.

Фамилии, названия учреждений, организаций, фирм, название изделий и другие имена собственные в тексте приводят на языке оригинала. Допускается транслитерировать имена собственные и приводить названия организаций в переводе на язык работы с добавлением (при первом упоминании) оригинального названия.

Формулы располагают отдельными строками в центре листа или внутри текстовых строк.

Нумеровать следует только формулы, на которые имеются ссылки в работе. Порядковые номера формул обозначают арабскими цифрами в круглых скобках у правого края страницы.

Обозначения, приведенные латинскими и русскими буквами, пишутся курсивом; остальные символы в формуле (включая греческие буквы, цифры, знаки, скобки) пишутся обычным шрифтом; математические знаки отбиваются пробелом. Непременно приводятся описания обозначений в формуле. В тексте должна быть ссылка на каждую пронумерованную формулу\*. Ссылки в тексте на порядковые номера формул дают в скобках.

(Прим.\*: нередко встречается путаница в нумерациях и обозначениях, отсутствие размерностей, пропуски в нумерации приведённых формулах или пропуски необходимых ссылок на них по тексту. Это не пустяк. В определённых случаях такого рода недостатки искажают суть материала, снижают меру доверия к нему.)

Таблицы и рисунки должны иметь названия и порядковую нумерацию. В тексте работы должны быть обязательно ссылки на них. Нумерация таблиц, рисунков и формул может быть как сквозной для всего текста, так и автономизированной по главам. Разумеется, в последнем из упомянутых вариантов в нумерацию вводится порядковый номер главы, например, конкретная запись вида «Таблица 02.03. Сведения о продаваемости ИТ продукта «Симплекс» за 2015 – 2021 гг.» означает, что названная таблица является третьей по счёту во второй главе. Аналогично дело обстоит с формулами и рисунками. Порядковый номер таблицы проставляется в правом верхнем углу над ее названием. Таблицу с большим количеством строк допускается переносить на другой лист (страницу). При переносе части таблицы на другой лист (страницу) слово «Таблица» и номер ее указывают один раз справа над первой частью таблицы, над другими частями пишут слово «Продолжение» и указывают номер таблицы, например: «Продолжение таблицы 03».

Порядковый номер рисунка и его название проставляются под рисунком, посередине строки. При построении графиков по осям координат вводятся соответствующие показатели, буквенные обозначения которых выносятся на концы координатных осей, фиксируемые стрелками. Там, где это явно необходимо, по осям указываются размерности\*. При необходимости вдоль координатных осей делаются поясняющие надписи. Шрифт подписи рисунка, как и таблицы, может быть на один пункт меньше.

(Прим.\*: достаточно распространённая забывчивость – не приводятся размерности физических величин, без чего рисунок превращается просто в иллюстративную картинку общего плана).

Страницы отчёта следует нумеровать арабскими цифрами, соблюдая сквозную нумерацию по всему тексту. Номер страницы проставляют в центре нижней части листа без точки. Титульный лист включают в общую нумерацию страниц, но номер страницы на титульном листе не ставится. Иллюстрации и таблицы, расположенные на отдельных листах, включают в общую нумерацию страниц работы. Иллюстрации и таблицы на листе формата АЗ учитывают как одну страницу. Для электронной версии отчёта это всё также справедливо.

Разделы отчёта должны иметь порядковые номера в пределах всего документа, обозначенные арабскими цифрами без точки и записанные с абзацного отступа. Подразделы должны иметь нумерацию в пределах каждого раздела. Номер подраздела состоит из номеров раздела и подраздела, разделенных точкой. В конце номера подраздела точка не ставится. Разделы, как и подразделы, могут состоять из одного или нескольких пунктов. Дальнейшее деление ещё большим дроблением не приветствуется. Если раздел не имеет подразделов, то нумерация пунктов в нем должна быть в пределах каждого раздела, и номер пункта должен состоять из номеров раздела и пункта, разделенных точкой. В конце номера пункта точка не ставится.

Внутри подразделов и пунктов могут быть приведены перечисления. Перед каждым перечислением следует ставить дефис или, при необходимости ссылки в тексте документа на одно из перечислений, строчную букву (за исключением ё, з, о, г, ь, и, ы, ъ), после которой ставится скобка.

Список использованных источников представляет собой библиографический список, в который включаются монографическая и статейная научная и учебная литература, материалы диссертаций всех уровней и их авторефераты, законодательные и инструктивные материалы, статистические сборники, идентифицируемые в сетях наименования технологических и программных пакетов и другие отчетные и учетные материалы, Интернет-сайты и порталы.

Сведения об источниках следует располагать в порядке появления ссылок на источники в тексте и нумеровать арабскими цифрами без точки и печатать с абзацного отступа. Возможно также расположение использованных источников в списке в алфавитном порядке (но это несколько сложнее). Ссылка на источник дается на языке указанного источника. Становится распространённым отдельно группировать: ссылки русскоязычные источники и источники на иностранных языках.

При ссылке на авторскую книгу указываются в следующем порядке: фамилия и инициалы автора(-ов), название книги, том (если есть), город, издательство, год, раздел (если требуется). При этом следует придерживать следующих правил:

- между фамилией и инициалами автора — только пробел; инициалы пишутся после фамилии, после каждого из них ставится точка, а между ними нет пробела;

- название книги пишется без кавычек, но с точкой на конце; слово «том» пишется сокращенно «Т.»; после номера тома ставится точка, название тома пишется без кавычек, но с точкой на конце;

- города пишутся полностью, например, Москва и Санкт-Петербург;

- город от издательства отделяется двоеточием;

- название издательства пишется без кавычек и без слова «издательство»;

- после издательства перед годом издания ставится запятая;

- после года слово «год» или сокращение «г.» не пишется, ставится точка;

- при ссылке на конкретные страницы книги пишется «С.», а через пробел диапазон номеров страниц, разделенных дефисом.

Примеры: Сидоров С.А., Огунин В.С. Теория управления в замкнутом энтропийном пространстве. 2-е изд. Москва: Наука, 2019. – 102 с.

При ссылке на источник без авторов (под редакцией или часто встречающееся явление в Интернете) указываются в следующем порядке: название источника, том, раздел (если есть), фамилия и инициалы редакторов (если приведены), город, издательство или учреждение, корпорация, в крайнем случае, только сайт, опубликовавший безымянный материал, год, лучше точная дата обращения в сеть.

При ссылке на статью в журналеуказываются в следующем порядке: фамилия, инициалы автора(-ов), название статьи, название журнала, год, том (volume), номер (number), выпуск (issue), страницы (pages).

Пример: Анничков В.А. Комплексное решение уравнения Дьюрка в задачах оптимизации встречных потоков // Молодой исследователь, 2020. № 2. ‑ С. 22-27.

При ссылке на доклад в сборнике трудов конференции, симпозиума, семинара и т.п. указываются в следующем порядке: фамилия и инициалы автора(-ов), название доклада, название мероприятия, в рамках которого выпущен сборник, тема мероприятия, год, том (если есть), страницы. При этом название мероприятия указываются на месте названия журнала, а тема мероприятия пишется в кавычках после его названия, что сопровождается указанием года проведения мероприятия.

Примеры: Иванов К.А. Применение алгоритма выравнивания весовых вкладов в полиномиальные эмерджентные оценки эффективности многомодульных производительных композиций // Ежегодная научно-техническая конференция студентов, аспирантов и молодых специалистов МХТТ: Тезисы докладов. Москва: МХТТ, 2016. ‑ С.43-49.

Belov A., Slastnikov S. A Metaheuristic Approach for the Problem of Motor Fuel Distribution // 2nd International Conference on Information Technology and Quantitative Management, ITQM 2014, [Volume 31.](http://www.sciencedirect.com/science/journal/18770509/31/supp/C) – Procedia Computer Science, 2014. ‑ P.143-150.

При ссылке на интернет-источникпридерживаются вышеизложенных правил и дополнительно указывают унифицированный указатель интернет-ресурса (URL) и обязательно дату обращения (во избежание недоразумений, связанных с временной ротацией Интернет публикаций). После библиографической записи ставится точка и указывается «[сайт] — URL: ссылка (дата обращения).

Пример:В.Г.Потемкин "Справочник по MATLAB". Графические команды и функции. [сайт] — URL: http://matlab.exponenta.ru/ml/book2/(дата обращения 08.04.2014).

Работая над текстом ВКР и размещая в нём ссылки на источники, следует руководствоваться следующими несложными правилами.

При заимствовании текста цитата приводится в кавычках (без кавычек только в случаях перефразирования в интересах выделения моментов, существенных для выполняемой работы), а после нее или комментарий к ней в квадратных скобках указывается ссылка на литературный источник по текущему списку использованной литературы.

Пример: «Метод Зиберфельда признаётся рядом авторов весьма продуктивным при решении поставленных перед настоящим исследованием задач» [4, 5].

Корректность в выполнении всего здесь указанного в существенной мере характеризует автора как профессионала, подготовленного к самостоятельной научной, инженерной, производственной работе.

## 02.04. Краткое описание выбранного в качестве примера проекта – прототипа из числа ВКР бакалавриата

В качестве примера выбора проекта – прототипа для анализа и последующего улучшения средств информационного менеджмента приводится краткое описание опубликованной в Интернете выпускной квалификационной работы бакалавриата Хиля А.А. на тему: «Исследование метода оптимального выбора средств защиты данных в компьютерных сетях» // Руководитель Мейкшан В.В.(СибГУТИ), Новосибирск, 2019 год – 88 с. (включая приложения). [18]

В представленном к анализу информационного менеджмента проекте -прототипе, в частности указывается, что «на сегодняшний день информационные ресурсы стали одними из самых значимых рычагов экономического влияния. Информация нужного качества в необходимое время, в необходимом месте - это залог достижения целей в абсолютно любом виде деятельности. Монопольные права на обладание информационными ресурсами оказываются, как правило, определяющими преимуществами в борьбе за рынок.

Сегодня невозможно представить организацию, пусть даже и самую мелкую, которая бы не имела в своем распоряжении современные средств обработки данных.

Формирование защищенной системы работы с персональными данными, дает предпосылки для значительного роста эффективности труда, порождает при этом ряд сложных и масштабных проблем. Одна из таких проблем - надежное обеспечение сохранности персональных данных и установленного статуса её использования.

Один из наиболее актуальных на сегодня вопросов, это вопрос организации защиты конфиденциальной информации в рамках действующего российского законодательства, учитывая последние тенденции в законотворчестве регуляторов, направленные на сертификацию всех уровней защиты информации. Особо острым является вопрос обеспечения безопасности информации в масштабных распределенных информационных системах.

Информационная безопасность сегодня развивается очень быстрыми темпами. Данному факту способствуют стремительное развитие информационных технологий, а также противоборство «нападающих» и «защищающих» информацию и информационные коммуникации.

К великому сожалению, указанная динамичность затрудняет обеспечение надежной защищенности информационных ресурсов. На это существуют конкретные причины:

‒ рост быстродействия микросхем, модернизация существующих и создание новых архитектур с высокой степенью параллелизма дает возможность с помощью «грубой силы» (перебором вариантов) более эффективно преодолевать криптографические барьеры, которые ранее были неприступными;

‒ модернизация сетей, рост числа связей в информационных системах, увеличение пропускной способности каналов связи – все это увеличивает число злоумышленников, имеющих технические возможность осуществлять нападение;

‒ рост конкуренции среди производителей ПО вынуждает компании уменьшать сроки разработки программных систем, а это ведет к уменьшению

уровня качества и выпуску продуктов с дефектами в системах защиты.

По завершению выполнения выпускной квалификационной работы получены следующие результаты:

‒ произведен анализ методов и технических средств защиты данных в компьютерных сетях;

‒ произведено исследование современных актуальных угроз информационной безопасности в компьютерных сетях;

‒ дана общая характеристика предприятию и обрабатываемых информационных ресурсов;

‒ произведен анализ применяемых мер защиты информации в исследуемой компьютерной сети;

‒ разработан комплекс программно-аппаратных мер по обеспечению защиты информации в ЛВС;

‒ произведена оценку экономических показателей проекта.

По завершению работы можно достоверно отметить, что все поставленные задачи были выполнены, а цель работы полностью достигнута».

В проекте не присутствует специально выделенный раздел, посвящённый информационному менеджменту, однако, связанные с ним аспекты фрагментально присутствуют в различных решённых задачах и авторских описаниях.

С этой точки зрения проект интересен в качестве прототипа, объекта исследования и внесения в него позиций информационного менеджмента в порядке частичного реинжиниринга.

Можно привести массу такого же рода проектных решений, представляющих интерес с позиций осваиваемой здесь дисциплины.

## 02.05. Технологическая среда и ПО администрирования и модерирования образовательными ресурсами в дистанционном обучении (ДО) и дистанционном информационно-методическом обеспечении работ и процедур, связанных с выполнением практических работ по дисциплине «Информационный менеджмент…»

В контексте дистанционного образования рассматривается программное обеспечение (ПО), удовлетворяющее конкретным задачам, возникающим в процессе дистанционного общения.

Одной из фундаментальных задач является построение устойчивой коммуникации между преподавателем и обучаемым. Для данных целей можно применять как десктопные, так и мобильные решения. По состоянию на период с 2020 по 2021 год ПО, используемое в РТУ МИРЭА для решения подобного рода задач, имеет, как правило, оба варианта исполнения, а иногда и веб-версии.

Для непосредственной коммуникации в режиме телеконференции используются программы Zoom (https://zoom.us/) или Jitsi (https://meet.jit.si/). Существуют как мобильные версии данного ПО, так и веб-версии. Данное ПО обеспечивает стабильную аудио и видео связь в большом коллективе. Присутствует возможность предоставить доступ к рабочему столу и ведение записи встречи (для Zoom этот функционал платный, в бесплатной версии также ограничена длительность конференции).

Для коммуникации внутри рабочих групп, чаще всего используются средства на мобильных платформах, а именно групповые чаты в WhatsApp (https://www.whatsapp.com/) и/или Telegram (https://telegram.org/).

Для проведения лекционных занятий преподаватели могут записывать видео-лекции, с последующим их размещением в соответствующем канале сервиса YouTube (https://www.youtube.com/). Либо воспользоваться платформой Webinar (https://webinar.ru/).

Для проведения практических занятий и лабораторных работ рекомендуется использовать учебные чаты групп (https://lk.mirea.ru/staff/group\_chat/). Возможно так же использование более продвинутого инструмента, портала https://online-edu.mirea.ru/ реализованного с помощью свободно-распространяемого ПО Moodle (https://moodle.org/). Обладая расширенными знаниями по администрированию портала, можно вести практические занятия, выдавать задания по лабораторным работам, принимать на проверку лабораторные и курсовые работы, вести форумы поддержки с различными вариантами уровней доступа (от конкретного студента, группы, до разбиения на магистров, бакалавров, аспирантов).

Для проведения аттестационных мероприятий, в рамках Университета, существует единый портал https://online-edu.mirea.ru/. На нём создаются как промежуточные, так и финальные тесты, к которым подключаются студенты или группы, задаётся время проведения и условия допуска.

Немаловажен вопрос облачного хранения рабочих материалов группы (учебные материалы, файлы данных, устанавливаемое ПО и т.п.). Такой функционал необходим для оперативного доступа к файлам всех членов группы в независимости от их физического нахождения, работы с актуальной информацией, единообразия используемого ПО и т.д. Сервис реализован как в рамках образовательного учреждения на облачной платформе РТУ МИРЭА (https://cloud.mirea.ru/), так и с помощью сторонних сервисов, таких как Яндекс Диск (https://disk.yandex.ru/) или DropBox (https://www.dropbox.com/). Широко применимы сервисы облачного хранения и файл-обмена от компаний Google, Google Drive (https://drive.google.com/). Наряду с функционалом по хранению файлов, они предоставляют возможность коллаборативной работы с документами формата электронных таблиц XLSX и текстовых документов формата DOCX в реальном времени с контролем версионности. Упомянутый функционал заметно упрощает ведение совместной отчётности, статистики, журналов посещаемости, подготовку совместных документов.

В работе тьютора ИТ и ДО по обслуживанию знаниевых ресурсов приходится также часто обращаться к различным расширениям и дополнениям, устанавливаемых на станциях пользователей в части реализаций, избирательного выбора контента, его копирования и «откачки», конвертирования, графических, аудио и видео редакторов и т.п.

Всё вместе составляет единую функциональную библиотеку инструментального багажа (кейса) тьютора по ИТ и ДО.

# Заключение

В качестве заключения к настоящей работе можно отметить, что в инновационном плане её материалы интересны и несколько сложны тем, что в них одновременно синтезированы вопросы, связанные с углублением освоения достаточно новых в практике ВО образовательных стандартов поколения 3++, вовлечением в их сферу в части развития комплексного компетентностного подхода к учебно-творческому процессу и её результатов соответствующих профессиональных стандартов и, одновременно, в силу сложившихся начиная с 2020 года условий, интенсивного и всеобъемлющего дистанционного обучения и взаимодействия, основанных к тому же на форсированном освоении интерактивных методов обучения и творчества.

Безусловно, многие положения настоящей работы [1-19] подлежат ежегодной ротации и дальнейшему развитию, что вполне вписывается в практику высшей школы.

**Контрольные вопросы по всем разделам:**

1. Предмет информационного менеджмента систем.
2. Сферы информационного менеджмента в применении к информационным процессам и системам.
3. Инструменты информационного менеджмента в экспертизах проектов систем по основным признакам качества.
4. Инструменты информационного менеджмента в экспертизах проектов систем по признакам когнитивных, пертинентных и релевантных свойств.
5. Инструменты информационного менеджмента в экспертизах проектов систем по основным системным признакам обрабатываемой и представленной в результате обработки информации.
6. Инструменты информационного менеджмента в экспертизах проектов систем по основным признакам низкой их зашумлённости.
7. Инструменты информационного менеджмента в экспертизах проектов систем по признакам возможных и нежелательных смысловой раздробленности, расслоенности.
8. Методологические основы информационного менеджмента систем
9. Методы и средства обеспечения эффективного информационного менеджмента систем.
10. Формирование информационной технологической среды менеджмента систем по признакам развития архитектуры (архитектоники).
11. Формирование информационной технологической среды менеджмента системы по признакам развития инфологии систем.
12. Формирование информационной технологической среды менеджмента системы по признакам развития морфологии систем.
13. Формирование информационной технологической среды менеджмента системы по признакам усиления средств поддержки менеджмента и обеспечение их обслуживания.
14. Стандартизация информационного менеджмента систем.
15. Информационный менеджмент в информационном обеспечении процессов управления предприятиями и их проектами.
16. Автоматизированные системы управления на вооружении информационного менеджмента систем.
17. Информационный менеджмент – основа безопасности развития организации и системного проекта.
18. Информационный менеджмент образовательных систем, в том числе дистанционного обучения.
19. Формирование информационной технологической среды менеджмента системы по признакам обеспечения жизненного цикла (ЖЦ).
20. Формирование информационной технологической среды менеджмента системы по признакам устойчивости проекта на всём его ЖЦ.
21. Формирование информационной технологической среды менеджмента системы по признакам категорийной оценочности.
22. Формирование информационной технологической среды менеджмента системы по признакам прототипирования и наследственности.
23. Формирование информационной технологической среды менеджмента системы по признакам уровня организационной структуры в области информатизации.
24. Формирование информационной технологической среды менеджмента системы по признакам основных эксплуатационных информационных характеристик систем в интересах становления и развития эффективного информационного менеджмента систем.
25. Формирование информационной технологической среды менеджмента системы по признакам формирования эффективной инновационной политики.
26. Формирование информационной технологической среды менеджмента системы по признакам развития института информационного менеджмента с опорой на современные научные методы, модели и технологии.
27. Формирование информационной технологической среды менеджмента системы по признакам эффективности управления персоналом в границах прерогатив информационного менеджмента систем.
28. Формирование информационной технологической среды менеджмента в частях кадрового обеспечения информационными менеджерами.
29. Описание обобщённых профессиональных функций и траекторий трудовой деятельности информационных менеджеров систем с опорой на реализацию компетентностных требований, соответствующих профильных профессиональных стандартов и инструкций Минтруда РФ по видам профессиональной деятельности (указать на конкретные профстандарты и дать им краткую характеристику в частях, описывающих обобщённые трудовые функции и действия).
30. Описание обобщённых трудовых функций и трудовых действий привлекаемых к информационному менеджменту системных аналитиков согласно положениям соответствующих профессиональных стандартов РФ.
31. Формирование информационной технологической среды менеджмента системы по совокупности признаков эффективности и управления капиталовложениями в сфере информатизации проектов.
32. Характеристика стратегического планирования и информационного менеджмента, а также их альтернативы последних лет.
33. Формирование информационной технологической среды менеджмента системы по признакам формирования и обеспечения комплексной защищённости информационных ресурсов.
34. Формирование информационной технологической среды менеджмента системы по признакам эффективного управления информационными потоками обеспечения.

# Источники

1. Модели разработки и тестирования по: инкрементная модель. [сайт] — URL: https://bytextest.ru/2017/11/23/incremental-model/ (дата обращения: 18.12.2020)
2. Гуков Обзор методов agile. [сайт] — URL: https://vc.ru/u/117584-dmitriy-gukov/135749-obzor-metodov-agile (дата обращения: 18.12.2020)
3. Гибкая методология Scrum. [сайт] — URL: https://kogio.ru/blog/scrum/ (дата обращения: 18.12.2020)
4. Разработка на основе функций. [сайт] — URL: https://intellect.ml/feature-driven-development-5180 (дата обращения: 18.12.2020)
5. Солонин Е.Б. Современные методики разработки информационных систем. [сайт] — URL: https://study.urfu.ru/Aid/Publication/13395/1/ Solonin\_FT.pdf (дата обращения: 18.12.2020)
6. Разработка программного обеспечения. Спиральная модель. [сайт] — URL: http://espressocode.top/software-engineering-spiral-model/ (дата обращения: 18.12.2020)
7. Ираидина М. Знакомимся с Lean: как создать ценность без потерь. [сайт] — URL: https://skillbox.ru/media/management/znakomimsya\_s\_lean/ (дата обращения: 18.12.2020)
8. Проектный менеджмент. Документальный анализ проекта (IEC 61160:2005, Design review, IDT). [сайт] — URL: https://files.stroyinf.ru/ Data/611/61142.pdf (дата обращения: 18.12.2020)
9. Карташев А.В. Генезис каталогизации наукоемкой продукции – Москва: Технополиграфцентр, 2019 – 237 с. [сайт] — URL: http://cals.ru/ sites/default/files/downloads/books/Genezis%20katalogizatsii.pdf (дата обращения 18.12.2020)
10. Каталожное описание продукции (КО). [сайт] — URL: https://normative\_ reference\_dictionary.academic.ru/25240/каталожное\_описание\_продукции (дата обращения 18.12.2020)
11. Становление информационного менеджмента. [сайт] — URL: https://helpiks.org/7-85256.html (дата обращения 18.12.2020)
12. Стандарты в сфере информационного менеджмента. [сайт] — URL: https://helpiks.org/7-85257.html (дата обращения 18.12.2020)
13. ГОСТ Р 43.0.9-2017 Информационное обеспечение техники и операторской деятельности. Информационные ресурсы. [сайт] — URL: http://docs.cntd.ru/document/1200146328 (дата обращения 18.12.2020)
14. ГОСТ 7.32-2001 СИБИД. Отчет о научно-исследовательской работе. Структура и правила оформления. [сайт] — URL: http://docs.cntd.ru/document/1200026224 (дата обращения 18.12.2020)
15. ГОСТ 2.105-95 Единая система конструкторской документации (ЕСКД). Общие требования к текстовым документам (с Изменением N 1, с Поправками). [сайт] — URL: http://docs.cntd.ru/document/1200001260 (дата обращения 18.12.2020)
16. ГОСТ Р 7.0.12-2011 Система стандартов по информации, библиотечному и издательскому делу. Библиографическая запись. Сокращение слов и словосочетаний на русском языке. Общие требования и правила (Переиздание). [сайт] — URL: http://docs.cntd.ru/ document/1200093114 (дата обращения 18.12.2020)
17. ГОСТ Р 7.0.5-2008 СИБИД. Библиографическая ссылка. Общие требования и правила составления (Переиздание). [сайт] — URL: http://docs.cntd.ru/document/1200063713 (дата обращения 18.12.2020)
18. Хиль А.А. Исследование метода оптимального выбора средств защиты данных в компьютерных сетях // Выпускная квалификационная работа бакалавра / Сибирский государственный университет телекоммуника-ций и информатики. - Новосибирск, 2019.- 79 с. [сайт] — URL: https:// sibsutis.ru/upload/631/7\_VKR\_Khil\_A.pdf (дата обращения 15.12.2020)
19. Модели и методологии разработки ПО. [сайт] — URL: https://geekbrains.ru/posts/methodologies (дата обращения 18.12.2020)