

Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Пекаревский Борис Владимирович  
Должность: Проректор по учебной и методической работе  
Дата подписания: 04.05.2023 13:49:08  
Уникальный программный ключ:  
3b89716a1076b80b2c167df0f27c09d01782ba84



**МИНОБРНАУКИ РОССИИ**  
федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Санкт-Петербургский государственный технологический институт  
(технический университет)»

**УТВЕРЖДАЮ**  
Проректор по учебной  
и методической работе  
\_\_\_\_\_ Б.В.Пекаревский  
«17» мая 2019 г.

**Рабочая программа дисциплины**  
**СОВРЕМЕННЫЕ ТЕХНОЛОГИИ РАЗРАБОТКИ ПРОГРАММНОГО**  
**ОБЕСПЕЧЕНИЯ**

Направление подготовки

**09.04.01 Информатика и вычислительная техника**

Направленность программы магистратуры

**Информационное и программное обеспечение автоматизированных систем**

Квалификация

**Магистр**

Форма обучения

**Заочная**

Факультет **информационных технологий и управления**

Кафедра **систем автоматизированного проектирования и управления**

Санкт-Петербург

2019

**Б1.О.06**

## ЛИСТ СОГЛАСОВАНИЯ

Должность разработчика	Подпись	Ученое звание, инициалы, фамилия
Доцент		И.Г. Корниенко
Ст. преподаватель		А.К. Федин

Рабочая программа дисциплины «Современные технологии разработки программного обеспечения» обсуждена на заседании кафедры систем автоматизированного проектирования и управления  
протокол от «18» апреля 2019 № 9  
Заведующий кафедрой

Т.Б. Чистякова

Одобрено учебно-методической комиссией факультета информационных технологий и управления  
протокол от «15» мая 2019 № 9  
Председатель

В.В.Куркина

## СОГЛАСОВАНО

Руководитель направления подготовки «Информатика и вычислительная техника»		профессор Т.Б. Чистякова
Директор библиотеки		Т.Н. Старостенко
Начальник методического отдела учебно-методического управления		Т.И. Богданова
Начальник учебно-методического управления		С.Н. Денисенко

## СОДЕРЖАНИЕ

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы .....	04
2. Место дисциплины в структуре образовательной программы.....	06
3. Объем дисциплины .....	06
4. Содержание дисциплины	
4.1. Разделы дисциплины и виды занятий.....	07
4.2. Занятия лекционного типа.....	08
4.3. Занятия семинарского типа.....	09
4.3.1. Семинары, практические занятия .....	09
4.3.2. Лабораторные занятия .....	10
4.4. Самостоятельная работа обучающихся.....	10
4.5. Темы и содержание контрольных работ.....	11
5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине .....	19
6. Фонд оценочных средств для проведения промежуточной аттестации.....	20
7. Перечень учебных изданий, необходимых для освоения дисциплины .....	21
8. Перечень электронных образовательных ресурсов, необходимых для освоения дисциплины.....	21
9. Методические указания для обучающихся по освоению дисциплины.....	21
10. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине	
10.1. Информационные технологии.....	22
10.2. Программное обеспечение.....	22
10.3. Базы данных и информационные справочные системы.....	23
11. Материально-техническое обеспечение освоения дисциплины в ходе реализации образовательной программы .....	23
12. Особенности освоения дисциплины инвалидами и лицами с ограниченными возможностями здоровья .....	23
Приложение № 1 Фонд оценочных средств для проведения промежуточной аттестации по дисциплине «Современные технологии разработки программного обеспечения».....	24
Приложение № 2 Шаблон задания на курсовой проект.....	34

**1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы.**

В результате освоения образовательной программы магистратуры обучающийся должен овладеть следующими результатами обучения по дисциплине:

Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Планируемые результаты обучения (дескрипторы)
<p><b>ОПК-2</b> Способен разрабатывать оригинальные алгоритмы и программные средства, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач</p>	<p><b>ОПК-2.2</b> Создание программных средств для решения задач проектирования, обработки информации, управления с использованием современных технологий и инструментальных средств разработки программного обеспечения</p>	<p><b>Знать:</b> современные технологии и средства разработки программного обеспечения для решения профессиональных задач (ЗН-1).</p> <p><b>Уметь:</b> обосновывать выбор современных технологий и программной среды при разработке оригинальных программных средств для решения профессиональных задач (У-1).</p> <p><b>Владеть:</b> навыками разработки оригинальных программных средств с использованием современных технологий и инструментальных средств разработки программного обеспечения для решения профессиональных задач (Н-1).</p>
<p><b>ОПК-5</b> Способен разрабатывать и модернизировать программное и аппаратное обеспечение информационных и автоматизированных систем</p>	<p><b>ОПК-5.1</b> Разработка и модернизация проблемно-ориентированного программного обеспечения информационных и автоматизированных систем различного назначения</p>	<p><b>Знать:</b> современное программное обеспечение информационных и автоматизированных систем (ЗН-2).</p> <p><b>Уметь:</b> модернизировать программное обеспечение информационных и автоматизированных систем для решения профессиональных задач (У-2).</p> <p><b>Владеть:</b> навыками разработки программного обеспечения информационных и автоматизированных систем для решения профессиональных задач (Н-2).</p>

Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Планируемые результаты обучения (дескрипторы)
<p><b>ОПК-8</b> Способен осуществлять эффективное управление разработкой программных средств и проектов</p>	<p><b>ОПК-8.1</b> Выбор и применение современных технологий и инструментальных средств при разработке приложений, баз данных, программных и пользовательских интерфейсов</p>	<p><b>Знать:</b> основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем (ЗН-3).</p> <p><b>Уметь:</b> применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для решения прикладных задач различных классов, ведения баз данных и информационных хранилищ (У-3).</p> <p><b>Владеть:</b> навыками программирования, отладки и тестирования прототипов программных комплексов (Н-3).</p>

## 2. Место дисциплины в структуре образовательной программы.

Дисциплина относится к дисциплинам обязательной части (Б1.О.06) и изучается на установочной сессии и зимней сессии 1 курса.

В методическом плане дисциплина опирается на элементы компетенций, сформированные при изучении дисциплин в бакалавриате «Программирование», «Операционные системы», «Базы данных», «Лингвистическое и программное обеспечение автоматизированных информационных систем». Полученные в процессе изучения дисциплины «Современные технологии разработки программного обеспечения» знания, умения и навыки могут быть использованы при прохождении производственной и преддипломной практики, а также при выполнении выпускной квалификационной работы.

## 3. Объем дисциплины.

Вид учебной работы	Всего, ЗЕ/акад. часов		
	Курс 1		Итого:
	Установочная сессия	Зимняя сессия	
<b>Общая трудоемкость дисциплины</b> (зачетных единиц/ академических часов)	<b>1/36</b>	<b>4/144</b>	<b>5/180</b>
<b>Контактная работа с преподавателем:</b>	<b>4</b>	<b>14</b>	<b>18</b>
занятия лекционного типа	4	–	4
занятия семинарского типа, в т.ч.	–	8	8
семинары, практические занятия	–	8	8
лабораторные работы	–	–	–
курсовое проектирование (КР или КП)	–	6	6
КСР	–	–	–
другие виды контактной работы	–	–	–
<b>Самостоятельная работа</b>	<b>32</b>	<b>121</b>	<b>153</b>
<b>Форма текущего контроля</b> (Кр, реферат, РГР, эссе)	–	Кр №1, Кр №2, Кр №3	3 Кр
<b>Форма промежуточной аттестации</b> (КР, КП, зачет, экзамен)	–	<b>КП, Экзамен/9</b>	<b>КП, Экзамен/9</b>

#### 4. Содержание дисциплины.

##### 4.1. Разделы дисциплины и виды занятий.

№ п/п	Наименование раздела дисциплины	Занятия лекционного типа, акад. часы	Занятия семинарского типа, акад. часы		Самостоятельная работа, акад. часы		Формируемые компетенции	Формируемые индикаторы		
			Семинары и/или практические занятия	Лабораторные работы						
			Курс 1	Курс 1					Курс 1	Курс 1
			Уст. сессия	Зим. сессия					Зим. сессия	Уст. сессия
1	Проблемы разработки сложных программ. Организация жизненного цикла ПО, каскадные и итеративные модели жизненного цикла.	1	2	-	8	25	ОПК-2	ОПК-2.2		
2	Стандарты, регулирующие процессы разработки ПО.	1	4	-	8	24	ОПК-8	ОПК-8.1		
3	Связь тестирования и качества разрабатываемого ПО	0,5	2	-	4	24	ОПК-5	ОПК-5.1		
4	Современные тенденции в архитектуре компьютеров – многоядерные процессоры – и архитектура ПО	1	-	-	8	24	ОПК-8	ОПК-8.1		
5	Виртуализация как новый подход к построению IT-инфраструктуры.	0,5	-	-	4	24	ОПК-2	ОПК-2.2		

#### 4.2. Занятия лекционного типа.

№ раздела дисциплины	Наименование темы и краткое содержание занятия	Объем, акад. часы	Инновационная форма
1	Проблемы разработки сложных программ. Жизненный цикл программного обеспечения (ПО) и процессов разработки ПО. Каскадные и итеративные модели жизненного цикла, и набор стандартов, регулирующих процессы разработки ПО. Мобильность и переносимость ПО. Преимущества и недостатки переносимости программ. Принципы разработки приложений в многоуровневой архитектуре ПО.	1	
2	Стандарты, регулирующие процессы разработки ПО. Методология быстрой разработки приложений (RAD), методологии унифицированного процесса разработки Rational (RUP) и экстремального программирования (XP). Влияние архитектуры ПО на его свойства. Структурный подход в проектировании ПО и классификация структурных методологий. Диаграммы «сущность-связь» (ERD), диаграммы потоков данных (DFD), SADT-модели (стандарт IDEF0). Унифицированный язык моделирования UML. Основные понятия унифицированного языка моделирования (UML). Виды диаграмм UML. Тенденции развития объектно-ориентированных инструментальных средств.	1	
3	Связь тестирования и качества разрабатываемого ПО. Классификация типов тестов. Документирование и анализ ошибок. Оценка степени тестируемости ПО. Тестирование производительности и стабильности ПО. Оптимизация ПО с учетом производительности. Инструменты и концепции повышения производительности. Тесты на производительность.	0,5	

№ раздела дисциплины	Наименование темы и краткое содержание занятия	Объем, акад. часы	Инновационная форма
4	Современные тенденции в архитектуре компьютеров – многоядерные процессоры – и архитектура ПО. Проблематика современных вычислений и предпосылки к параллельным вычислениям. Математические основы параллельных вычислений. Высокопроизводительные и параллельные вычисления в научных исследованиях. Характер увеличения скорости программы в результате распараллеливания. Реализация параллельных вычислений. Интерфейсы прикладного программирования для разработки параллельных программ. Проблемы распределения ресурсов и исключения тупиков при разработке сложных программ. Основные направления борьбы с тупиками.	1	
5	Виртуализация как новый подход к построению IT-инфраструктуры. Две фундаментально различающиеся категории виртуализации: виртуализация платформ и виртуализация ресурсов. Виртуализация уровня операционной системы. Виртуализация уровня приложений. Применение виртуализации при разработке и тестировании приложений. Нагрузочное тестирование	0,5	КтСм

### 4.3. Занятия семинарского типа.

#### 4.3.1. Семинары, практические занятия.

№ раздела дисциплины	Наименование темы и краткое содержание занятия	Объем, акад. часы	Инновационная форма
1	Жизненный цикл ПО Использование моделей жизненного цикла ПО: организация жизненного цикла ПО; каскадные и итеративные модели жизненного цикла; набор стандартов, регулирующих процессы разработки ПО.	2	
2	Методологии разработки ПО. Использование основных методологий разработки ПО: методология быстрой разработки приложений (RAD); методологии унифицированного процесса разработки Rational (RUP); экстремальное программирование (XP)	1	

№ раздела дисциплины	Наименование темы и краткое содержание занятия	Объем, акад. часы	Инновационная форма
2	Влияние архитектуры ПО на его свойства. Использование базовых диаграмм при проектировании ПО: структурный подход в проектировании ПО; классификация структурных методологий; диаграммы «сущность-связь» (ERD); диаграммы потоков данных (DFD); SADT-модели (стандарт IDEF0).	2	
2	Унифицированный язык моделирования UML. Использование базовых диаграмм UML при проектировании ПО: основные понятия UML; диаграммы прецедентов; диаграммы классов; диаграммы взаимодействий; диаграммы последовательности действий; диаграммы состояний, компонентные диаграммы; тенденции развития объектно-ориентированных инструментальных средств.	1	
3	Тестирование ПО. Использование тестирования ПО разных типов: классификация типов тестов; документирование и анализ ошибок; оценка степени тестируемости ПО; тестирование производительности и стабильности ПО.	2	КтСм

#### 4.3.2. Лабораторные занятия.

Учебным планом не предусмотрены.

#### 4.4. Самостоятельная работа обучающихся.

№ раздела дисциплины	Перечень вопросов для самостоятельного изучения	Объем, акад. часы	Форма контроля
2	<b>Структурный подход в проектировании ПО,</b> Классификация структурных методологий,, диаграммы «сущность-связь» (ERD), диаграммы потоков данных (DFD), SADT-модели (стандарт IDEF0).	17	Кр №1 Кр №2
2	<b>Унифицированный язык моделирования UML.</b> Основные диаграммы. Тенденции развития объектно-ориентированных инструментальных средств.	17	Кр №1
1,3,4	<b>Индекс производительности.</b> Исследование системного ПО, поставляемого в комплекте операционной системы, повышающего производительность ПО и средств других фирм производителей.	17	Кр №3
1,2,3,4	<b>Оптимизация ПО.</b>	17	Кр №1

№ раздела дисциплины	Перечень вопросов для самостоятельного изучения	Объем, акад. часы	Форма контроля
	Инструменты повышения производительности. Оптимизирующие компиляторы. Мониторы производительности Microsoft. Профилировщик Microsoft Visual C.		Кр №2
4	<b>Основы многопроцессорной обработки.</b> Параллельное программирование. Управление программными потоками. Многопоточное программирование. Проблемы многопоточности..	17	Кр №1 Кр №3
4	<b>Реализация многопоточности</b> средствами операционной системы. Спецификация OpenMP. Библиотека MPI.	17	Кр №1
5	<b>Общая методика создания виртуальных компьютеров.</b> Архитектура виртуальной машины	17	Кр №1
4,5	<b>Использование виртуальных машин.</b> Использование облачных технологий Системное тестирование Нагрузочное тестирование	17	Кр №1 Кр №2
1,2	<b>Коллективная разработка ПО</b> Контроль версий в программных проектах Свободно-распространяемые системы СКВ Сборка и выпуск программных проектов Средства управления сборкой	17	Кр №1 Кр №3

#### 4.5. Темы и содержание контрольных работ.

Предполагается написание магистрантами письменных трех контрольных работ. Оценка контрольных работ осуществляется с применением двухбалльной шкалы «зачтено» – «не зачтено».

По контрольной работе № 1 соответствующие оценки устанавливаются в зависимости от процента правильных ответов магистранта на тестовые задания: «зачтено» – 50% – 100%; «не зачтено» – менее 50%.

По контрольным работам № 2 и № 3 соответствующие оценки устанавливаются в зависимости от полноты и качества содержания и оформления представленных отчетов о контрольной работе, результатов демонстрации работоспособности разработанного программного обеспечения и устного собеседования с магистрантом. Оценка «зачтено» ставится, если магистрант владеет необходимыми знаниями, умениями и навыками при выполнении контрольных заданий, а разработанное им программное обеспечение является работоспособным. Оценка «не зачтено» ставится, если магистрант непоследователен в изложении результатов работ, не в полной мере владеет необходимыми умениями и навыками при выполнении контрольных заданий, а программное обеспечение либо не разработано, либо работает с ошибками, либо не обладает всеми указанными в задании функциональными возможностями

В контрольную работу № 2 включается одно задание на аналитический обзор вопросов, отраженных в лекции, и практическое задание на понимание этих вопросов, связанное с выбором тематики курсового проекта и формированием задания на курсовой проект.

В контрольную работу № 3 включается одно задание на аналитический обзор вопросов, отраженных в лекции, и практическое задание на понимание этих вопросов, связанное с формированием эксплуатационно-технического документа по ЕСПД по тематике курсового проекта.

Индивидуальное задание на курсовой проект приведет к оригинальному содержанию контрольных работ.

### **Контрольная работа 1:**

- 1 Модель жизненного цикла, использующая итерационный подход к разработке программы, называется:
  - 1) каскадной моделью;
  - 2) итерационной моделью;
  - 3) спиральной моделью;
  - 4) линейной моделью.
- 2 Каким из стандартов регламентируется жизненный цикл программы:
  - 1) ISO;
  - 2) Oracle;
  - 3) RAD;
  - 4) ГОСТ 34.
- 3 Экспериментальным выполнением разработанной программы под управлением CASE-средства является:
  - 1) апробирование;
  - 2) пилотный проект;
  - 3) тестирование;
  - 4) сопровождение.
- 4 Профили программного обеспечения применяются:
  - 1) для улучшения качества программ;
  - 2) увеличения скорости разработки программ;
  - 3) качественного применения CASE-средства;
  - 4) качественного применения стандартов.
- 5 К организационным процессам жизненного цикла ПО относится:
  - 1) процесс верификации;
  - 2) процесс управления;
  - 3) процесс аттестации;
  - 4) процесс обеспечения качества.
- 7 К промышленным технологиям проектирования программного обеспечения относится:
  - 1) технология RAD;
  - 2) технология RUP;
  - 3) технология DATA;
  - 4) технология Rational Rose.
- 8 Наименование стадии и программный документ, содержащий описание реализованного изделия это:
  - 1) проект;
  - 2) рабочий проект;
  - 3) техническое задание;
  - 4) программное задание.
- 9 Одна из частей процесса создания программы, установленная нормативными документами и заканчивающаяся выпуском проектной документации, называется:
  - 1) этапом проекта;
  - 2) стадией проекта;
  - 3) фазой проекта;

- 4) моделью проекта.
- 10 Свойство осуществлять требуемое преобразование информации при сохранении выходных решений программы в пределах допусков называется:
- 1) восстанавливаемостью ПО;
  - 2) конкретностью ПО;
  - 3) устойчивостью ПО;
  - 4) защищенностью ПО.
- 11 Программные средства, поддерживающие процессы создания и сопровождения программных продуктов, называются:
- 1) САПР-средствами;
  - 2) CASE-средствами;
  - 3) визуальными средствами;
  - 4) Oracle-средствами.
- 12 Технология, представляющая собой методологию проектирования программ, а также набор инструментальных средств, позволяющих в наглядной форме моделировать предметную область, анализировать модель на всех этапах разработки и сопровождения программных систем, называется:
- 1) RAD–технологией;
  - 2) RUP–технологией;
  - 3) CASE–технологией;
  - 4) технологией DATARUN.
- 13 Какое из приведённых средств не относится к CASE-средствам:
- 1) Silverrun;
  - 2) Oracle Designer;
  - 3) Rational Rose;
  - 4) RUP.
- 14 Фаза тестирования, выполняемая разработчиками для подтверждения, что все фрагменты правильно интегрированы в систему, а сама система работает надёжно, называется:
- 1) альфа-тестированием;
  - 2) бета-тестированием;
  - 3) анализом;
  - 4) автономным тестированием.
- 15 Контроль и испытание системы по отношению к исходным целям называется:
- 1) комплексным тестированием;
  - 2) контролем;
  - 3) испытанием;
  - 4) тестированием.
- 16 Попытка найти ошибки, выполняя программу в тестовой или моделируемой среде, называется:
- 1) комплексным тестированием;
  - 2) контролем;
  - 3) испытанием;
  - 4) тестированием.
- 17 Свойство безошибочной реализации требуемого алгоритма при отсутствии таких мешающих факторов, как ошибки входных данных, ошибки операторов, называется:
- 1) восстанавливаемостью ПО;
  - 2) конкретностью ПО;
  - 3) используемостью ПО;
  - 4) защищенностью ПО.
- 18 Методология RAD – это методология:
- 1) быстрой разработки приложений;

- 2) автоматизированной разработки приложений;
- 3) разработки по каскадной модели;
- 4) разработки без использования подпрограмм.

19 Программа, являющаяся продуктом промышленного производства, называется:

- 1) лицензионной программой;
- 2) программным изделием;
- 3) внедрённой программой;
- 4) проектным решением.

20 Операционная система Windows \* может повышать приоритет потока (называемый в этом случае .....), если поток не полностью использовал свой квант, или понижать его в противном случае.

21 В начале работы поток- писатель устанавливает его в несигнальное состояние, а все потоки- читатели выполнили вызов Wait (X) (где X - указатель на это событие) и переходят в состояние ожидания наступления этого события. Это - ...

- объект-файл
- объект-мьютекс
- объект-событие
- объект-процесс
- объект-поток

22 Выборка потока на ..... осуществляется в соответствии с принятым в данной операционной системе правилом с учетом всех существующих в данный момент потоков и процессов.

23 В основу управления потоками положена концепция дискретных состояний. Укажите состояние, которое не определено для потоков в операционной системе:

- выполнение
- синхронизация
- ожидание
- готовность
- взаимоисключение

24 Синхронизация необходима для исключения гонок и ..... при обмене данными между потоками, разделении данных, при доступе к процессору и устройствам ввода-вывода.

25 Ресурс, предоставляемый потоку во время его выполнения и допускающий его одновременное использование более чем одним потоком, называется ...

- параллельно используемым
- одновременным
- общим
- разделяемым
- комплексным
- взаимоисключающим

## Контрольная работа 2:

### 1 Аналитическая часть

1.1 Жизненный цикл программного обеспечения и процессов разработки программного обеспечения и методологии разработки программного обеспечения

1.2 Структурные и объектно-ориентированные методы анализа и проектирования программного обеспечения

1.2.1 Структурный подход в проектировании ПО:

- SADT-модели (стандарт IDEF0),
- Диаграммы «сущность-связь» (ERD),
- Диаграммы потоков данных (DFD).

1.2.2 Унифицированный язык моделирования UML:

- Диаграммы прецедентов,

- *Диаграммы классов,*
- *Диаграммы взаимодействий,*
- *Диаграммы последовательности действий,*
- *Диаграммы состояний,*
- *Компонентные диаграммы.*

1.3 Повышение производительности разрабатываемого программного обеспечения в многопроцессорных системах

1.4 Связь тестирования и качества разрабатываемого ПО

- *Тестирование производительности и стабильности ПО.*
- *Использование виртуализации для системного тестирования*

1.5 Инструментальная поддержка процесса коллективной разработки программного обеспечения

- *Основные задачи при групповой работе с проектом*
- *Контроль версий в программных проектах*
- *Сборка и выпуск программных проектов*

## **2 Практическая часть**

Формирование задания на курсовой проект по дисциплине «Современные технологии разработки программного обеспечения» с учетом темы магистерской диссертации.

Оцените требования к графическому материалу в задании на курсовой проект с учетом методов анализа и проектирования программного обеспечения.

### **Контрольная работа 3:**

#### **1 Аналитическая часть**

1.1 Жизненный цикл программного обеспечения и методологии разработки программного обеспечения с учетом жизненного цикла.

1.2 ЕСПД как одна из методологий разработки программного обеспечения

1.3 Разработайте описание математического (формализованное описание, алгоритмы), информационного (структура баз данных) и программного обеспечения (структура программного комплекса) по тематике магистерской диссертации

1.4 Определите методику тестирования программного комплекса в соответствии с выбранной тематикой

#### **2 Практическая часть**

К программному обеспечению, рассмотренному в курсовом проекте, предлагается разработать функциональный или эксплуатационный документ по ЕСПД.

Единая система программной документации (ЕСПД) - комплекс государственных стандартов, устанавливающих взаимосвязанные правила разработки, оформления и обращения программ и программной документации. В стандартах ЕСПД устанавливают требования, регламентирующие разработку, сопровождение, изготовление и эксплуатацию программ.

Документирование должно начинаться одновременно с разработкой продукта или даже раньше. В процессе разработки создаются функциональные и эксплуатационные программные документы. Программная и эксплуатационная документация используется для изготовления и сопровождения программного изделия, для его тестирования (испытания), для его эксплуатации.

Оцените содержание задания на курсовой проект, с точки зрения возможности составления требуемого эксплуатационно-технического документа.

В зависимости от выбранной тематики курсового проекта, связанной с темой магистерской диссертации, определен вид документа (таблица 4.4.1), который необходимо разработать.

Таблица 4.5.1 – Виды документов для разработки

Тема	Документ
Структурные и объектно-ориентированные методы анализа и проектирования программного обеспечения»	эксплуатационно-технической документ по ГОСТ 19.402-78 ЕСПД «Описание программы»
Повышение производительности разрабатываемого программного обеспечения за счет использования параллельных алгоритмов в многопроцессорных системах	эксплуатационно-технической документ по ГОСТ 19.504-79. ЕСПД «Руководство программиста»
Разработка информационного обеспечения программного комплекса	эксплуатационный документ по ГОСТ 19.503–79 ЕСПД «Руководство системного программиста»
Связь тестирования и качества разрабатываемого программного обеспечения	эксплуатационный документ по ГОСТ ГОСТ 19.301-79 «Программа и методика испытаний»
Инструментальная поддержка процесса коллективной разработки программного обеспечения	эксплуатационно-технической документ по ГОСТ ГОСТ 19.502-78 ЕСПД «Описание применения»

Далее приводятся требования к содержанию соответствующих документов:

— **Описание программы**

Есть пользователи, операторы, администраторы, которым приходится непосредственно работать с программой. А есть менеджеры (или в каких-то случаях ведущие специалисты), которые сами могут к программе не притрагиваться, но принимают решения относительно приобретения программы и порядка ее применения. Им необходимо понимать, что программа «умеет», и каких ресурсов она потребует. Проинформировать их об этом — задача таких документов, как описание программы и описание применения.

В описании программы и описании применения должно быть сказано:

- для чего предназначена программа, какие задачи она позволяет решать;
- какие ресурсы необходимы для выполнения программы;
- какие данные программа принимает на вход;
- что программа выдает в качестве выходных данных.

Описание программы раскрывает основные вопросы применения, структуры и функционирования программы. Приводятся основные принципы организации разработки и построения программы, общая структура и алгоритмы функционирования, структура основных компонентов программы, и содержание основных потоков данных информационного обмена с техническими средствами. Рассматриваются вопросы о входных и выходных данных, используемых в системе, конфигурации технических средств, необходимых для функционирования ПК.

В системе ЕСПД требования к содержанию и оформлению данного руководства регулируются стандартом ГОСТ 19.402-78. В соответствии с ним, документ должен содержать следующие разделы:

- 1 **Общие сведения** - обозначение и наименование программы; программное обеспечение, необходимое для функционирования программы; языки программирования, на которых написана программа.
- 2 **Функциональное назначение** - классы решаемых задач и/или назначение программы и сведения о функциональных ограничениях на ее применение.
- 3 **Описание логической структуры** - алгоритм программы; используемые методы; структура программы с описанием функций составных частей и связи между ними;

связи программы с другими программами. Описание логической структуры программы выполняют с учетом текста программы на исходном языке программирования.

- 4 **Используемые технические средства** - типы ЭВМ и устройств, которые используются при работе программы.
- 5 **Вызов и загрузка** - способы вызова программы с соответствующего носителя данных; входные точки в программу. Допускается здесь указывать сведения об использовании оперативной памяти, объем программы.
- 6 **Входные данные** - характер, организация и предварительная подготовка входных данных, формат, описание и способ кодирования входных данных.
- 7 **Выходные данные** - характер и организация выходных данных; формат, описание и способ кодирования выходных данных.

#### — **Руководства программиста**

Руководство программиста относится к эксплуатационно-технической документации и требуется в тех случаях, когда система тем или иным образом предоставляет возможность написания, редактирования или использования программного кода.

Примерами могут служить:

- библиотека функций;
- платформа или среда для разработки ПО;
- ПО с открытым кодом.

Документ должен предоставлять всю необходимую информацию для того, чтобы разработчик мог воспользоваться возможностями системы. Для решения этой задачи содержание документа может включать в себя:

- назначение, структуру входных и выходных данных программных функций;
- возможности по созданию программного кода, особенности его интерпретации и компиляции;
- синтаксические особенности используемого языка программирования;
- возможные правила и ограничения при работе с программным кодом;
- различные инструкции по работе с программой.

Список возможных тем этим не ограничивается, все зависит от особенностей конкретной системы. Надо сказать, что руководство программиста бывает очень полезно и для разработчиков системы, являясь справочником по текущей реализации логики работы ПО.

В системе ЕСПД требования к содержанию и оформлению данного руководства регулируются стандартом ГОСТ 19.504-79. В соответствии с ним, документ должен содержать следующие разделы:

- 1 **Назначение и условия применения программы**, где указывают область применения ПО и технические требования, необходимые для его работы.
- 2 **Характеристика программы**, где описывают режим работы программы, показатели скорости ее работы и другие важные для использования характеристики.
- 3 **Обращение к программе**, где указывают способы и параметры запуска программы;
- 4 **Входные и выходные данные**, где описывают формат, способ организации и другие требования к входным и выходным данным;
- 5 **Сообщения**, где приводят тексты сообщений, выдаваемых программой в различных ситуациях и действия, которые необходимо при этом предпринять.

#### — **Руководство системного программиста**

Сегодня во многих компаниях сотрудникам запрещено устанавливать на своих рабочих местах программы по своему усмотрению. Даже простые программы там ставит только **системный администратор**.

Системный администратор – это сотрудник, должностные инструкции которого подразумевают обеспечение штатной работы парка компьютерной техники, сети и программного обеспечения. Соответственно, именно этот человек будет заниматься установкой конкретного программного обеспечения на конкретных технических средствах, будь то персональный компьютер с определенными характеристиками, либо промышленный серверный комплекс.

**Руководство системного администратора** призвано максимально подробно раскрыть информацию о комплексе технических средств, необходимых для работы вашего программного обеспечения, а также описать процесс и особенности установки, настройки и поддержки работоспособности вашего ПО.

В обязанности системного администратора также входит поддержание работоспособности программ, используемых в рамках тех или иных систем. Эта деятельность может заключаться в периодической проверке логов, резервном копировании данных, замерах производительности, устранении различных технических проблем.

В ЕСПД специалист, сходный по обязанностям с современным системным администратором, называется *системным программистом*, а адресованный ему документ — **руководством системного программиста**.

В системе ЕСПД требования к содержанию и оформлению данного руководства регулируются стандартом ГОСТ 19.503–79. В соответствии с ним, документ должен содержать следующие разделы:

- 1 **Общие сведения о программе** – назначение и функции программы и сведения о технических и программных средствах, обеспечивающих выполнение данной программы.
- 2 **Структура программы** – сведения о структуре программы, ее составных частях, о связях между составными частями и о связях с другими программами.
- 3 **Настройка программы** – описание инструкций по настройке программы на условия конкретного применения (настройка на состав технических средств, выбор функций и др.).
- 4 **Проверка программы** – описание способов проверки, позволяющих дать общее заключение о работоспособности программы (контрольные примеры, методы прогона, результаты).
- 5 **Дополнительные возможности** – описание дополнительных разделов функциональных возможностей программы и способов их выбора.
- 6 **Сообщения системному программисту** – тексты сообщений, выдаваемых в ходе выполнения настройки, проверки программы, а также в ходе выполнения программы, описание их содержания и действий, которые необходимо предпринять по этим сообщениям.

#### — **Программа и методика испытаний**

Для формализации процесса испытания системы в комплект технической документации включают документ Программа и методика испытаний (ПМИ).

После формирования технического предложения и **разработки эскизного и технического проектов** наступает момент, когда необходимо четко определить, как будет решаться вопрос о соответствии системы всем требованиям, описанным в проектной документации, как будет определяться степень ее надежности, а также, что самое главное, уровень соответствия системы своему назначению. Именно для этого и предназначен документ Программа и методика испытаний (ПМИ). В этом документе содержится

описание того, что и как необходимо сделать, дабы убедиться (и убедить Заказчика) в правильности работы программы.

В системе ЕСПД требования к содержанию и оформлению данного руководства регулируются стандартом ГОСТ с ГОСТ 19.301-79. В соответствии с ним, документ должен содержать следующие разделы:

- 1 **Объект испытаний.** Здесь указывают наименование системы, ее обозначение и область применения.
- 2 **Цель испытаний.** В данном разделе указывают цель проведения тестов, или цели, если их несколько. Обычно основной целью является проверка системы на соответствие требованиям.
- 3 **Требования к программе.** Здесь перечисляют требования, описанные в Техническом задании на разработку системы, соответствие которым необходимо установить.
- 4 **Требования к программной документации.** В этом разделе указывают список документов, которые должны быть предъявлены к моменту тестирования, а также особые требования к тестам.
- 5 **Средства и порядок испытаний.** Здесь приводят список технических и программных средств, необходимых для проведения тестирования, а также порядок проведения отдельных этапов тестирования.
- 6 **Методы испытаний.** В этом разделе необходимо описать способы и методики тестирования. Тут же приводят описания проверок с указанием результатов проведения тестов (перечней тестовых примеров, контрольных распечаток тестовых примеров и т. п.). Сами тестовые примеры, графики, схемы, рисунки и т.д. для удобства использования могут быть представлены в виде приложений к документу.

– **Описание применения**

Этот документ относится к эксплуатационным документам и содержит сведения для обеспечения функционирования и эксплуатации программы. Включает сведения о назначении программы, области применения, применяемых методах, классе решаемых задач, ограничениях для применения, минимальной конфигурации технических средств.

Предназначен для пользователей, которым приходится непосредственно работать с программой. Им необходимо понимать, что программа «умеет», и каких ресурсов она потребует..

В системе ЕСПД требования к содержанию и оформлению данного руководства регулируются стандартом ГОСТ 19.503-79. В соответствии с ним, документ должен содержать следующие разделы:

- 1 **Назначение программы** - назначение, возможности программы, ее основные характеристики, ограничения, накладываемые на область применения программы.
- 2 **Условия применения** - условия, необходимые для выполнения программы (требования к необходимым для данной программы техническим средствам и другим программам, общие характеристики входной и выходной информации, а также требования и условия организационного, технического и технологического характера и т.п.).
- 3 **Описание задачи** - определения задачи и методы ее решения.
- 4 **Входные и выходные данные** - сведения о входных и выходных данных.

**5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине.**

Методические указания для обучающихся по организации самостоятельной работы по дисциплине, включая перечень тем самостоятельной работы, формы текущего контроля по дисциплине и требования к их выполнению размещены в электронной

#### **6. Фонд оценочных средств для проведения промежуточной аттестации.**

Промежуточная аттестация по дисциплине проводится в форме защиты курсового проекта и экзамена.

Экзамен предусматривает выборочную проверку освоения предусмотренных элементов компетенций и комплектуется вопросами (заданиями) двух видов: теоретический вопрос (для проверки знаний) и комплексная задача (для проверки умений и навыков).

При сдаче экзамена студент получает три вопроса из перечня вопросов, время подготовки студента к устному ответу – до 45 мин.

Пример варианта вопросов на экзамене:

##### Вариант № 1

1. Расскажите о методологии функционального моделирования SADT.
2. Перечислите инструменты повышения производительности программного обеспечения (ПО).
3. Начертите граф распределения ресурсов при возникновении простого тупика с участием трех процессов и трех ресурсов.

Результаты освоения дисциплины считаются достигнутыми, если для всех элементов компетенций достигнут пороговый уровень освоения компетенции на данном этапе – оценка «удовлетворительно».

## **7. Перечень учебных изданий, необходимых для освоения дисциплины.**

### **а) печатные издания:**

- 1 Бройдо, В. Л. Вычислительные системы, сети и телекоммуникации : учеб. пособие для вузов / В. Л. Бройдо, О. П. Ильина. – 4-е изд. – М. ; СПб. ; Н. Новгород : Питер, 2011. – 554 с.
- 2 Жадановская, Н. П. Создание приложений для операционной системы Microsoft Windows : метод. указания / Н. П. Жадановская ; СПбГТИ(ТУ). Каф. систем автоматизир. проектирования и упр. – СПб. : [б. и.], 2006. – 59 с.
- 3 Коваленко, В. В. Проектирование информационных систем : учеб. пособие для вузов / В. В. Коваленко. – М. : Форум, 2012. – 319 с.
- 4 Мельников, В. П. Информационная безопасность и защита информации : учеб. пособие для вузов / В. П. Мельников, С. А. Клейменов, А. М. Петраков ; под ред. С. А. Клейменова. – 5-е изд., стер. – М. : Academia, 2011. – 331 с.
- 5 Норенков, И. П. Автоматизированные информационные системы : учеб. пособие / И. П. Норенков. – М. : Изд-во МГТУ им. Н.Э. Баумана, 2011. – 342 с.
- 6 Советов, Б. Я. Базы данных. Теория и практика : учеб. для вузов / Б. Я. Советов, В. В. Цехановский, В. Д. Чертовской. – 2-е изд., стер. – М. : Высш. шк., 2007. – 463 с.
- 7 Тенишев, Д. Ш. Лингвистическое и программное обеспечение автоматизированных систем : учеб. пособие для вузов / Д. Ш. Тенишев ; под ред. Т. Б. Чистяковой ; СПбГТИ(ТУ). Каф. систем автоматизир. проектирования и упр. – СПб. : ЦОП «Профессия», 2010. – 403 с.

### **б) электронные учебные издания:**

- 8 Вейцман, В. М. Проектирование информационных систем : учеб. пособие / В. М. Вейцман. – СПб. ; М. ; Краснодар : Лань, 2019. – 316 с. (ЭБС «Лань»)
- 9 Гвоздева, А. В. Проектирование информационных систем. Стандартизация : учеб. пособие / Т. В. Гвоздева, Б. А. Баллод. – СПб. ; М. ; Краснодар : Лань, 2019. – 252 с. (ЭБС «Лань»)
- 10 Зубкова, Т. М. Технология разработки программного обеспечения : учеб. пособие / Т. М. Зубкова. – СПб. ; М. ; Краснодар : Лань, 2019. – 324 с.
- 11 Незнанов, А. А. Программирование и алгоритмизация : учеб. для вузов / А. А. Незнанов. – М. : Академия, 2010. – 304 с. (ЭБ)
- 12 Остроух, А. В. Проектирование информационных систем / А. В. Остроух, Н. Е. Суркова. – СПб. ; М. ; Краснодар : Лань, 2019. – 164 с. (ЭБС «Лань»)

## **8. Перечень электронных образовательных ресурсов, необходимых для освоения дисциплины.**

Учебный план, РПД и учебно-методические материалы:

<http://media.technolog.edu.ru>.

Электронно-библиотечные системы:

«Электронный читальный зал – БиблиоТех»: <https://technolog.bibliotech.ru/>;

«Лань (Профессия)»: <https://e.lanbook.com/books/>.

## **9. Методические указания для обучающихся по освоению дисциплины.**

Все виды занятий по дисциплине «Современные технологии разработки программного обеспечения» проводятся в соответствии с требованиями следующих СТП (СТО):

СТП СПбГТИ 040-02 КС УКДВ. Виды учебных занятий. Лекция. Общие требования;

СТО СПбГТИ 018-2014 КС УКДВ. Виды учебных занятий. Семинары и практические занятия. Общие требования к организации и проведению;

СТП СПбГТИ 048-2009 КС УКДВ. Виды учебных занятий. Самостоятельная планируемая работа студентов. Общие требования к организации и проведению.

Планирование времени, необходимого на изучение данной дисциплины, лучше всего осуществлять на две сессии (установочную и зимнюю), предусматривая при этом регулярное повторение пройденного материала.

Основными условиями правильной организации учебного процесса для студентов является:

- плановость в организации учебной работы;
- серьезное отношение к изучению материала;
- постоянный самоконтроль.

На занятия студент должен приходить, имея знания по уже изученному материалу.

## **10. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине.**

### **10.1. Информационные технологии.**

В учебном процессе по данной дисциплине предусмотрено использование информационных технологий:

- чтение лекций с использованием слайд-презентаций;
- взаимодействие с обучающимися посредством электронной информационно-образовательной среды.

### **10.2. Программное обеспечение.**

В учебном процессе используется лицензионное системное и прикладное программное обеспечение, приведенное в таблице 1.

Таблица 1 – Лицензионное программное обеспечение

Наименование программного продукта	Лицензия
Microsoft Windows	Подписка Azure DevTools for Teaching Subscription ID 1831112343
Microsoft Visual Studio Community	
Microsoft Access	
Microsoft Visio	
Apache OpenOffice.org	Открытая лицензия Apache License 2.0

### **10.3. Базы данных и информационные справочные системы.**

Справочно-поисковая система «Консультант-Плюс».

## **11. Материально-техническое обеспечение освоения дисциплины в ходе реализации образовательной программы.**

Для проведения занятий по дисциплине на кафедре систем автоматизированного проектирования и управления СПбГТИ(ТУ) имеется необходимая материально-техническая база, соответствующая действующим санитарным и противопожарным правилам и нормам:

Таблица 3 – Характеристика материально-технической базы

Наименование компьютерного класса кафедры	Оборудование
Класс информационных и интеллектуальных систем	40 посадочных мест. Учебная мебель, пластиковая доска. Персональные компьютеры (20 шт.): четырёхядерный процессор Intel Core i7-920 (2666

Наименование компьютерного класса кафедры	Оборудование
	МГц), ОЗУ 6 Гб; НЖМД 250 Гб; CD/DVD привод, DVD-RW; видеокарта NVIDIA GeForce GT 220 (1024 Мб); звуковая и сетевая карты, встроенные в материнскую плату. Персональные компьютеры объединены в локальную вычислительную сеть кафедры и имеют выход в сеть «Интернет».
Лекционная аудитория	56 посадочных мест. Учебная мебель. Мультимедийный проектор NEC NP41. Ноутбук Asus абј на базе процессора Intel Core Duo T2000. Мультимедийная интерактивная доска ScreenMedia.

## **12. Особенности освоения дисциплины инвалидами и лицами с ограниченными возможностями здоровья.**

Для инвалидов и лиц с ограниченными возможностями учебные процесс осуществляется в соответствии с Положением об организации учебного процесса для обучения инвалидов и лиц с ограниченными возможностями здоровья СПбГТИ(ТУ), утвержденным ректором 28.08.2014.

**Фонд оценочных средств  
для проведения промежуточной аттестации по  
дисциплине «Современные технологии разработки программного обеспечения»**

**1. Перечень компетенций и этапов их формирования.**

Индекс компетенции	Содержание	Этап формирования
ОПК-2	<b>Способность разрабатывать оригинальные алгоритмы и программные средства, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач</b>	начальный
ОПК-5	<b>Способность разрабатывать и модернизировать программное и аппаратное обеспечение информационных и автоматизированных систем</b>	начальный
ОПК-8	<b>Способность осуществлять эффективное управление разработкой программных средств и проектов</b>	начальный

## 2. Показатели и критерии оценивания компетенций на различных этапах их формирования, шкала оценивания

Код и наименование индикатора достижения компетенции	Показатели сформированности (дескрипторы)	Критерий оценивания	Уровни сформированности (описание выраженности дескрипторов)		
			«удовлетворительно» (пороговый)	«хорошо» (средний)	«отлично» (высокий)
ОПК-2.2 Создание программных средств для решения задач проектирования, обработки информации, управления с использованием современных технологий и инструментальных средств разработки программного обеспечения	<b>Называет</b> современные технологии и средства разработки программного обеспечения для решения профессиональных задач (ЗН-1).	Правильные ответы на вопросы №1-4 к экзамену	Перечисляет принципы разработки ПО в многоуровневой архитектуре с ошибками.	Перечисляет принципы разработки ПО в многоуровневой архитектуре; путается в преимуществах и недостатках мобильности и переносимости ПО.	Перечисляет принципы разработки ПО в многоуровневой архитектуре без ошибок; хорошо ориентируется в преимуществах и недостатках мобильности и переносимости ПО.
	<b>Объясняет</b> выбор современных технологий и программной среды при разработке оригинальных программных средств для решения профессиональных задач (У-1).	Правильные ответы на вопросы №5-17 к экзамену	Выбирает платформы (платформу), операционную систему (операционные системы) и определенную характеристику архитектуры (например, архитектура "файл-сервер" или "клиент-сервер") для решения профессиональных задач с ошибками.	Путается при выборе платформы (платформ), операционной системы (операционных систем) и определенных характеристик архитектуры (например, архитектура "файл-сервер" или "клиент-сервер") для решения профессиональных задач.	Способен самостоятельно осуществить выбор платформы (платформ), операционной системы (операционных систем) и определенных характеристик архитектуры (например, архитектура "файл-сервер" или "клиент-сервер") для решения профессиональных задач.
	<b>Демонстрирует</b> навыки разработки оригинальных программных средств с использованием современных технологий и инструментальных средств разработки программного			Разрабатывает структуру ПО для достижения должной производительности параллельной обработки данных с ошибками.	Путается при разработке структуры ПО для достижения должной производительности параллельной обработки данных.

Код и наименование индикатора достижения компетенции	Показатели сформированности (дескрипторы)	Критерий оценивания	Уровни сформированности (описание выраженности дескрипторов)		
			«удовлетворительно» (пороговый)	«хорошо» (средний)	«отлично» (высокий)
	обеспечения для решения профессиональных задач (Н-1).				данных.
ОПК-5.1 Разработка и модернизация проблемно-ориентированного программного обеспечения информационных и автоматизированных систем различного назначения	<b>Перечисляет</b> современное программное обеспечение информационных и автоматизированных систем (ЗН-2).	Правильные ответы на вопросы №18-23 к экзамену	Перечисляет действующие стандарты программных интерфейсов с ошибками.	Путается в действующих стандартах программных интерфейсов.	Уверенно и без ошибок перечисляет действующие стандарты программных интерфейсов.
	<b>Объясняет</b> способы модернизации программного обеспечения информационных и автоматизированных систем для решения профессиональных задач (У-2).	Правильные ответы на вопросы №24-29 к экзамену	Применяет программные интерфейсы разрабатываемом программном обеспечении с ошибками.	Путается в применении программных интерфейсов в разрабатываемом программном обеспечении.	Хорошо разбирается в применении программных интерфейсов в разрабатываемом программном обеспечении.
	<b>Демонстрирует</b> навыки разработки программного обеспечения информационных и автоматизированных систем для решения профессиональных задач (Н-2).	Правильные ответы на вопросы №30-34 к экзамену	Использует интегрированных сред разработки при создании ПО с ошибками.	Путается при использовании интегрированных сред разработки при создании ПО.	Демонстрирует навыки использования интегрированных сред разработки при создании ПО.
ОПК-8.1 Выбор и применение современных технологий и инструментальных средств при разработке приложений, баз данных, программных	<b>Перечисляет</b> основные языки программирования и работы с базами данных, операционные системы и оболочки, современные программные среды разработки информационных систем (ЗН-3).	Правильные ответы на вопросы №35-40 к экзамену	Перечисляет современные тенденции в архитектуре компьютеров - многоядерные процессоры и оценивает их влияние на архитектуру ПО с ошибками.	Путается при перечислении современных тенденций в архитектуре компьютеров - многоядерные процессоры и оценивает их влияние на архитектуру ПО.	Перечисляет современные тенденции в архитектуре компьютеров - многоядерные процессоры и оценивает их влияние на архитектуру ПО.

Код и наименование индикатора достижения компетенции	Показатели сформированности (дескрипторы)	Критерий оценивания	Уровни сформированности (описание выраженности дескрипторов)		
			«удовлетворительно» (пороговый)	«хорошо» (средний)	«отлично» (высокий)
и пользовательских интерфейсов	<b>Объясняет</b> , как применять языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для решения прикладных задач различных классов, ведения баз данных и информационных хранилищ (У-3).	Правильные ответы на вопросы №41-46 к экзамену	Имеет слабое представление об использовании потоков в роли главного программного интерфейса в многопоточном программировании	Путается при объяснении использования потоков в роли главного программного интерфейса в многопоточном программировании	Объясняет использование потоков в роли главного программного интерфейса в многопоточном программировании.
	<b>Демонстрирует</b> навыки программирования, отладки и тестирования прототипов программных комплексов (Н-3).	Правильные ответы на вопросы № 47-52 к экзамену	Выполняет разработку и отладку ПО в интегрированных средах с ошибками.	Путается при разработке и отладке ПО в интегрированных средах.	Демонстрирует навыки разработки и отладки ПО в интегрированных средах.

### 3. Типовые контрольные задания для проведения промежуточной аттестации

#### а) Вопросы для оценки сформированности элементов компетенции ОПК-2:

1. Перечислите основные характеристики программ.
2. Приведите существующую классификацию программного обеспечения.
3. Дайте определение и перечислите основные характеристики системного программного обеспечения.
4. Дайте определение и перечислите основные характеристики прикладного программного обеспечения.
5. Дайте определение и охарактеризуйте инструментарий технологии программирования.
6. Укажите документацию по сопровождению программных средств
7. Назовите виды программных документов.
8. Назовите виды эксплуатационных документов
9. Перечислите возможности для проектирования множественных прикладных сред. Общая методика создания виртуальных компьютеров.
10. Расскажите об архитектуре виртуальной машины
11. Укажите области использования виртуальных машин.
12. Что такое «управление требованиями»?
13. В чем заключается анализ проблемы?
14. Какие виды ограничений на создаваемое ПО необходимо выявить в процессе работы над требованиями?
15. Каковы существующие методы выявления требований к ПО?
16. Приведите эксплуатационные требования к ПО.
17. Перечислите функциональные требования к ПО.

#### б) Вопросы для оценки сформированности элементов компетенции ОПК-5:

18. Чем определяется выбор архитектуры ПО?
19. Расскажите об особенностях создания программного продукта.
20. Что такое жизненный цикл программного обеспечения?
21. Каковы основные свойства каскадной (итерационной) модели жизненного цикла?
22. Опишите варианты использования системы.
23. Чем описывается поведение системы?
24. Охарактеризуйте методологию функционального моделирования SADT.
25. Охарактеризуйте состав функциональной модели. Иерархия диаграмм. Типы связей между функциями.
26. Приведите примеры функциональных моделей в стандарте IDEF0.
27. Расскажите о моделировании данных. Методология IDEF1.
28. Охарактеризуйте понятие UML.
29. Перечислите основные диаграммы UML.
30. Оптимизация программного обеспечения.
31. Перечислите инструменты повышения производительности. Поясните назначение Оптимизирующие компиляторы, мониторы производительности, профилировщики.
32. Поясните основы многопроцессорной обработки.
33. Расскажите об особенностях управления программными потоками. Параллельное программирование. Проблемы многопоточности.
34. Расскажите о реализации многопоточности средствами операционной системы, спецификация OpenMP, библиотека MPI.

#### в) Вопросы для оценки сформированности элементов компетенции ОПК-8:

35. Какие виды ошибок существуют?
36. Что такое тест? Какими свойствами должен обладать тест?
37. Каковы критерии выбора тестов?
38. Дайте краткую характеристику каждому критерию выбора теста.
39. Что входит в понятие надежности ПО?
40. Какие виды отказов существуют?
41. Каковы количественные характеристики надежности программ?
42. Что представляют собой методы оценки и измерения характеристик надежности ПО?
43. Какова стоимость исправления ошибок в ПО на различных стадиях его разработки?
44. Что представляет собой нагрузочное тестирование с использованием виртуальных машин.
45. Как происходит выбор среды программирования?
46. Сравните технологию .NET с технологией Java.
47. Как производится оценка эффективности ПО на этапе эксплуатации?
48. Что такое CASE-технологии?
49. Что такое RAD-технологии?
50. Что представляют собой проектирование на основе перехода от статических моделей к реляционным БД.
51. Что представляет собой промежуточное ПО в клиент-серверных системах. Проектирование сервисных подсистем.
52. Что представляет собой проектирование распределенных программных архитектур, основанных на компонентах. Составные подсистемы и компоненты. Моделирование компонент с помощью UML.

При сдаче экзамена студент получает три вопроса из перечня, приведенного выше. Время подготовки студента к устному ответу на вопросы – до 45 мин.

#### **4. Темы курсовых проектов:**

Целью курсового проекта является получение теоретических и практических навыков по разработке, тестированию и документированию сложных программных систем на основе концепций жизненного цикла программной системы, современных подходов и методов разработки программного обеспечения в многоуровневой архитектуре программного обеспечения на их основе и методологий тестирования.

Поскольку курс «Современные технологии разработки программного обеспечения» предшествует написанию магистерской диссертации, тема курсового проекта связывается с созданием программного обеспечения по теме диссертации.

В рамках выполнения курсового проекта магистрант должен разработать программный комплекс, реализующий поставленные задачи, Провести полное тестирование. Привести контрольные примеры. Исследовать вопросы, связанные с тестирование производительности. Определить такой показатель производительности как ускорение, зависящий от количества физических потоков, организованных в параллельной реализации доступа и обмена информацией. Оценить возможность масштабирования разрабатываемой системы с учетом увеличения количества ядер современных компьютеров.

Далее приводятся примеры тематик курсового проекта.

**Тематика курсового проекта – «Структурные и объектно-ориентированные методы анализа и проектирования программного обеспечения»**

*Содержание курсового проекта:*

1. Проблемы разработки сложных программных систем
2. Структурные методы анализа и проектирования ПО (SADT (IDEF0), IDEF3, DFD)
3. Моделирование данных (ER)
4. Объектно-ориентированные методы анализа и проектирования ПО (UML)
5. Архитектура программного обеспечения
6. Обоснование выбора методов анализа и проектирования разрабатываемого программного обеспечения
7. Определение цели и задач проектирования
8. Выбор способа организации входных и выходных данных
9. Разработка структурных схем программного продукта
10. Разработка структурированного перечня функциональных и нефункциональных требований к программе
11. Разработка UML диаграммы вариантов использования с описанием: главные актёры, второстепенные актёры, предусловия, основной поток, постусловия, альтернативные потоки.
12. Разработка UML диаграммы классов, с описанием назначения всех классов и отношений между ними.
13. Разработка UML диаграммы последовательностей для вариантов использования с участием классов и выявленных для них операций.
14. Разработка UML диаграммы деятельности для описания поведения программы.
15. Тестирование проектной модели на последовательность и конечную выполнимость и тестирование классов на точное соответствие их спецификации.
16. Оформление документации по курсовому проекту (пояснительная записка, описание программы) и презентации проекта для защиты

***Тематика курсового проекта – «Повышение производительности разрабатываемого программного обеспечения за счет использования параллельных алгоритмов в многопроцессорных системах»***

*Содержание курсового проекта:*

1. Аналитический обзор интерфейсов прикладного программирования для разработки параллельных программ. Потоки как главный программный интерфейс в многопоточном программировании
2. Математические основы параллельного программирования. Характер увеличения скорости программы в результате распараллеливания
3. Особенности процесса и технологии промышленного тестирования
4. Определение цели и задач проектирования
5. Определение требований к исходной информации
6. Разработка функциональной структуры программного комплекса, включающая информационное обеспечение, математическое обеспечение, интерфейсы пользователя и разработчика.
7. Выбор и обоснование показателей и критериев эффективности системы
8. Разработка ПО с использованием базовых алгоритмов решения поставленной задачи
9. Мотивация параллельности в разрабатываемой системе
10. Разработка ПО в виде многопоточного приложения для решения поставленной задачи
11. Разработка методики тестирования многопоточного программного приложения
12. Тестирование производительности и стабильности разработанного ПО
13. Использование виртуализации для реализации нагрузочного тестирования
14. Определение такого показателя производительности как ускорение, зависящего от количества физических потоков, организованных в параллельной реализации
15. Обоснование выбора оптимального количества потоков для поставленной задачи

16. Оформление документации по курсовому проекту (пояснительная записка, руководство программиста) и презентации проекта для защиты.

**Тематика курсового проекта – «Связь тестирования и качества разрабатываемого программного обеспечения»**

*Содержание курсового проекта:*

1. Тестирование как способ обеспечения качества программного продукта
2. Различия тестирования и отладки, фазы и технология тестирования, проблемы тестирования.
3. Критерии выбора тестов: структурные, функциональные, стохастические, мутационный, оценки покрытия проекта.
4. Разновидности тестирования: модульное, интеграционное, системное, регрессионное, автоматизация тестирования, издержки тестирования.
5. Особенности процесса и технологии индустриального тестирования: планирование тестирования, подходы к разработке тестов, особенности ручной разработки и генерации тестов, автоматизация тестового цикла, документирование тестирования.
6. Системное тестирование: тестирование восстановления, тестирование безопасности, стрессовое тестирование, тестирование производительности.
7. Особенности подхода к обеспечению качества программного продукта средствами тестирования.
8. Определение цели и задач проектирования.
9. Определение требований к исходной информации.
10. Разработка функциональной структуры программного комплекса, включающая информационное обеспечение, математическое обеспечение, интерфейсы пользователя и разработчика.
11. Выбор и обоснование показателей и критериев эффективности системы.
12. Разработка программного обеспечения.
13. Разработка методики тестирования разработанной программы.
14. Тестирование производительности и стабильности разработанного программного обеспечения.
15. Использование виртуализации для реализации нагрузочного тестирования.
16. Оформление документации по проекту (пояснительная записка, программа и методика испытаний) и презентации проекта для защиты.

**Тематика курсового проекта – «Разработка информационного обеспечения программного комплекса»**

*Содержание курсового проекта:*

1. Обзор программных продуктов, представленных на рынке специализированного программного обеспечения
2. Обзор и обоснование архитектуры информационной системы (двухуровневая система или трехуровневая)
3. Обзор технологий и интерфейсов доступа к данным
4. Обоснование выбора программных средств для создания базы данных и интерфейсов пользователей
5. Определение цели и задач проектирования
6. Определение требований к исходной информации
7. Разработка функциональной структуры программного комплекса, включающая информационное обеспечение, математическое обеспечение, интерфейсы пользователя и разработчика.
8. Проектирование базы данных (концептуальная модель, инфологическая модель и даталогическая)

9. Разработка представления для отображения результатов выборки из базы данных
10. Разработка механизмов управления данными в базе данных (добавление, удаление и обновление данных)
11. Проработка технологии доступа к базе данных (определение круга пользователей базы данных и оценка возможности разграничения полномочий пользователей);
12. Организация возможности обмена данными между созданной базой данной и другими приложениями (например, офисными приложениями)
13. Обеспечение резервного копирования базы данных для обновления и восстановления информационного обеспечения
14. Разработка программного комплекса
15. Использование виртуализации для реализации системного тестирования и проверки надежности и целостности разрабатываемого информационного обеспечения
16. Оформление документации по проекту (пояснительная записка, руководство системного программиста) и презентации проекта для защиты.

**Тематика курсового проекта – «Инструментальная поддержка процесса коллективной разработки программного обеспечения»**

*Содержание курсового проекта:*

1. Проблемы разработки сложных программных систем
2. Основные задачи при групповой разработке программного обеспечения (общий доступ к файлам, сохранение истории модификации файлов, поддержка и развитие нескольких историй файла и т.д.)
3. Назначение системы контроля версий
4. Типы систем контроля версий
5. Централизованные и распределённые системы хранения версий
6. Операции в системах контроля версий
7. Обоснование выбора средств инструментальной поддержки процесса групповой разработки программного обеспечения
8. Определение цели и задач проектирования
9. Выбор способа организации входных и выходных данных
10. Разработка структурных схем программного продукта
11. Получение проекта
12. Обновление последних версий
13. Сохранение изменений
14. Добавление нового документа
15. Решение конфликтных ситуаций
16. Оформление документации по проекту (пояснительная записка, описание применения) и презентации проекта для защиты.

Для любой выбранной темы пояснительная записка к проекту должна содержать функциональную структуру программного комплекса, UML диаграмму прецедентов использования, UML диаграмму последовательности событий, UML диаграмму компонентов разработанной системы, характеристику разработанной политики защиты и разграничения прав пользователей, тестовые примеры работы программного комплекса, результаты тестирования, отражающие оценки производительности, характеристику аппаратного и программного обеспечения и разработанную программную документацию по ЕСПД.

**5. Методические материалы для определения процедур оценивания знаний, умений и навыков, характеризующих этапы формирования компетенций.**

Промежуточная аттестация по дисциплине проводится в соответствии с требованиями СТО СПбГТИ(ТУ) 016-2015 КС УКДВ. Порядок организации и проведения

зачетов и экзаменов.

По дисциплине промежуточная аттестация проводится в форме защиты курсового проекта, экзамена.

Шкала оценивания на экзамене балльная («отлично», «хорошо», «удовлетворительно», «неудовлетворительно»).

**Шаблон задания на курсовой проект  
по дисциплине «Современные технологии разработки программного обеспечения»**

МИНОБРНАУКИ РОССИИ  
федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Санкт-Петербургский государственный технологический институт  
(технический университет)»

**ЗАДАНИЕ НА КУРСОВОЙ ПРОЕКТ**

УГС	09.00.00	Информатика и вычислительная техника
Направление подготовки	09.04.01	Информатика и вычислительная техника
Программа магистратуры		Информационное и программное обеспечение автоматизированных систем
Факультет	Информационных технологий и управления	
Кафедра	Систем автоматизированного проектирования и управления	
Учебная дисциплина	<b><i>Современные технологии разработки программного обеспечения</i></b>	
Курс I		Группа
Студент	<i>Фамилия Имя Отчество</i>	

**Тема: Разработка информационного обеспечения программного комплекса**

***Исходные данные к проекту:***

- 1 Орлов, С. А. Технологии разработки программного обеспечения : учебник для вузов / С. А. Орлов, Б. Я. Цилькер – СПб. : Питер, 2012. – 608 с.
- 2 Рамбо, Дж. UML 2.0. Объектно-ориентированное моделирование и разработка / Дж. Рамбо, М. Блаха – СПб. : Питер, 2007. – 544 с.
- 3 Фаулер, М. UML. Основы. Краткое руководство по стандартному языку объектного моделирования / М. Фаулер пер. с англ. – СПб : Символ-Плюс, 2011. – 192 с.
- 4 Норенков, И. П. Автоматизированные информационные системы : учеб. пособие для вузов / И. П. Норенков. – М. : Изд-во МГТУ им. Н.Э. Баумана, 2011. – 342 с.
- 5 Литература по описанию объекта проектирования
- 6 Литература по инструментальной среде
- 7 Лекции, стандарты (по дисциплинам: модели анализа и проектирования организационных систем, проектирование программных систем)
- 8 Интернет – сайты по предметной области

**Перечень вопросов, подлежащих разработке**

***Аналитический обзор***

- 1.1 Обзор программных продуктов, представленных на рынке специализированного программного обеспечения
- 1.2 Обзор и обоснование архитектуры информационной системы
- 1.3 Обзор технологий и интерфейсов доступа к данным
- 1.3 Обоснование выбора программных средств для создания базы данных и интерфейсов пользователей

***Основная часть. Технология разработки программного комплекса***

- 2.1 Определение цели и задач проектирования

## Продолжение приложения 2

2.2 Определение требований к исходной информации

2.3 Разработка функциональной структуры программного комплекса, включающая информационное обеспечение, математическое обеспечение, интерфейсы пользователя и разработчика.

2.4 Проектирование базы данных (концептуальная модель, инфологическая модель и даталогическая) и разработка представления для отображения результатов выборки из базы данных

2.5 Разработка механизмов управления данными в базе данных (добавление, удаление и обновление данных)

2.6 Проработка технологии доступа к базе данных (определение круга пользователей базы данных и оценка возможности разграничения полномочий пользователей);

2.7 Организация возможности обмена данными между созданной базой данной и другими приложениями (например, офисными приложениями)

2.8 Обеспечение резервного копирования базы данных для обновления и восстановления информационного обеспечения

2.9 Разработка программного комплекса

2.10 Использование виртуализации для реализации системного тестирования и проверки надежности и целостности разрабатываемого информационного обеспечения

2.11 Оформление документации по проекту (пояснительная записка, руководство системного программиста) и презентации проекта для защиты.

### Перечень графического материала

1 Функциональная структура программного комплекса.

2 UML диаграмма прецедентов использования.

3 UML диаграмма компонентов разработанной системы.

5 Инфологическая модель данных

4 Даталогическая модель с указанием связей между таблицами

5 Разработанная политика защиты и разграничения прав пользователей

6 Тестовые примеры работы программного комплекса.

### Требования к аппаратному и программному обеспечению

*Аппаратное обеспечение:* характеристика аппаратного обеспечения (технологическое оборудование, технические средства автоматизации, ЭВМ, периферийные устройства).

*Программное обеспечение:* характеристика программного обеспечения (системного, прикладного).

Консультант по работе \_\_\_\_\_

Дата выдачи задания

Дата представления проекта к защите

Заведующий кафедрой		
	(подпись, дата)	(инициалы, фамилия)
Лектор, должность		
	(подпись, дата)	(инициалы, фамилия)
Руководитель проекта, должность		
	(подпись, дата)	(инициалы, фамилия)
Руководитель магистранта, должность		
	(подпись, дата)	(инициалы, фамилия)
Задание принял к выполнению		
	(подпись, дата)	(инициалы, фамилия)