

Министерство образования и науки Российской Федерации
Вологодский государственный университет

Кафедра автоматики и вычислительной техники

Управление программными проектами

Методические указания
к выполнению лабораторных занятий
для студентов очной формы обучения

Факультет электроэнергетический

Направления подготовки:

09.03.01 – ИНФОРМАТИКА И ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА

Профиль подготовки: программное обеспечение вычислительной
техники и автоматизированных систем

09.03.04 – ПРОГРАММНАЯ ИНЖЕНЕРИЯ

Профиль подготовки: разработка программно-информационных систем

Вологда
2016

УДК 65.011.56(075.8)

Управление программными проектами: методические указания к выполнению лабораторных занятий для студентов очной формы обучения. – Вологда: ВоГУ, 2016. – 40 с.

Приводится содержание лабораторных занятий и указания по их выполнению, список литературы.

Утверждено редакционно-издательским советом ВоГУ

Составитель: А.М. Полянский, канд. техн. наук, доцент

Рецензент: А.Н. Швецов, д-р техн. наук, профессор кафедры ИСиТ

Подписано в печать 29.02.2016
Печать офсетная

Усл. печ. л. 2,5
Бумага писчая.

Тираж 12 экз.
Заказ № 53.

Отпечатано: РИО ВоГУ, г. Вологда, ул. Орлова, 6

ВВЕДЕНИЕ

Методические указания предназначены для студентов, обучающихся по направлениям подготовки бакалавриата 09.03.01 – ИНФОРМАТИКА И ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА, профиль подготовки: программное обеспечение вычислительной техники и автоматизированных систем и 09.03.04. – ПРОГРАММНАЯ ИНЖЕНЕРИЯ, профиль подготовки: разработка программно-информационных систем, но могут быть полезны студентам любых направлений, которые каким-либо образом связаны с разработкой и применением программно-информационных систем и продуктов.

Методические указания содержат две главы. В первой из них приводится общая характеристика, цели и задачи преподавания дисциплины. Во второй главе рассматривается содержание лабораторных занятий и порядок их выполнения.

1. ОБЩАЯ ХАРАКТЕРИСТИКА ДИСЦИПЛИНЫ

1.1. ЦЕЛИ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

Целями освоения учебной дисциплины «Управление программными проектами» являются:

1. Формирование у обучаемых профессиональных компетенций в области планирования, управления разработкой и применением программных продуктов, проектного менеджмента;
2. Обеспечение способности выпускника к самостоятельной профессиональной деятельности для решения задач предприятия отрасли программной инженерии в рыночных условиях.

1.2. МЕСТО УЧЕБНОЙ ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП ВПО

Дисциплина относится к профессиональному циклу ОПОП ВО, изучается в 8 семестре.

Для освоения данной дисциплины как последующей необходимо изучение следующих дисциплин ОПОП: введение в программную инженерию, математическая логика и теория алгоритмов, экономика, экономика и организация производства, экономика программной инженерии, разработка и анализ требований. Взаимосвязь данной дисциплины с предшествующими отражена в матрице междисциплинарных связей.

Требования к «входным» знаниям, умениям и готовности студента, необходимым при освоении данной дисциплины и приобретенным в результате освоения предшествующих дисциплин, включают следующее:

знать: методику математического анализа; особенности проявления объективных экономических законов в деятельности фирмы; механизмы ценообразования и конкуренции в современных условиях; закономерности формирования издержек производства; особенности функционирования рынков ресурсов и формирования доходов; основы стандартизации продуктов и процессов;

уметь: работать с источниками информации, структурировать информацию, работать с персональным компьютером в объёме офисных приложений; применять ЭВМ к решению практических задач по работе с информацией; разбираться в закономерностях функционирования организационных и экономических систем и тенденциях их развития; использовать на практике инструменты исследования и методы оценки экономических процессов;

владеть: методами поиска и обработки информации; методами математического анализа; навыками владения инструментами исследования и методами оценки организационных и экономических процессов.

Освоение данной дисциплины как предшествующей необходимо при планировании и выполнении выпускной квалификационной работы.

1.3. КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ / ОЖИДАЕМЫЕ РЕЗУЛЬТАТЫ ОБРАЗОВАНИЯ И КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ ПО ЗАВЕРШЕНИИ ОСВОЕНИЯ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ (МОДУЛЯ)

В результате освоения дисциплины студент должен:

знать: состав, порядок формирования и методы оценки эффективности использования ресурсов разработки и сопровождения программного продукта; современные стандарты управления жизненным циклом программных продуктов; показатели и методы ценообразования и оценки эффективности (рентабельности) программного обеспечения; основы проектного менеджмента в программной инженерии; современные методы управления командой проекта; методы организации и планирования процессов разработки и сопровождения программных продуктов; особенности сертификации продуктов и лицензирования деятельности в программной инженерии (ПК-9, 10, 11, 14, 15, 24 для направления 09.03.04 и ОК-2,3,4,6; ПК-1 для направления 09.03.01);

уметь: принимать экономически обоснованные инженерно-технические, организационные и управленческие решения при разработке и сопровождении программных продуктов; применять современные управленческие методы, способствующие повышению эффективности использования привлеченных ресурсов для обеспечения научных исследований, производства и применения программных продуктов; планировать проектную деятельность и распределять ресурсы проекта; проводить экономические расчеты и оценивать трудоёмкость разработки и сопровождения программных продуктов; организовывать коллективное планирование и реализацию проектов; использовать современные программные средства управления проектами (ПК-9, 10, 11, 14, 15, 22, 23, 24 для направления 09.03.04 и ОК-2,3,4,6; ПК-1 для направления 09.03.01);

владеть: методами анализа трудоёмкости, рисков проекта, эффективного управления командой проекта; современными методами и средствами

управления программными проектами (ПК-9, 10, 11, 14, 15, 24 для направления 09.03.04 и ОК-2,3,4,6; ПК-1 для направления 09.03.01).

1.4. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ (МОДУЛЯ)

Общая трудоемкость дисциплины составляет 5 ЗЕТ (180 часов), в том числе в семестрах:

Семестр №	Трудоемкость						РПР, курсовая работа, курсовой проект	Форма промежуточной аттестации	
	Всего		Контактная работа			СРС			Экз.
	ЗЕТ	час.	час.			час.			час.
			Лк	Пр	Лаб.				
8	5	180	24	12	24	84	36	-	зачёт, экзамен

2. СОДЕРЖАНИЕ ЛАБОРАТОРНЫХ ЗАНЯТИЙ И ПОРЯДОК ИХ ВЫПОЛНЕНИЯ

Целью лабораторных занятий является закрепление материала, полученного на лекциях и в ходе самостоятельной проработки теоретических вопросов, а также использование его в дальнейшем применительно к сфере практической деятельности выпускника вуза. В качестве процесса, управляемого проектным методом, рекомендуется использовать разработку и внедрение программного продукта по тематике выпускной квалификационной работы (ВКР).

2.1. Лабораторная работа 1: Разработка VISION по методике RUP для программного продукта, представляемого в ВКР

Видение программного продукта (Vision) предполагает ответ на пять основных вопросов:

- кто будет приобретать и использовать продукт (целевая аудитория);
- какие нужды пользователей продукт удовлетворяет;
- каковы критические характеристики продукта, чтобы удовлетворить эти нужды и сделать продукт успешным;
- чем продукт похож на существующие аналоги и чем он от них отличается;
- какой предполагается срок и бюджет для создания продукта.

Форма документа "Vision" по методике Rational Unified Process (RUP) содержит следующие основные разделы:

- **Введение** - описываются цель документа, его контекст, основные определения, акронимы и сокращения, даются ссылки на другие документы и характеризуется содержание документа.

- Раздел "**позиционирование**" содержит определение решаемой проблемы, указывается целевой заказчик продукта и исследуются преимущества изделия перед аналогами на рынке.
- Описание **совладельцев и пользователей** продукта, демография рынка и репутация разработчика.
- **Краткий обзор** изделия, описание его перспектив и ключевых возможностей в среде применения, предположения и зависимости, оценка стоимости разработки (калькуляция), вопросы лицензирования и инсталляции.

Основные **возможности продукта** в развёрнутом изложении.

- **Ограничения** технические, технологические и прочие обстоятельства, которые необходимо учитывать при создании продукта.
- **Показатели качества**, включая показатели эффективности, надежности, отказоустойчивости и проч.
- **Приоритеты** реализации требований по степени важности, очередности реализации и т.п.
- **Прочие требования к продукту** - применяемые стандарты, системные требования, эксплуатационные требования, требования к окружающей среде.
- **Требования к документации** - содержание руководства пользователя, интерактивной справки, руководства по установке и конфигурированию и прочих документов, сопровождающих продукт.

Приложения могут содержать расчёты и подтверждения характеристик проекта: статус, выгода, объем работ, риск, стабильность, назначение, причина создания продукта и проч.

Задание: Для программного продукта, представляемого в ВКР, составить развёрнутое описание видения продукта. Общий объём документа не должен превышать 4-6 листов.

2.2. Лабораторная работа 2: Разработка и анализ требований к программному продукту ВКР и проекту его разработки и поддержки

Требования - это исходные данные, на основании которых проектируются и создаются программные продукты.

Требования к проекту создания продукта касаются в основном регламентации процесса создания программного продукта и его аудита. В отличие от требований к продукту, требования к проекту имеют более узкий временной горизонт реализации — обычно до сдачи продукта в промышленную эксплуатацию или до завершения гарантийного периода. Регламентация процесса создания продукта позволяет снизить риски проекта, но приводит к дополнительным накладным расходам, поэтому нужен разумный компромисс между этими факторами. В качестве требований к проекту может быть задан регламент отчетов, совместных совещаний по оценке промежуточных результатов проекта, определены характеристики компетенций

участников рабочей группы, их количество, методика управления проектом. **Требования к продукту.** Цель заказчика и пользователей - получить конечный программный продукт как инструмент своей деятельности, функциональный, надёжный и удобный в использовании на всех стадиях его жизненного цикла. Обычно выделяют три уровня требований:

- **бизнес-требования** (business requirements) обычно формулируются топ-менеджерами либо средним звеном управленцев предприятия;
- **требования пользователей** (user requirements) исходят от персонала, который непосредственно будет применять продукт по назначению, и часто плохо структурированы, дублируются, противоречивы;
- **функциональные требования** (functional requirements) формируются обычно экспертным способом на основе анализа аналогов и перспектив применения продукта.

Основным бизнес-требованием к продукту со стороны производителя является его **коммерческая состоятельность** или выгодность производства, т.е. способность получать при производстве продукта и связанных с ним услуг устойчивый доход, гарантирующий не только простое воспроизводство, но и развитие производителя. Основным бизнес-требованием к продукту со стороны потребителя, очевидно, будет **выгодность применения** продукта для целей его жизнедеятельности в сфере собственного бизнеса, частной или общественной жизни, досуга. Наконец, со стороны третьих лиц основным требованием будет, в первую очередь, **безопасность** продукта для их существования.

Полезно сформулировать и **системные требования**, выдвигаемые прикладной программной системой к среде своего функционирования (системной, аппаратной). Пример таких требований - тактовая частота процессора, объём и характеристики памяти, каналы связи, требования к выбору операционной системы, т.е. требования к **программно-аппаратной платформе**.

Функциональные требования записываются, как правило, при посредстве **предписывающих правил**: "система должна позволять оператору формировать приходные и расходные документы". Другим способом являются так называемые варианты использования (users cases). Это - главный, определяющий основные характеристики продукта вид требований.

Нефункциональные требования регламентируют внутренние и внешние условия или атрибуты функционирования системы. Выделяют следующие основные группы нефункциональных требований:

Внешние интерфейсы (External Interfaces), в т.ч. интерфейс пользователя,
Атрибуты качества (Quality Attributes),
Ограничения (Constraints).

Основные **атрибуты качества продукта**:

- Применимость,
- Надёжность,
- Производительность,
- Эксплуатационная пригодность.

Эти атрибуты присутствуют в модели FURPS+ (спецификация **Rational**

Unified Process). Акроним FURPS обозначает следующие категории требований:

- Functionality (Функциональность)
- Usability (Применимость)
- Reliability (Надежность)
- Performance (Производительность)
- Supportability (Сопровождаемость, эксплуатационная пригодность).

Символ "+" расширяет FURPS-модель, добавляя к ней:

- ограничения проекта,
- требования выполнения,
- требования к интерфейсу,
- физические требования.

Ограничения - формулировки условий, модифицирующих требования или наборы требований, сужая выбор возможных решений по их реализации. Выбор платформы реализации и/или развертывания (протоколы, серверы приложений, баз данных, ...), которые, в свою очередь, могут относиться, например, к внешним интерфейсам. Кроме того, в спецификациях RUP выделяются такие категории требований, как:

- требования, указывающие на необходимость согласованности с некоторыми юридическими и нормативными актами;
- требования к лицензированию,
- требования к документированию.

Функциональная спецификация программного продукта состоит из трех частей:

1. Описание **внешней информационной среды**, с которой будет взаимодействовать разрабатываемое программное обеспечение.
2. Определение **функций программного обеспечения**. Вводятся обозначения всех определяемых функций, специфицируются их входные данные и результаты выполнения, с указанием типов данных и ограничений, которым должны удовлетворять эти данные и результаты. Определяется содержание каждой из этих функций.
3. Описание **исключительных ситуаций** и реакций на эти ситуации, которые должен обеспечить продукт.

При формулировке требований полезно учитывать рекомендации следующих отечественных стандартов:

ГОСТ 19.201-78 (ЕСПД. Техническое задание. Требования к содержанию и оформлению). Стандарт устанавливает порядок построения и оформления технического задания на разработку программного продукта независимо от его назначения и области применения.

ГОСТ 34.602-89 (Информационная технология. Комплекс стандартов на автоматизированные системы. Техническое задание на создание автоматизированной системы). Стандарт распространяется на автоматизированные системы (АС) для автоматизации различных видов

деятельности (управление, проектирование, исследование и т. п.), включая их сочетания, и устанавливает состав, содержание, правила оформления документа «Техническое задание на создание (развитие или модернизацию) системы».

Задание:

На основе утвержденной темы и задания на ВКР разработать и проанализировать требования к программному продукту ВКР и проекту его разработки и поддержки в соответствии с приведёнными выше рекомендациями.

2.3. Лабораторная работа 3: Оценка трудоёмкости и сроков разработки проекта по методикам SETIN и COSOMO

Методика COSOMO

В модели COSOMO используется ряд допущений:

- исходный код конечного продукта включает в себя все (кроме комментариев) строки кода;
- начало цикла разработки совпадает с началом разработки продукта, окончание совпадает с окончанием приемочного тестирования, завершающим стадию интеграции и тестирования (работа и время, затрачиваемые на сбор и анализ требований, оцениваются отдельно как дополнительный процент от оценки разработки в целом);
- виды деятельности включают в себя только непосредственно направленные на выполнение проекта работы, в них не входят обычные вспомогательные виды деятельности, такие, как административная поддержка, техническое обеспечение и проч.;
- человеко-месяц состоит из 152 час.;
- проект управляется надлежащим образом, в нем используются стабильные требования.

Уравнения COSOMO для оценки номинальных значений трудоемкости имеют следующий вид [2.1].

$$PM = A \times SIZE^E \times \prod_{i=1}^n EM_i$$

$$A = 2,94$$

$$E = B + 0,01 \times \sum_{j=1}^5 SF_j$$

$$B = 0,91$$

где

SIZE — размер продукта в KSLOC или в количестве функциональных точек без учета поправочных коэффициентов (UFP), определенном по методике 1FPUG, с последующим преобразованием в количество строк кода.

EM_i — (effort multiplier) множители трудоемкости

SF_j — (scale factor) факторы масштаба
 n=7 — для предварительной оценки
 n=17 — для детальной оценки

Для оценки времени реализации проекта (time to develop) применяется следующая формула [2.1]:

$$TDEV = C \times (PM_{NS})^{D+0,2 \times 0,01 \times \sum_{j=1}^S SF_j} \times \frac{SCED}{100},$$

где

$C = 3,67$; $D = 0,28$;

PM_{NS} — (person-months) трудоемкость проекта без учета множителя $SCED$, определяющего сжатие расписания;

$SCED$ — степень сжатия расписания.

Пример 1: Средней сложности проект в 100 тыс.строк. Для него все коэффициенты-мультипликаторы =1,0, E=1,15.

Трудоёмкость составит : $PM = 2,94 \times (100) \times 1,15 = 586,61$ чел.-мес.

$TDEV = 3,67(586,6) \times (0,28 + 0,2 \times (1,15 - 0,91)) = 3,67(586,7) \times 0,328 = 29,7$ мес.

Средняя численность персонала программистов $\Pi = 586,6/29,7 = 19,75$, т. е. 20 чел.

Главной особенностью методики является то, что для оценки трудоемкости необходимо знать размер программного продукта в тысячах строк исходного кода (KSLOC, Kilo Source Lines Of Code). Размер программного продукта может быть, например, оценен экспертами с применением метода PERT. Если анализ продукта выполнен методом функциональных точек, то его размер может быть рассчитан с использованием собственных статистических данных фирмы или с использованием статистики по отрасли (Таблица 3.1) [2.1].

Таблица 3.1

Оценка количества строк, необходимых на реализацию одной не выровненной функциональной точки для некоторых распространенных языков программирования [2.1]

Язык программирования	Оценка количества строк		
	Наиболее вероятная	Оптимистичная	Пессимистичная
Assembler	172	86	320
C	148	9	704
C++	60	29	178
C#	59	51	66
J2EE	61	50	100
JavaScript	56	44	65
PL/SQL	46	14	110
Visual Basic	50	14	276

Факторы масштаба

В методике используются пять факторов масштаба SF_i, которые определяются следующими характеристиками проекта [2.1]:

PREC — прецедентность, наличие опыта аналогичных разработок (Very Low — опыт в продукте и платформе отсутствует; Extra High — продукт и платформа полностью знакомы).

FLEX — гибкость процесса разработки (Very Low — процесс строго детерминирован; Extra High — определены только общие цели).

RESL — архитектура и разрешение рисков (Very Low — риски неизвестны/не проанализированы; Extra High — риски разрешены на 100%).

TEAM — слаботанность команды (Very Low — формальные взаимодействия; Extra High — полное доверие, взаимозаменяемость и взаимопомощь).

PMAT — зрелость процессов (Very Low — CMM Level 1; Extra High — CMM Level 5)

Значения фактора масштаба приведены в Таблице 3.2.

Таблица 3.2

Значение фактора масштаба, в зависимости от оценки его уровня [2.1]

Фактор масштаба	Оценка уровня фактора					
	Very Low	Low	Nominal	High	Very High	Extra High
PREC	6.20	4.96	3.72	2.48	1.24	0.00
FLEX	5.07	4.05	3.04	2.03	1.01	0.00
RESL	7.07	5.65	4.24	2.83	1.41	0.00
TEAM	5.48	4.38	3.29	2.19	1.10	0.00
PMAT	7.80	6.24	4.68	3.12	1.56	0.00

Пример экспоненциального коэффициента масштаба — коэффициент зрелости процессов (PMAT— Process Maturity). Значение коэффициента PMAT зависит в основном от уровня зрелости процессов в соответствии с **моделью СММ**. Процедура определения значения PMAT основана на определении процента соответствия для каждой из 18 основных групп процессов (key process areas — КРА), определенных в СММ.

Множители трудоемкости

Для предварительной оценки трудоёмкости программного проекта необходимо определить уровень **семи множителей трудоемкости** M_i [2.1]:

PERS — квалификация персонала (Extra Low — аналитики и программисты имеют низшую квалификацию, текучесть больше 45%; Extra High — аналитики и программисты имеют высшую квалификацию, текучесть меньше 4%)

RCPX — сложность и надежность продукта (Extra Low — продукт простой, специальных требований по надежности нет, БД маленькая, документация не требуется; Extra High — продукт очень сложный, требования по надежности жесткие, БД сверхбольшая, документация требуется в полном объеме)

RUSE — разработка для повторного использования (Low — не требуется; Extra

High — требуется использование в других продуктах)

PDIF — сложность платформы разработки (Extra Low — специальные ограничения по памяти и быстродействию отсутствуют, платформа стабильна; Extra High — жесткие ограничения по памяти и быстродействию, платформа нестабильна)

PREX — опыт персонала (Extra Low — новое приложение, инструменты и платформа; Extra High — приложение, инструменты и платформа хорошо известны)

FCIL — оборудование (Extra Low — инструменты простейшие, коммуникации затруднены; Extra High — интегрированные средства поддержки жизненного цикла, интерактивные мультимедиа коммуникации)

SCED — сжатие расписания (Very Low — 75% от номинальной длительности; Very High — 160% от номинальной длительности)

Влияние множителей трудоемкости определяется их числовыми значениями, которые представлены в Таблице 3.3.

Таблица 3.3

Значения множителей трудоемкости, в зависимости от оценки их уровня [2.1]

	Оценка уровня множителя трудоемкости						
	Extra Low	Very Low	Low	Nominal	High	Very High	Extra High
PERS	2.12	1.62	1.26	1.00	0.83	0.63	0.5
RCPX	0.49	0.60	0.83	1.00	1.33	1.91	2.72
RUSE	n/a	n/a	0.95	1.00	1.07	1.15	1.24
PDIF	n/a	n/a	0.87	1.00	1.29	1.81	2.61
PREX	1.59	1.33	1.22	1.00	0.87	0.74	0.62
FCIL	1.43	1.30	1.10	1.0	0.87	0.73	0.62
SCED	n/a	1.43	1.14	1.00	1.00	1.00	n/a

Задание 1:

Изучить методическую и нормативную базу методики COCOMO (COCOMO II, Model definition manual. V 2.1. Center for SW Engineering, 2000).

Выполнить расчёт трудоёмкости разработки ПП для ВКР (этап ранней разработки).

Методика CETIN

Казахстанской Ассоциацией ИТ компаний совместно с компанией системных исследований "Фактор" разработана **Методика оценки трудоемкости и стоимости разработки и сопровождения прикладного программного обеспечения при создании информационных систем** (Методика CETIN). Этот подход позволяет оценить стоимость создания продукта на ранних стадиях проектирования.

Методика основана на измерении функционального размера информационной системы на основе **функциональных требований пользователей** к разрабатываемой информационной системе.

Функциональный размер информационной системы измеряется **пятью функциональными единицами** измерения. Методика содержит алгоритмы определения функционального размера информационной системы, расчета трудоемкости и стоимости разработки и сопровождения прикладного программного обеспечения. Методика учитывает также требования качества и технические требования к информационной системе. Методика базируется на методологии RUP фирмы Rational Software. Жизненный цикл разработки ПП согласно RUP состоит из шести основных и трех вспомогательных процессов.

Порядок оценки трудоемкости разработки ПО:

- 1) оценка функционального размера разрабатываемой ИС;
- 2) оценка базовой трудоемкости разработки ПО;
- 3) определение значений поправочных коэффициентов трудоемкости разработки и сопровождения ПО;
- 4) расчет трудоемкости разработки ПО с учетом поправочных коэффициентов;
- 5) оценка срока разработки ПО;
- 6) корректировка трудоемкости разработки ПО при уменьшении срока разработки;
- 7) оценка стоимости разработки ПО.

Функциональный размер ИС представляет собой пять функциональных единиц измерения:

- количество **вариантов использования** – С;
- количество **типов объектов** – Е;
- количество **свойств типов объектов** – Т;
- количество **взаимодействий между** типами объектов – I;
- количество **типов узлов** – N.

Функциональный размер обозначается как $SIZE = \{C, E, T, I, N\}$. Функциональный размер, записанный в виде $SIZE = \{12, 26, 134, 102, 4\}$ означает, что модель ИС имеет следующие значения функциональных единиц измерения:

- 1) 12 вариантов использования;
- 2) 26 типов объектов;
- 3) 134 свойств типов объектов;
- 4) 102 взаимоотношения между типами объектов;
- 5) 4 типа узла ИС.

Функциональный размер ПО ИС определяется путем подсчета значений функциональных единиц измерения для модели ИС. Входными документами для оценки функционального размера ППО ИС могут служить:

- 1) видение ИС;
- 2) концепция ИС;
- 3) техническое задание на разработку ИС.

Для применения методики расчета строят следующие диаграммы:

- 1) диаграмма вариантов использования (Use case diagram);
- 2) диаграмма классов (Class diagram);

- 3) диаграмма коммуникаций (Communication diagram);
- 4) диаграмма развертывания (Deployment diagram).

После построения диаграмм расчет выполняется следующим образом:

этап 1 - количество вариантов использования (C) определяется из диаграммы вариантов использования модели информационной системы;

этап 2 - количество типов объектов (E) оценивается подсчетом количества **неодинаковых классов**, изображенных на диаграммах классов;

этап 3 - количество свойств типов объектов (T) оценивается подсчетом количества свойств классов, изображенных на диаграммах классов;

этап 4 - количество взаимодействий между типами объектов (I) оценивается подсчетом количества связей (отношений) между классами на диаграмме коммуникаций (в первых версиях языка — диаграмма кооперации, collaboration diagram);

этап 5 - количество типов узлов (N) оценивается подсчетом количества **типов узлов** на диаграмме развертывания.

В случае отсутствия модели ИС оценщику предлагается заполнить **анкету**, представляющую собой перечень вопросов об информационной системе. Такой подход позволит определить функциональность системы на ранних этапах, оценить границы системы. В методике представлены алгоритмы определения функционального размера.

Базовая трудоемкость S_j процесса разработки с номером j рассчитывается по следующей формуле:

$$S_j = 1/165 \cdot [C \cdot S_j(C) + E \cdot S_j(E) + T \cdot S_j(T) + I \cdot S_j(I) + N \cdot S_j(N)],$$

где:

S_j - трудоемкость процесса разработки с номером j в [человеко-месяц];
 j - номер процесса разработки (значения от 1 до 6 по методике RUP);

$S_j(C)$ - норматив трудоемкости реализации одного варианта использования в процессе разработки с номером $j=1,2,...,6$, {[человеко-час]/[вариант использования]};

$S_j(E)$ - норматив трудоемкости реализации одного типа объектов в процессе разработки с номером $j=1,2,...,6$. {[человеко-час]/[тип объектов]};

$S_j(T)$ - норматив трудоемкости реализации одного свойства типа объекта в процессе разработки с номером $j=1,2,...,6$. {[человеко-час]/[свойство типа объектов]};

$S_j(I)$ - норматив трудоемкости реализации одного взаимодействия между типами объектов в процессе разработки с номером $j=1,2,...,6$. {[человеко-час]/[взаимодействие между типами объектов]};

$S_j(N)$ - норматив трудоемкости реализации одного типа узла в процессе разработки с номером $j=1,2,...,6$. {[человеко-час]/[узел]};

165 - количество человеко-часов в одном человеко-месяце;

{C,E,T,I,N} - функциональный размер ИС, определенный в параграфе 5 главы 2 Расчета, в функциональных единицах измерения.

Расчёт ведётся для шести основных процессов RUP: **бизнес-**

моделирование, управление требованиями, анализ и проектирование, реализация, тестирование, развёртывание.

Согласно выявленным характеристикам системы и внешним факторам по отношению к системе определяются значения **поправочных коэффициентов**. Поправочные коэффициенты основаны на применяемых стандартах разработки ПП. На основании поправочных коэффициентов делается **корректировка базовой трудоемкости** и расчет трудоемкости разработки ИС.

На основании трудоемкости делается оценка **срока разработки** проекта, которая основывается на математической модели зависимости срока разработки от состава команды.

Оценка стоимости разработки ПО рассчитывается как произведение трудоемкости разработки на ПО на стоимость одного человеко-месяца с учётом накладных расходов и налоговых отчислений.

Определение значений поправочных коэффициентов трудоемкости.

Оценка базовой трудоемкости создания ППО ИС определяет трудоемкость реализации функциональных требований пользователя, которые **не включают в себя технические требования к системе и требования качества пользователя**. Влияние этих требований в Расчете учитывается через поправочные коэффициенты. Поправочные коэффициенты трудоемкости процессов разработки ППО ИС определяются и рассчитываются через частные поправочные коэффициенты разработки и сопровождения ППО:

$$КП1=K11 \cdot K16 \cdot K17; (2)$$

$$КП2=K1 \cdot K2 \cdot K4 \cdot K5 \cdot K6 \cdot K7 \cdot K8 \cdot K9 \cdot K16 \cdot K17 \cdot K18; (3)$$

$$КП3=K1 \cdot K2 \cdot K4 \cdot K5 \cdot K6 \cdot K7 \cdot K8 \cdot K9 \cdot K11 \cdot K12 \cdot K13 \cdot K14 \cdot K15 \cdot K16 \cdot K17 \cdot K18; (4)$$

$$КП4 = K1 \cdot K2 \cdot K4 \cdot K5 \cdot K6 \cdot K7 \cdot K8 \cdot K9 \cdot K10 \cdot K12 \cdot K13 \cdot K14 \cdot K15 \cdot K16 \cdot K17 \cdot K18; (5)$$

$$КП5=K1 \cdot K2 \cdot K4 \cdot K5 \cdot K6 \cdot K7 \cdot K8 \cdot K9 \cdot K10 \cdot K11 \cdot K12 \cdot K13 \cdot K14 \cdot K15 \cdot K16 \cdot K17 \cdot K18; (6)$$

$$КП6=K1 \cdot K2 \cdot K11 \cdot K16 \cdot K18. (7)$$

Частные поправочные коэффициенты трудоемкости разработки и сопровождения ППО приведены в Приложении 3 к Нормативам.

Все **частные поправочные коэффициенты** разработки и сопровождения ППО являются безразмерными величинами и сгруппированы в три группы в зависимости от типов влияющих факторов:

- 1) внутренние факторы;
- 2) факторы среды;
- 3) факторы данных.

Каждая группа состоит из соответствующих факторов, влияющих на трудоемкость разработки, а каждый фактор из возможных значений фактора.

На основании поправочных коэффициентов трудоемкости разработки ППО ИС делается расчет трудоемкости создания ППО ИС с учетом поправочных коэффициентов по следующей формуле (8):

$$S=КП1 \cdot S1+КП2 \cdot S2+КП3 \cdot S3+КП4 \cdot S4+КП5 \cdot S5+КП6 \cdot S6, (8)$$

где S - скорректированная трудоемкость процесса разработки ППО в человеко-

месяцах;

S_j - базовая трудоемкость процесса разработки с номером j в человеко-месяцах;

КП $_j$ - поправочный коэффициент трудоемкости процесса разработки с номером j .

Задание 2:

Изучить методическую и нормативную базу методики CETIN (Приказ и.о. Министра транспорта и коммуникаций Республики Казахстан от 25 октября 2012 года № 722. «Об утверждении нормативов затрат на создание, развитие и сопровождение информационных систем»).

Выполнить расчёт трудоёмкости разработки ПП для ВКР.

2.4. Лабораторная работа 4: Планирование закупок программного проекта

Планирование и организация закупок и поставок — важный этап в управлении ресурсами проекта. Планирование и организация осуществляются на основе данных о необходимых для успешного выполнения проекта ***внешних ресурсах*** в увязке с общим планом проекта и учитывают ***длительность цикла закупок и доставки*** грузов. План закупки состоит из этапов, включающих выбор поставщиков, размещение заказов, контроль за поставками и ряд других.

Для программного проекта по тематике ВКР следует определить те ресурсы, которые нужно приобрести для успешной разработки проекта, а в случае охвата жизненным циклом проекта и стадии внедрения продукта у приобретающей стороны — ресурсы, необходимые для функционирования продукта в штатном режиме.

В общем случае для стадии проектирования к числу таких ресурсов нужно отнести: АРМ разработчика со средствами разработки, моделирования, тестирования и документирования продукта; затраты на инфраструктуру поддержки разработки (обеспечение помещения, электроснабжения и защиты, доступ в интернет и т.п.); расходные материалы; внешние информационные ресурсы; независимая экспертиза (рецензент) и проч.

Для стадии внедрения к числу таких ресурсов нужно отнести: программно-аппаратную платформу — среду функционирования продукта; затраты на инфраструктуру поддержки функционирования (обеспечение помещения, электроснабжения и защиты, доступ в интернет и т.п.); расходные материалы на период опытной эксплуатации; внешние информационные ресурсы; подготовку персонала, затраты на ввод/конвертирование необходимой постоянной и переменной информации и проч.

Выбор поставщиков осуществляется на основе изучения квалификационных анкет, отражающих управленческие, технические, производственные и финансовые возможности; список претендентов согласуется с заказчиком и руководителем проекта; окончательный выбор поставщиков осуществляется в результате конкурсных процедур.

Размещение заказов — совместно с проектной организацией

разрабатываются мероприятия по стандартизации закупок; общие заказы оформляются на основе работ по сокращению номенклатуры закупок; оценка заявок и проведение торгов предшествуют заключению контрактов, которое производится в результате дополнительных согласований с победителями торгов по требованиям к перевозке и хранению грузов, порядку платежей и передачи активов.

Контроль за поставками — осуществляется на основе специальных графиков; организуется по каждому из видов поставок; основывается на общем плане проекта; все изменения вносятся в общий график проекта; основывается на стандартных формах отчетности.

Информация по закупкам должна описывать заказанную продукцию, включая, где это необходимо, требования:

- а) к официальному одобрению продукции, процедур, процессов и оборудования;
- б) к квалификации персонала;
- с) к системе менеджмента качества.

Организация должна обеспечивать достаточность установленных требований к закупкам до их сообщения поставщику.

Верификация закупленной продукции

Организация должна разработать и осуществлять контроль или другую деятельность, необходимую для обеспечения соответствия закупленной продукции установленным требованиям к закупкам. Если организация или ее потребитель предполагает осуществить верификацию у поставщика, то организация должна установить меры по верификации и порядок выпуска продукции в информации по закупкам. Для планирования закупок полезно использовать модель процесса приобретения в соответствии с ГОСТ Р ИСО/МЭК 12207-2010.

Задание:

На основе сметы затрат по проекту спланировать закупки необходимых материальных и информационных ресурсов, отразить закупочные работы в ИСР.

Представить график закупок в табличной форме в привязке к календарю проекта.

2.5. Лабораторная работа 5: Ролевая модель команды

Роли и ответственности участников проекта разработки ПО принято делить на **пять основных групп** [2.1]:

- **Анализ.** Извлечение, документирование и сопровождение требований к продукту.
- **Управление.** Определение и управление производственными процессами.
- **Производство.** Проектирование и разработка ПО.
- **Тестирование.** Тестирование ПО.
- **Обеспечение.** Производство дополнительных продуктов и услуг.

Группа анализа включает в себя следующие основные роли:

- Бизнес-аналитик. Строит модели предметной области (онтологии).
- Бизнес-архитектор. Разрабатывает бизнес-концепцию системы. Определяет видение продукта, его интерфейсы, поведение и ограничения.
- Системный аналитик. Отвечает за перевод требований к продукту в функциональные требования к ПО.
- Специалист по требованиям. Документирует и сопровождает требования к продукту.
- Менеджер продукта (функциональный заказчик). Представляет в проекте интересы пользователей продукта.

Группа управления состоит из следующих ролей:

- Руководитель проекта. Отвечает за достижение целей проекта при заданных ограничениях (по срокам, бюджету и содержанию), управляет операциями проекта и выделенными ресурсами.
- Куратор проекта. Оценивает планы и ход исполнения проекта. Выделяет ресурсы.
- Системный архитектор. Разрабатывает техническую концепцию системы. Готовит ключевые проектные решения по внутреннему устройству программного продукта и его технических интерфейсов.
- Администратор проекта. Отвечает за информационное обеспечение руководителя проекта, организацию и ведение документооборота по проекту.
- Руководитель группы тестирования. Определяет цели и стратегию тестирования, управляет тестированием.
- Ответственный за управление изменениями, конфигурациями, за сборку и поставку программного продукта.

В **производственную группу** входят:

- Проектировщик. Проектирует компоненты и подсистемы в соответствии с общей архитектурой, разрабатывает архитектурно значимые модули.
- Проектировщик базы данных.
- Проектировщик интерфейса пользователя.
- Разработчик. Проектирует, реализует и отлаживает отдельные программные модули.

В большом проекте может быть несколько производственных групп, ответственных за отдельные подсистемы. Как правило, руководитель проекта делегирует свои полномочия ключевым исполнителям, но общая ответственность за ход и результаты проекта остаётся за ним.

Группа тестирования в проекте состоит из следующих ролей:

- Проектировщик тестов. Разрабатывает тестовые сценарии.
- Разработчик автоматизированных тестов. Проектирует, реализует и отлаживает тестовые программы.
- Тестировщик. Испытывает продукт. Анализирует и документирует результаты.

Участники **группы обеспечения** иногда не входят в команду проекта, если есть определённые **гарантии качества и сроков** выполняемых ими работ. Они

выполняют работы по модели процессной деятельности. К группе обеспечения можно отнести следующие проектные роли:

- Технический писатель.
- Переводчик.
- Дизайнер графического интерфейса.
- Разработчик учебных курсов, тренер.
- Участник рецензирования.
- Продажи и маркетинг.
- Системный администратор.
- Технолог.
- Специалист по инструментальным средствам.
- Другие.

В зависимости от масштаба проекта одну роль могут исполнять несколько человек. Например, разработчики, тестировщики, технические писатели. Некоторые роли должен исполнять *только один человек*, по крайней мере, как главный специалист в проекте. Например, Руководитель проекта, Системный архитектор. Возможны также совмещения ролей, например:

- Руководитель проекта + системный аналитик (+ системный архитектор)
- Системный архитектор + разработчик
- Системный аналитик + проектировщик тестов (+ технический писатель)
- Системный аналитик + проектировщик интерфейса пользователя
- Ответственный за управление конфигурациями + ответственный за сборку и поставку (+ разработчик)

Не рекомендуется совмещать в одном проекте следующие роли:

- Разработчик + руководитель проекта
- Разработчик + системный аналитик.
- Разработчик + проектировщик интерфейсов пользователя.
- Разработчик + тестировщик

В ходе дипломного проектирования в вузе обычно реализуется схема индивидуальной работы над проектом одного студента. Для такого случая рекомендации по ограничению объединения ролей в одном лице могут быть реализованы только в режиме разделения времени: несовместимые роли реализуются студентом в разные периоды времени, переход от одной роли к другой фиксируется отчётом о выполнении работ по предыдущей роли.

Роль руководителя проекта исполняет закреплённый за студентом преподаватель, он может также частично исполнять роли функционального заказчика, системного аналитика и системного архитектора. Руководитель проекта может частично делегировать свои полномочия студенту.

Роль куратора проекта принадлежит заведующему кафедрой или специально уполномоченному им лицу, при описании модели команды можно не учитывать, считая внешней ролью.

Внешними по отношению к проекту являются также роли нормо-

контролёра, рецензента и аттестационной комиссии.

Задание:

По разработанной на предыдущих занятиях иерархической структуре работ и результатам расчёта трудоёмкости проектных работ выполнить распределение работ по ролям проектной команды. Учесть при совмещении ролей принцип временного разделения конкурирующих ролей и работ. Исходя из квалификационных требований к тем или иным ролям проекта и часовых ставок оплаты труда, определить стоимость работ, ролей и общий фонд оплаты труда по проекту. Распределение работ по ролям представить в табличном процессоре.

Минимальный набор ролей, требующий распределения и привязки к ИСР:

- Руководитель проекта.
- Системный архитектор.
- Администратор проекта.
- Бизнес-аналитик.
- Бизнес-архитектор.
- Системный аналитик.
- Специалист по требованиям.
- Менеджер продукта (функциональный заказчик).
- Проектировщик.
- Разработчик.
- Проектировщик тестов.
- Разработчик тестов.
- Тестировщик.
- Технический писатель.

Примерный размер часовых ставок оплаты труда для расчёта, руб./час.:

Руководитель проекта, Бизнес-архитектор, Системный архитектор, Менеджер продукта	200
Бизнес-аналитик, Системный аналитик, Проектировщик, Разработчик, Проектировщик тестов	170
Разработчик тестов, Технический писатель, Специалист по требованиям	150
Тестировщик, Администратор проекта	120

При определении стоимости роли учесть, что назначение низко квалифицированной работы высококвалифицированному исполнителю не должно сопровождаться снижением принятой для него часовой ставки оплаты труда.

2.6. Лабораторная работа 6: План коммуникаций проекта и план управления конфигурациями

2.6.1. План коммуникаций, т.е. обмена информацией между участниками является составной частью плана проекта. Он включает в себя:

- план сбора информации (источники информации и методы ее получения);
- план распределения информации (потребители информации и способы ее доставки);
- детальное описание каждого документа, который должен быть получен или передан, включая формат, содержание, уровень детальности и используемые определения;
- план ввода в действие выбранных видов коммуникаций;
- методы обновления и совершенствования плана коммуникаций.

Задание 1:

Составить план коммуникаций ВКР по следующему шаблону:

- план сбора информации для ВКР с указанием источников информации и методов ее получения;
- план распределения информации с указанием потребителей информации и способов ее доставки;
- примерный перечень документов, которые должны быть получены или переданы, включая краткую характеристику содержания;
- план ввода в действие тех или иных видов коммуникаций (очные консультации, электронная почта и проч.);
- поводы и методы обновления и совершенствования плана коммуникаций.

2.6.2. План управления конфигурацией

Поскольку в любом сложном проекте результат (продукт) бывает представлен в виде нескольких взаимосвязанных, но достаточно самостоятельных объектов (программные модули, техническая и эксплуатационная документация, обучающие материалы, презентации и проч.), и объекты эти периодически изменяются (по мере разработки, сопровождения и т. п.), возникает задача создания и поддержки **согласованного описания** всех версий этих компонент, взаимоувязанных друг с другом, обеспечивающего **целостность результата** (продукта). Важно иметь возможность по конфигурации осуществить сборку целостного, непротиворечивого и актуального на каждый момент времени продукта.

План управления конфигурацией проекта (продукта) обычно содержит следующие основные разделы:

а). Введение. Обзор содержания документа. Включает цели, область действия, определения, акронимы, сокращения, ссылки и обзор плана конфигурационного управления.

б). Конфигурационное управление программным продуктом. Описывает все технические и технологические аспекты управления конфигурацией в проекте.

Организация, распределение ответственности и взаимодействия.

Указывается, кто будет ответственным за выполнение различных задач конфигурационного управления. Дополнение - подраздел, описывающий политику доступа. Обычно для этого выбирают способ описания либо только доступных операций, либо только запрещенных.

Инструментарий, рабочая среда и инфраструктура. Рассматриваются рабочая среда и программное обеспечение, которое будет использовано при выполнении функций конфигурационного управления в ходе жизненного цикла проекта или программного продукта. Описываются инструменты и процедуры, которые нужно использовать для версионного контроля объектов конфигурационного управления, создаваемых в ходе жизненного цикла проекта или программного продукта.

Поскольку в команде проекта минимум два участника: руководитель и студент, важно решить проблему ***синхронизации версий*** при обмене создаваемыми документами проекта. Часто бывает, что разделы ВКР, отправленные преподавателю для промежуточного контроля, подвергаются параллельной несогласованной правке обоими, и в результате собрать из них целостный результат бывает проблематично. Другая актуальная проблема — различия в офисных продуктах или их версиях у членов команды проекта.

в). Программа конфигурационного управления

Конфигурационная идентификация

Описывается, как именуются, маркируются и нумеруются артефакты проекта или программного продукта. Схема идентификации должна включать оборудование, системное программное обеспечение, продукты внешних разработчиков и все артефакты разрабатываемого приложения, указанные в структуре директорий программного продукта; например, модели, планы, компоненты, тестовое ПО, результаты и данные, исполняемые файлы и т.д. Очень важно описать ***все правила именования*** объектов УК. Например, презентация к ВКР представляется в виде файла с именем ***Презентация_ВКР_Иванов_версия_08_05_2015_01.ppt***, генерируемого в среде MS OfficeXP rus. Различие между версиями презентации определяется датой создания и порядковым номером в пределах текущих суток, в явном виде прописанными в имени файла. Должна быть детально расписана ***структура каталогов проекта***.

Базовые версии проекта - Project Baselines. Базовые версии предоставляют официальную версию продукта, на котором основывается последующая работа и для которого проводятся только авторизованные изменения. Определяются точки жизненного цикла проекта или продукта, где должны создаваться базовые версии. Базовые версии должны быть в конце каждой из фаз обследования, проработки проекта, построения системы и передачи в эксплуатацию. Базовые версии также могут создаваться в конце итераций внутри различных фаз. Определяется, кто может создавать базовые версии (обычно это студент, но может быть и по-другому) и что входит в их состав. Здесь описывается то, каким образом будет происходить сама работа со средством управления конфигурациями: как будут ставиться метки, как выпускаться релизы, сколько

ветвей для реализации проекта будет использовано и по какому принципу ветви будут именоваться.

Обычно в ВКР ветвь одна, а базовые линии продукта начинаются с концепции продукта, далее результаты первого этапа разработки, рабочей документации, завершения тестирования, завершения согласований и рецензирования, завершения внедрения. При этом не все объекты конфигурации разрабатываются в полном объеме и доводятся до внедрения.

Контроль конфигураций и изменений. Процесс управления состоит из двух частей – *управление изменениями и управление версиями*. Управлять необходимо любыми изменениями: от заявок пользователей до исправляемых дефектов. Раздел содержит полное описание всех запросов на изменения, включая атрибуты продукта и жизненный цикл. В ВКР это уведомления при обмене результатами проектирования между основными ролями команды проекта.

Отработка и утверждение запросов на изменение. Рассматриваются процессы, которые обеспечивают внесение, рассмотрение и упорядочение проблем и изменений. Определяются *типы запросов*: дефект, запрос на расширение, задача и заявка.

Группа управления изменениями. Описывается, кто входит в состав группы управления изменениями, и процедуры, которым она следует, для отработки и утверждения запросов на изменение. Для ВКР в группу входят руководитель проекта и студент.

Учет состояния конфигурации.

Хранение материалов проекта и выпуск релизов. Описываются *правила хранения и регламенты резервирования* объектов проекта, действия на случай непредвиденных обстоятельств. Описание процесса выпуска релизов включает их содержание, для кого они предназначены и имеются ли какие-либо известные проблемы и инструкции по инсталляции (можно вынести в отдельное приложение).

Отчеты и проверки. Рассматривается содержание, формат и цель запрашиваемых отчетов и проверок состояния конфигурации. Отчеты используются для получения *данных о качестве* программного продукта в любой заданный момент времени жизненного цикла программного продукта или проекта. Отчетность по дефектам, основанная на запросах на изменения, может обеспечить некоторые индикаторы качества. По отчетам можно проследить ход выполнения работ. Необходимо определить отчеты по ролям участников проекта и описать их формат. Также рекомендуется сформировать регламент сбора отчета, то есть, с какой периодичностью собираются метрики (в реальном времени, раз в день... и т. д.). Желательно выделить различные типы отчетов и периодичность сбора их метрик.

Документирование конфигурации. Раздел определяет способы создания и типы документов.

Описание версии. Данный документ описывает диски, CD или другие носители, используемые для поставки ПО. Раздел также определяет состав документов, поставляемых с версией ПО и доступных для конечных пользователей.

Примерный состав документов:

- архив релизов с описанием (Release Media);
- описание релиза (Release Notes);
- описание функций;
- перечень решенных проблем в релизе;
- перечень новых возможностей;
- инструкция по установке ПО;
- инвентаризация, опись.

г). Этапы.

Детально рассматриваются этапы работ для заказчика и внутренние, относящиеся к работам по УК для программного продукта или проекта. Эта секция обычно включает детальное описание того, когда может быть модифицирован сам план конфигурационного управления. В зависимости от выбранной модели может измениться содержание этапов. Рекомендуется описать, что выполняется в УК в зависимости от этапа проекта.

д). Обучение и ресурсы.

Рассматриваются инструментальные средства, персонал и обучение, требуемые для реализации описанных в плане задач.

е). Субподрядчики и контроль программного обеспечения со стороны поставщиков.

Описывается, как будет интегрировано программное обеспечение, разработанное вне среды УК проекта. К работе над проектом могут привлекаться субподрядчики. Данный раздел описывает, каким образом будет происходить работа с субподрядчиком. Для ВКР раздел актуален, если разрабатываются компоненты СПО, или ВКР является взаимосвязанной частью с другими частями единого проекта.

ж). Приложения

Состав приложений к Плану не определяется стандартами. Обычно включает в себя такие документы, как:

- регламенты;
- инструкции по использованию средств УК (как пользовательские, так и административные);
- различные методические пособия;
- планы обучения;
- инструкции по установке и администрированию средств УК и т. д.

Для ВКР в качестве приложения к плану должны быть, как минимум, указаны методические рекомендации по дипломному проектированию, перечень элементов конфигурации, схемы идентификации для элементов и их версий, правила именования объектов и информация по автоматизированным средствам УК.

Задание 2: на основе устава и описания содержания проекта установить структуру и организацию элементов, образующих конфигурацию

разрабатываемого в ВКР программного продукта, установить **схемы идентификации** для элементов и их версий, а также задать инструменты и описать технику, которая будет использоваться для управления этими элементами. Оформить в виде Плана управления конфигурацией продукта, разрабатываемого в ВКР.

2.7. Лабораторная работа 7: Анализ и планирование рисков проекта ВКР

Риск проекта - это суммарный эффект вероятностей наступления неопределенных событий, способных оказать отрицательное или положительное влияние на цели проекта. Риски подразделяются на известные и неизвестные. Известные риски идентифицируются и подлежат управлению - создаются планы реагирования на риски и резервы на возможные потери. Неизвестные риски нельзя определить, и, следовательно, невозможно спланировать действия по реагированию на такой риск. Поэтому создают определённый **резерв ресурсов** для реагирования на них.

Событие риска - потенциально возможное событие, которое может нанести ущерб или принести выгоды проекту.

Вероятность возникновения риска. Все риски имеют вероятность больше нуля и меньше 100%. Риск с вероятностью 0 не может произойти и не считается риском. Риск с вероятностью 100% также не является риском, поскольку это достоверное событие, которое должно быть предусмотрено планом проекта. Последствия риска, если он случится, выражаются через дни расписания, трудозатраты, деньги и определяют степень воздействия на цели проекта.

Величина риска - показатель, объединяющий вероятность возникновения риска и его последствия. Величина риска рассчитывается путем умножения вероятности возникновения риска на соответствующие последствия.

План управления рисками - документ, разрабатываемый в начале проекта и содержащий описания структуры управления рисками проекта и порядок его выполнения в рамках проекта; включается в состав плана управления проектом.

План управления рисками обычно содержит следующие элементы:

- а). Методика** - определение подходов, инструментов и источников данных, которые могут использоваться для управления рисками в данном проекте.
- б). Распределение ролей и ответственности** - список позиций выполнения, поддержки и управления рисками для каждого вида операций, включенных в план управления рисками, назначение сотрудников на эти позиции и разъяснение их ответственности.
- в).** Определение **операций по управлению рисками**, которые необходимо включить в расписание проекта.
- г).** Определение **сроков и частоты выполнения операций** по управлению рисками на протяжении всего жизненного цикла проекта.
- д).** Выделение **ресурсов и оценка стоимости** мероприятий, необходимых для управления рисками. Эти данные включаются в **базовый план по стоимости проекта**.
- е). Классификация рисков** (или категории рисков) - структура, на основании

которой производится систематическая и всесторонняя **идентификация рисков** с нужной степенью детализации.

ж). Определение **вероятности возникновения рисков и их последствий**.

Общие определения уровней вероятности и уровней воздействия адаптируются отдельно для каждого проекта в ходе процесса планирования управления рисками и используются в процессе качественного анализа рисков. Можно применить относительную шкалу, на которой вероятность обозначена описательно, со значениями от "крайне маловероятно" до "почти наверное", или шкалу, на которой вероятности соответствует цифровое значение, например: 0,1 - 0,3 - 0,5 - 0,7 — 0,9.

Матрица вероятности и последствий (Табл. 7.1) содержит комбинации вероятности и воздействия, при помощи которых рискам присваивается определенный **ранг**: низкий, средний или высший. Матрица может содержать описательные термины или цифровые обозначения и строится на основании **шкал оценки вероятности и оценки степени влияния** возможного риска. Левый столбец матрицы содержит значения вероятности возникновения риска, в первой строке расположена шкала со значениями возможных последствий. Ячейки заполняются результатами перемножения значений этих шкал.

Таблица 7.1

Матрица вероятности и последствий [2.1]

Вероятность	Угрозы					Благоприятные возможности				
0,9	0,05	0,09	0,18	0,36	0,72	0,72	0,36	0,18	0,09	0,05
0,7	0,04	0,07	0,14	0,2	0,56	0,56	0,28	0,14	0,07	0,04
0,5	0,03	0,05	0,1	0,2	0,4	0,4	0,2	0,1	0,05	0,03
0,3	0,02	0,03	0,06	0,12	0,24	0,24	0,12	0,06	0,03	0,02
0,1	0,01	0,01	0,02	0,04	0,08	0,08	0,04	0,02	0,01	0,01
	0,05	0,1	0,2	0,4	0,8	0,8	0,4	0,2	0,1	0,05

Сопоставляя значение ячейки матрицы со шкалой оценки воздействия, риски можно разделить по категориям - **малые, средние и большие**. Риски, имеющие очень высокие вероятности, но незначительные последствия, а также риски, имеющие низкие вероятности и незначительные последствия, считаются рисками, не оказывающими воздействия (клетки таблицы серого цвета). Риски с очень большими последствиями, но малой вероятностью, как и риски с незначительными последствиями и высокой вероятностью (клетки белого цвета) имеют среднее воздействие на проект. Риски, которым необходимо уделять особое внимание, имеют достаточно высокую вероятность и существенные последствия (клетки таблицы, окрашенные темно-серым цветом).

з). **Уточненная толерантность к рискам** участников проекта.

и). **Формы отчетности**. Определяет **формат реестра рисков** и его содержание, а также любых других требуемых отчетов по рискам. Определяет, каким образом производится документирование, анализ и обмен информацией о результатах процесса управления рисками.

к). **Отслеживание.** Документирует порядок регистрации всех аспектов операций по рискам в интересах данного проекта, а также для будущих проектов и для включения в документы по накопленным знаниям. Документирует, в каких случаях и как будет проводиться аудит процессов управления рисками.

В ходе разработки плана управления рисками и последующих действий по их идентификации и оценке составляют **карты рисков** проекта.

Карты рисков проекта строятся в рамках трёх процессов.

1. Идентификация рисков. На данном этапе можно ограничиться заполнением первых восьми столбцов карты рисков (табл. 7.2) [2.1].

Таблица 7.2

№	Наименование риска	Организация	Событие	Вид риска	Описание риска	Причины	Последствия	Общий риск		
								Вероятность	Последствия	Оценка
1	2	3	4	5	6	7	8	9	10	11

Столбцы (2-5). Указать наименование риска, конкретное событие и вид риска в соответствии с приведенной классификацией рисков. Каждое событие (риск) должно однозначно соответствовать одной из групп риска. *Например:*

Наименование риска (2) — Конкуренты;

Событие (4) — Появление новых конкурентов на рынке;

Вид риска (5) — Стратегические риски.

Описание риска (6) должно включать определение риска с указанием факторов, причин и/или событий, обуславливающих вероятность проявления данного риска, а также обоснование возможных последствий.

Причины (7). Более подробный перечень причин, обуславливающих наступление данного события. Отсутствие контроля или методов воздействия на риск причинами не является.

Последствия (8). Более подробный перечень возможных последствий от наступления рискового события, выраженных в конкретных денежных потерях, потерях времени, качества, потерях репутации, доли рынка и т. д.

2. Качественный анализ рисков. На данном этапе заполняют 9-11 графы карты.

Общая оценка риска (9-11). Риск оценивается по двум параметрам: вероятность и последствия. На первом этапе риск оценивается до применения методов воздействия на риск. Оценка риска до воздействия является общей оценкой.

Вероятность (9) наступления рискового события оценивается в виде коэффициента, значение которого определяется экспертным методом в соответствии с табл. 1.

Последствия (10). В данном столбце необходимо указать коэффициент существенности последствий наступления рисковых событий в течение

выполнения работ по данному Проекту с предположением того, что данный риск не управляется и не контролируется. Значение коэффициента определяется экспертным методом в соответствии с табл. 2.

Существенность последствия рисков событий можно также оценить и в денежном выражении. Для каждого Проекта коэффициентам последствий могут соответствовать различные денежные выражения. В качестве базы для оценки последствий в денежном или другом выражении можно использовать: годовой оборот, годовую выручку, позицию на рынке, репутацию и т. п.

Оценка (11) риска рассчитывается как произведение коэффициента вероятности на коэффициент последствий и показывает общую оценку риска с предположением того, что данный риск не управляется и не контролируется.

Для наглядности и последующего анализа строится матрица вероятности и последствий.

3. Количественный анализ рисков и планирование реагирования на риски

В ходе планирования необходимо заполнить следующую часть карты рисков (табл. 7.3) [2.1].

Таблица 7.3

Владелец процесса	Воздействие на риск	Стоимость воздействия	Регламентация процесса	Остаточная оценка риска		
				Вероятность	Последствия	Оценка
12	13	14	15	16	17	18

Владелец процесса управления данным риском (12) – лицо, отвечающее за оценку, контроль и управление данным риском, т. е. мотивированное на его снижение и располагающее достаточными полномочиями для воздействия на него.

Воздействие на риск (13) – описание конкретных методов воздействия на риск, применяемых владельцем риска или другими подразделениями. Любое мероприятие или функция, способствующая снижению вероятности или последствий наступления рисков события, может быть рассмотрена как метод воздействия.

Стратегиями реагирования на негативные риски (угрозы) являются уклонение, передача или снижение.

Стратегиями реагирования на позитивные риски (благоприятные возможности) являются использование, совместное использование и усиление.

Стоимость воздействия (14). Экспертная оценка стоимости применяемых методов воздействия на риск. Рекомендуется использовать следующую шкалу: высокая, средняя и низкая стоимость воздействия, выраженную в денежном эквиваленте.

Документы, регламентирующие процессы управления риском (15). Утвержденные внутренние документы, регламентирующие тем или иным образом процессы управления риском: выявление, оценку, анализ, контроль или воздействие. К таким документам может относиться любой документ, в котором прямо или косвенно регламентирован процесс, позволяющий тем или иным образом снизить оценку данного риска.

Остаточная оценка риска (16-18). Оценивается по аналогии с общим риском по двум параметрам: вероятность и последствия.

Вероятность (16) наступления рискованного события с учетом того, что данный риск управляется и контролируется, т. е. после воздействия указанных выше методов управления. Оценивается экспертным методом в соответствии со шкалой идентичной оценки вероятности на предыдущем этапе.

Последствие (17) наступления рискованного события с учетом того, что данный риск управляется и контролируется, т. е. после воздействия методов управления. Оценивается экспертным методом в соответствии со шкалой идентичной оценки последствий на предыдущем этапе.

Оценка (18) — остаточная оценка риска с предположением того, что данный риск управляется и контролируется, т. е. после воздействия методов управления. Рассчитывается как произведение вероятности на последствия.

После этого для наглядности и последующего анализа ответственным за управление рисками проекта строится **матрица вероятности и последствий** с учетом реагирования на риски аналогично представленной матрицы без учета процедур реагирования.

Когда исключить все риски проекта маловероятно, либо невозможно найти подходящей стратегии реагирования на риски, команда проекта должна **принять риск**. Это называется **стратегией принятия**. Такая стратегия применима и к угрозам, и к благоприятным возможностям и может быть как пассивной, так и активной. В случае активной стратегии необходимо создать резерв на непредвиденные обстоятельства, включающий в себя время, деньги или ресурсы для управления принятыми (и неизвестными) рисками. В этом случае необходимо установить жесткий контроль (мониторинг) над событиями, запускающими в действие данные риски, т. е. составить список событий для постоянного наблюдения.

Результатами планирования реагирования на риски проекта должны стать **новая версия карты рисков**, обновленный **план управления проектом**, включающий в себя операции по реагированию на риски, и **контрактные соглашения** для определения ответственности каждой из сторон на случай возникновения каждого отдельного риска.

Карта рисков на начальном этапе может быть представлена в виде **реестра рисков проекта**.

Инструкции по заполнению реестра рисков

ID	Уникальный идентификатор риска (rs + инкремент)
Статус риска	"Открыт" для рисков, которые актуальны; "закрыт" для рисков, более не актуальных на проекте (реализовавшихся, ставших невозможными и т.п.)
Влияние риска	Элемент качественной оценки риска ("высокое / среднее / низкое") (выделение цветом автоматическое, соответственно: красный / желтый / зеленый)
Вероятность риска	Элемент качественной оценки риска ("высокая / средняя / низкая") (выделение цветом автоматическое, соответственно: красный / желтый / зеленый)

Уровень риска	Интегральная качественная оценка риска ("красный" – для самых опасных и приоритетных рисков, "желтый" – для умеренных", "зеленый" – для наименее существенных). Оценка формируется автоматически, на основании данных столбцов «влияние риска» и «вероятность риска». Соответственно: <table><tr><td></td><td></td><td>(зеленый + зеленый) = зеленый уровень риска</td></tr><tr><td></td><td></td><td>(зеленый + желтый) = зеленый уровень риска</td></tr><tr><td></td><td></td><td>(желтый + желтый) = желтый уровень риска</td></tr><tr><td></td><td></td><td>(желтый + красный) = красный уровень риска</td></tr><tr><td></td><td></td><td>(красный + красный) = красный уровень риска</td></tr></table>			(зеленый + зеленый) = зеленый уровень риска			(зеленый + желтый) = зеленый уровень риска			(желтый + желтый) = желтый уровень риска			(желтый + красный) = красный уровень риска			(красный + красный) = красный уровень риска
		(зеленый + зеленый) = зеленый уровень риска														
		(зеленый + желтый) = зеленый уровень риска														
		(желтый + желтый) = желтый уровень риска														
		(желтый + красный) = красный уровень риска														
		(красный + красный) = красный уровень риска														
Описание риска	Тезисно – в чем суть (причина, содержание) риска															
Влияние на проект	Тезисно – в чем суть влияния риска на проект															
Область риска	Общее название группы, к которой можно отнести данный риск															
План А (contingency plan)	Что будем делать для того, чтобы реализовать стратегию обработки риска															
Триггеры	Условия для запуска действий по плану Б (одновременно – признак того, что риск реализовался)															
Тип стратегии обработки риска	Для позитивных рисков: "использование / усиление / разделение"; для негативных рисков: "предотвращение / смягчение / перенос"; для обоих видов риска: "принятие"															
Хозяин риска	Лицо ответственное за мониторинг триггера и запуск contingency плана															
План Б (fallback plan)	Что будем делать, если риск реализовался (не заполняется, если тип стратегии обработки риска – "принятие")															

Задание: разработать План управления рисками проекта создания программного продукта ВКР и соответствующие реестр и карту рисков проекта. Из 10 наиболее распространенных рисков программного проекта для ВКР наиболее актуальны:

Нереалистичные сроки и бюджет.

Реализация несоответствующей функциональности.

Разработка неправильного пользовательского интерфейса.

Перфекционизм, ненужная оптимизация и оттачивание деталей.

Непрекращающийся поток изменений.

Нехватка информации о внешних компонентах, определяющих окружение системы или вовлеченных в интеграцию.

Простейшая форма Реестра рисков (табл. 7.4):

Таблица 7.4

Идентификация риска									
№	Дата возникновения риска	Дата регистрации риска	Наименование риска	Описание риска	Инициатор	Причины, вызвавшие риск	Последствия	Владелец риска	Дата окончания действия риска
1	2	3	4	5	6	7	8	9	10

Пример карточки с описанием риска [2.1]

Номер: R-101	Категория: Технологический.
Причина: Недостаток квалифицированных кадров.	Симптомы: Разработчики будут использовать новую платформу — J2EE.
Последствия: Низкая производительность разработки	Воздействие: Увеличение сроков и трудоемкости разработки.
Вероятность: Очень вероятно.	Степень воздействия: Критичная.
Близость: Очень скоро.	Ранг: 6.
Исходные данные: «Содержание проекта», «План обеспечения ресурсами», Протоколы совещаний №21 от 01.06.2008, №27 от 25.06.2008.	

2.8. Лабораторная работа 8: Планирование подготовки и выпуска документов проекта

Документ проекта – это законченный (самостоятельный) документ, подробно описывающий специфические действия организации при начале, планировании, выполнении и завершении проектов.

Документация используется для хранения и передачи знаний и фактов самого разного характера, она **формализует договоренности и обязательства**, под которыми можно понимать широкий спектр обещаний, намерений или заданий, циркулирующих между одной или несколькими организациями.

Документацию, используемую в проектах разработки ПО, можно разделить на следующие группы:

Документация проекта	Документация продукта
Описание проекта Планы	Технические требования Технические спецификации
Задания исполнителям Отчеты о ходе работ	Сведения о выпуске Руководства
Протоколы Отчеты о результатах активностей Журналы	

Задание:

Определить перечень документов, необходимых для проекта программного продукта, разрабатываемого в ВКР, дать их краткую аннотацию, определить сроки и субъектов создания, согласования и утверждения.

2.9. Лабораторная работа 9: График разработки и внедрения программного продукта с учётом конкурирующих ресурсов

Разработку графика выполнения проекта обеспечивают пять процессов:

Определение состава операций - процесс определения конкретных плановых операций, которые необходимо выполнить для получения результатов проекта - внедрения продукта.

Определение взаимосвязей операций - процесс выявления и документирования последовательности выполнения плановых операций.

Определение ресурсов операции - процесс определения необходимых для выполнения каждой плановой операции ресурсов и их количества.

Определение длительности операций - процесс определения продолжительности выполнения каждой плановой операции.

Разработка расписания - процесс составления расписания проекта с учетом последовательностей операций, их длительности, требований к ресурсам и ограничений на сроки выполнения проекта в целом.

Инструментальным средством для определения состава операций, а также для оценки их взаимосвязи и длительности, служит **ИСР**. В процессе декомпозиции определяется нижний уровень управления, с которым работает руководитель проекта, - уровень **пакетов работ**.

Пакет работ состоит из операций, имеющих общие функции или конечный результат. Пакеты работ разбивают на операции.

Операция - это единица работ, в результате которой имеется конкретный результат по внедрению информационной системы.

Перед началом определения состава операций рекомендуется еще раз проанализировать описание содержания проекта, ограничения и допущения с точки зрения полноты списка операций, который будет основой для составления смет, планирования сроков выполнения и контроля проектных работ.

Список операций - перечень работ, запланированных для выполнения.

Параметры операций - могут включать в себя идентификатор операции, коды операции, длительность, начало, окончание, исполнителя операции, перечни предшествующих и последующих операций, логические взаимосвязи, опережения и задержки, плановую трудоемкость работ и другие необходимые для управления проектом параметры операций.

Список контрольных событий (вех проекта) - определяет все контрольные события расписания, необходимые для мониторинга хода выполнения и для управления проектом. Список контрольных событий является элементом плана управления проектом. Веха проекта определяет момент перехода проекта из одного состояния в другое.

Пример состава операций представлен в таблице 9.1.

Пример списка состава операций

Наименование пакета работ	Наименование операций
Обследование	Формирование и согласование плана проведения интервью
	Подготовка и рассылка опросных листов для интервью
	Проведение интервью для описания бизнес-процессов
Описание бизнес-процессов	Описание бизнес-процессов по функциональным областям
	Описание интеграционных, управляющих и обеспечивающих бизнес-процессов
Разработка продукта	Разработка решений по функциональной архитектуре
	Подготовка функционального дизайна расширений
	Настройка системы
	Техническое проектирование расширений
	Разработка расширений
	Техническое проектирование программ конвертации данных
	Разработка программ конвертации данных
	Планирование тестирования приложения и интеграционного тестирования
	Разработка сценариев тестирования
	Подготовка тестовых данных
Тестирование системы	Проведение тестирования по функциональным областям/модулям
	Проведение интеграционного тестирования
	Проведение тестирования конвертации данных

Планирование сроков проекта может быть выполнено с помощью специализированных программных средств.

Определение взаимосвязей операций включает в себя идентификацию и документирование логических взаимосвязей между плановыми операциями. Взаимосвязи операций могут быть последовательными, с собственными отношениями предшествования, а также опережением и задержками. В этом случае каждый выходной элемент операции используется как входной элемент другой операции или является частью поставки. Взаимосвязи операций могут быть с перекрытиями, когда еще незавершенная операция имеет достаточно выходных элементов для начала зависящей от нее операции, или с параллельным выполнением операций.

Для определения последовательности операций используется три типа зависимостей.

Жесткая или **обязательная зависимость** - зависимость, при которой последовательность работ не может изменяться. Обязательные зависимости

являются неотъемлемым свойством выполняемой работы и часто подразумевают физические ограничения на последовательность выполнения операций.

Нежесткая или **произвольная зависимость** - последовательность определяется командой проекта и может изменяться.

Внешняя зависимость - последовательность работ определяется внешними по отношению к проекту воздействиями. Внешние зависимости включают взаимоотношения операций проекта с внешними операциями. Например, в проекте по разработке программного обеспечения сроки операции тестирования могут зависеть от поставки аппаратного обеспечения сторонней организацией.

Оценка ресурсов плановой операции призвана определить, какие ресурсы (люди, оборудование и проч.) будут использоваться и в каком количестве и когда каждый из ресурсов будет доступен для выполнения проектных операций.

Оценка "снизу вверх".

Когда плановую операцию нельзя оценить с достаточной степенью уверенности, то работы в пределах такой операции разбиваются на более мелкие элементы. Ресурсные потребности каждого более детализированного элемента работ оцениваются, и эти оценки объединяются в общее количество по каждому ресурсу плановой операции. Плановые операции могут быть связаны отношениями зависимости, которые могут влиять на привлечение и использование ресурсов, но могут и не иметь такой связи. Если отношений зависимости нет, то эта специфика применения ресурсов отражается в оценочных требованиях плановой операции и фиксируется документально. Процесс оценки длительности операций требует, чтобы были оценены объем работы, расчетное количество ресурсов и определено количество рабочих. Оценка длительности операции проводится с помощью ИСР. Общая длительность проекта рассчитывается как результат процесса разработки расписания.

Оценка по аналогам подразумевает оценку фактической длительности аналогичной предыдущей плановой операции в качестве основы для оценки длительности будущей плановой операции и использует историческую информацию и экспертную оценку.

Параметрическая оценка. Оценочную величину длительности операций можно вычислить путем умножения количества работы на производительность труда. Для определения длительности операций по рабочим периодам общее количество ресурсов умножается на количество рабочего времени или производительность за рабочий период и делится на количество привлеченных ресурсов.

Оценка по трем точкам. Точность оценки длительности операций можно увеличить, если в исходной оценке учитывать размер рисков. Оценка по трем точкам основана на определении трех типов оценок: Наиболее вероятная - длительность плановой операции с учетом предварительного выделения ресурсов, их производительности, реалистичной оценки их доступности для выполнения данной плановой

операции, отношений зависимости с другими участниками, задержек.

Оптимистичная. Длительность операции основывается на оптимистичном сценарии, представленном в наиболее вероятной оценке.

Пессимистичная. Длительность операции основывается на пессимистичном сценарии, представленном в наиболее вероятной оценке.

Оценка длительности операции может быть выведена с использованием средней из трех оценок длительности по методу PERT.

Длительность операции = ((наиболее вероятная оценка) x 4 + оптимистичная + пессимистичная)/6

При разработке расписания учитываются два основных типа ограничений по времени:

Требуемые даты для начала или завершения операции, которые можно использовать для ограничения начала или завершения операции.

Контрольные события, вследствие чего получение определенных результатов работ привязывается к определенным датам, изменить которые можно только посредством одобренных изменений.

Необходимо также учитывать План управления стоимостью, План управления содержанием проекта, План закупок и План управления рисками.

Задание:

Для программного продукта, создаваемого в ВКР, на основе данных предыдущих практических и лабораторных работ уточнить состав операций, перечень событий, определить взаимосвязи операций и их ресурсы, определить длительности операций и разработать расписание проекта (от начала разработки до завершения внедрения продукта у заказчика) и календари ресурсов. Расписание представить в виде диаграммы Ганта или сетевого графика с комментариями по событиям, операциям и ресурсам.

2.10. Лабораторная работа 10: Планирование и управление бюджетом расходов программного проекта

Бюджет проекта представляет собой распределение статей расходов и доходов по периодам времени (например по дням, месяцам, кварталам). Процесс формирования, учета и контроля выполнения бюджетов называется бюджетированием. Элементами процесса бюджетирования являются:

структура расходов и доходов;

распределение расходов и доходов во времени;

структура центров ответственности и распределение ответственности между ними за статьи доходов и расходов;

процессы планирования, учета и контроля, которые предусматривают сбор и интеграцию плановой и фактической информации по центрам ответственности.

Распределение ответственности за статьи доходов и расходов выполняется путем построения **матрицы ответственности** за статьи затрат (таблица 10.1). **Статьи затрат** определяются по классификации доходов и

расходов, принятой в компании; **центры ответственности** определены на основании организационной структуры проекта. Каждый проект имеет свои статьи доходов и расходов, некоторые проекты имеют только расходные статьи бюджета. Структура статей расходов включает **прямые затраты**, структурируемые **по ИСР**, и затраты на **накладные расходы** проекта. Разработка бюджета расходов - процесс назначения оценок стоимости всем операциям проекта, результатом которого является создание **базового плана по стоимости** проекта. Бюджет расходов содержит объединенные оценки стоимости отдельных плановых операций или пакетов работ. Процесс разработки бюджета расходов включает в себя процесс разработки бюджета для **непредвиденных обстоятельств**, куда закладываются ожидаемые значения воздействия всех идентифицированных рисков. Ожидаемое значение рассчитывается на основе оптимистичной, пессимистичной и наиболее вероятной оценок величины риска. В бюджет расходов следует включить **управленческий резерв** - деньги, предусмотренные в бюджете проекта на неидентифицированные риски.

Таблица 10.1

Шаблон матрицы ответственности за статьи затрат

Статья затрат	Центр ответственности (подразделения)									
	100 0	110 0	111 0	112 0	120 0	121 0	122 0	— —	— —	— —
Заработная плата персонала проекта										
Оплата услуг связи										
Командировочные расходы										
Аренда помещения										
Амортизация оборудования										

Результаты разработки бюджета расходов.

Базовый план по стоимости - распределенный по времени бюджет, используемый для мониторинга и контроля исполнения стоимости проекта. Он разрабатывается путем суммирования оценок стоимости расходов по периодам времени и обычно имеет вид S-кривой. Базовый план по стоимости является элементом плана управления проектом. Проекты, особенно крупные, могут иметь несколько базовых планов по стоимости, отражающих разные аспекты процесса исполнения стоимости: расходы, входящие платежи, выполненную стоимость.

Таблица 10.2

Определение статей расходов и доходов для различных типов базового плана

№	Тип базового плана	Статьи расходов и доходов
1	Базовый план для измерения <i>исходящего</i> потока денежной наличности	Заработная плата персонала проекта. Выплаты подрядчикам. Закупка оборудования. Аренда помещения под проектный офис. Транспортные расходы. Оплата за услуги связи (телефон, Интернет) и проч.
2	Базовый план для измерения <i>входящего</i> потока денежной наличности	Поступления от Заказчиков за поставленные услуги: разработка технического задания; реализация проектных решений; тестирование интеграционных сценариев тестирования; разработка документов Системы; разработка Материалов Обучения; Централизованное обучение IT-специалистов Заказчика

3. **Установление критериев** формирования базового плана - установление отношений между оценкой стоимости и параметрами времени. Критерии определяют события проекта, инициирующие выплаты по статьям расходов базового плана, и интервал времени между инициирующим событием и датой выплаты (таблица 10.3).

Таблица 10.3

Критерии базового плана стоимости

Статьи расходов	Событие, инициирующее выплаты	Интервал между событием и выплатой	Примечание
Зарплата управленческого персонала	Согласно рабочему расписанию	1 месяц	Политика компании
Зарплата консультантов	Согласно рабочему расписанию	1 месяц	Политика компании
Оплата услуг связи	Согласно договору на услуги связи	1 неделя	
Аренда помещения	Согласно договору об аренде	1 неделя	

Далее следует:

4. Распределение статей доходов по временным периодам.
5. Суммирование оценочных значений расходов по временным периодам.
6. Графическое и табличное изображение базового плана стоимости (рис. 3).

Требования к финансированию проекта формируются на основании базового плана стоимости и представляют сумму средств, указанных в базовом плане по стоимости, и резерва на непредвиденные обстоятельства.

Задание:

На основании ИСР и сметы затрат по проекту, разрабатываемому в ВКР, составить базовый план для исходящего потока денежной наличности, таблицу критериев базового плана и детальный план-график платежей, построить график выполнения бюджета расходов нарастающим итогом на весь горизонт планирования проекта. Временные интервалы выбрать сообразно основным событиям проекта.

2.11. Лабораторная работа 11: План закрытия проекта и оформление отчётности

Для качественного завершения проекта, достижения удовлетворённости сторон, предотвращения потерь и сохранения опыта действия по закрытию проекта должны носить плановый характер.

План закрытия проекта включает в себя все действия, необходимые для административного завершения проекта или фазы, включая:

действия и операции, необходимые для удовлетворения критериев завершения или выхода для фазы или проекта;

действия и операции, необходимые для передачи продуктов, услуг или результатов проекта в следующую фазу или в производство и/или операционную деятельность;

операции, необходимые для сбора документов проекта или фазы, проверки успешности или неудачи проекта, аккумуляирования полученных знаний и архивирования информации по проекту для будущего использования организацией.

В ходе проекта накапливается большой объем самой разнообразной информации:

чертежи, технические спецификации и руководства по эксплуатации оборудования, закупленного для проекта;

техническое описание или схему результата проекта;

планы и программы работ по проекту;

учетные документы, касающиеся расходов на реализацию проекта;

контракты и субконтракты по проекту.

Осуществление закрытия состоит из следующих *операций*:

составить список открытых вопросов и заключительных работ;

сдать результаты проекта Заказчику;

принять поручение Заказчика о закрытии проекта;

закрыть все ресурсы и передать их на новые объекты;

выполнить заключительную оценку финансовой ситуации (постпроектный отчет), закрыть все финансовые операции и проследить, чтобы все счета были оплачены;

выполнить документирование и анализ опыта выполнения проекта: создать заключительный отчет по проекту и сдать проектную документацию;

разрешить все спорные вопросы;

оценить работу проектной команды, членов проектной команды и руководителя проекта (в отдельной части отчёта и в виде благодарностей или иных

документов морального и материального стимулирования, например, КТУ — коэффициент трудового участия);
оформить отпуск и перераспределение членов проектной команды.

Итоговый отчет должен содержать следующую информацию.

Итоги проекта:

Достижение целей проекта

Дополнительные полезные результаты

Фактические сроки

Фактические расходы

Обоснование отклонения от целей

Отклонения результатов от требований

Уроки проекта:

Проблемы проекта и способы их решения

Материалы программные компоненты для последующего использования

Предложения по изменению технологий или стандартов компании

Задание:

Составить список открытых вопросов и заключительных работ по фазе ВКР.

Разработать **План закрытия проекта ВКР** как фазы разработки программного продукта по факту реально выполненных в рамках ВКР работ по созданию и внедрению программного продукта.

Подготовить **Итоговый отчет по фазе ВКР**

Составить **перечни документов и программных компонент** с краткой аннотацией и номером версии, передаваемых на кафедру, руководителю проекта и в архив проекта (личный и/или заказчика). Сюда может, в частности, входить информация по проблемам и рискам, а также по успешно примененным методам, которые могут быть использованы в будущих проектах.

Библиографический список

Основная литература

1. Сооляттэ, А.Ю. Управление проектами в компании: методология, технологии, практика [Электронный ресурс]: учебник / А.Ю. Сооляттэ. - Москва: Московский финансово-промышленный университет «Синергия», 2012. - 816 с.: ил., табл., схемы Режим доступа: <http://biblioclub.ru/index.php?page=book&id=252963>

Дополнительная литература

1. Архипенков, С. Лекции по управлению программными проектами [Электронный ресурс] / С. Архипенков. – Москва, 2009. – Режим доступа: http://www.arkhipenkov.ru/resources/sw_project_management.pdf
2. Вендров, А.М. Проектирование программного обеспечения информационных систем / А.М. Вендров. – Москва: Финансы и статистика, 2006. – 544 с.
3. Вендров, А. М. Практикум по проектированию программного обеспечения экономических информационных систем: учеб. пособие для вузов по специальностям: "Прикладная информатика в экономике", "Мат. обеспечение и администрирование информ. систем" / А. М. Вендров. – 2-е изд., перераб. и доп. – Москва: Финансы и статистика, 2006. – 191 с.
4. Липаев, В.В. Документирование сложных программных комплексов [Электронный ресурс]:

- электронное дополнение к учебному пособию «Программная инженерия сложных заказных программных продуктов» / В.В. Липаев. – Москва; Берлин: Директ-Медиа, 2015. - 158 с. Режим доступа: <http://biblioclub.ru/index.php?page=book&id=260688>
5. Липаев, В.В. Программная инженерия: методологические основы [Электронный ресурс]: учебник / В.В. Липаев. – Москва; Берлин: Директ-Медиа, 2015. - 608 с. Режим доступа: <http://biblioclub.ru/index.php?page=book&id=260690>
6. Липаев, В.В. Программная инженерия сложных заказных программных продуктов [Электронный ресурс]: учебное пособие / В.В. Липаев. – Москва; Берлин: Директ-Медиа, 2015. - 305 с. Режим доступа: <http://biblioclub.ru/index.php?page=book&id=278971>
7. Орлов, С. А. Технологии разработки программного обеспечения: разработка сложных програм. систем: учебник для вузов по направлению "Информатика и вычисл. техника" / С. А. Орлов. - 3-е изд. - Санкт-Петербург [и др.]: Питер , 2004 . - 526 с.
8. Павлов, А. Н. Управление проектами на основе стандарта PMI PMBOK®: изложение методологии и опыт применения/ А. Н. Павлов. - 2-е изд., испр. - Москва: БИНОМ. Лаборатория знаний, 2012. - 208 с.: табл., схемы.
9. Ройс, У. Управление проектами по созданию программного обеспечения: пер. с англ. / У. Ройс. – Москва: ЛОРИ, 2002. – 424 с.
10. Сергушичева, А. П. Жизненный цикл программного продукта: учебное пособие/ А. П. Сергушичева. - Вологда: ВоГТУ, 2010. - 135, [1] с.: ил., табл. Режим доступа: http://www.library.vstu.edu.ru/biblio/sergushicheva/book7/2010_sergushicheva_jizn_cikl.pdf
11. Фатрелл, Р.Т. Управление программными проектами. Достижение оптимального качества при минимуме затрат / Р.Т.Фатрелл, Д.Ф.Шафер, Л.И.Шафер. – Москва: Вильямс, 2004. – 1125 с.

Программное обеспечение и Интернет-ресурсы

1. Guide to the Software Engineering Body of Knowledge (SWEBOK), IEEE 2004 Version. [Электронный ресурс]. – Режим доступа: http://www.swebok.org/ironman/pdf/Swebok_Ironman_June_23_%202004.pdf /.
2. НОЦ технологий управления информацией: когда бизнес верит в инновации. Логика процессного управления [Электронный ресурс]. – Режим доступа: <http://www.bpmlogic.info/2011/09/sbpm.html/>.
3. Нижегородский государственный университет им. Н.И. Лобачевского. Факультет вычислительной математики и кибернетики ННГУ. Учебно-исследовательская лаборатория "Математические и программные технологии для современных компьютерных систем (Информационные технологии): Обзор моделей жизненного цикла разработки программного обеспечения [Электронный ресурс]. – Режим доступа: <http://www.itlab.unn.ru/MiniProjects/LCM/Conspect.doc>
4. Фонд поддержки системного проектирования, стандартизации и управления проектами (Фонд ФОСТАС) [Электронный ресурс]. – Режим доступа: <http://www.fostas.ru/about/arch.php/>.