


ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ  
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«КАМЧАТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»  
(ФГБОУ ВО «КамчатГТУ»)

Факультет информационных технологий

Кафедра «Информационные системы»

УТВЕРЖДАЮ

Декан факультета ИТ

  
И.А. Рычка

«12» марта 2020 г.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ**

**«Тестирование программного обеспечения»**

направление подготовки  
09.03.04 Программная инженерия  
(уровень бакалавриата)

Петропавловск-Камчатский,  
2020

Рабочая программа разработана в соответствии с ФГОС ВО по направлению подготовки 09.03.04 Программная инженерия, профиль «Разработка программно-информационных систем», и учебного плана ФГБОУ ВО «КамчатГТУ».

Составители рабочей программы:

Доцент кафедры ИС



---

С.В. Чебанюк

Доцент кафедры ИС

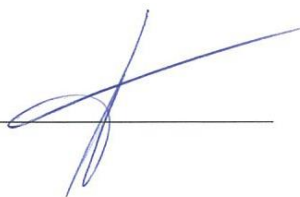


---

И.А. Рычка

Рабочая программа рассмотрена на заседании кафедры «Информационные системы» Протокол №8 от «12» марта 2020 года.

Заведующий кафедрой  
«12» марта 2020 г.



---

И.Г. Проценко

## 1. Цели и задачи учебной дисциплины

Целью освоения дисциплины «Тестирование программного обеспечения» является ознакомление студентов с основными видами и методами тестирования программного обеспечения.

Задачами изучения дисциплины «Тестирование программного обеспечения» является

- получение студентами знаний о теоретических основах технологии тестирования;
- приобретение студентами навыков документирования технологии тестирования;
- приобретение студентами навыков создания собственных тест-кейсов;
- освоение современных инструментов тестирования;
- получение опыта тестирования компьютерных программ.

## 2. Требования к результатам освоения дисциплины

Процесс изучения дисциплины направлен на формирование следующих компетенций:

- Способность обосновать принимаемые проектные решения, осуществлять постановку и выполнение экспериментов по проверке их корректности и эффективности (ПКС-6).

Таблица – Планируемые результаты обучения по дисциплине, соотнесенные с установленными в программе бакалавриата индикаторами достижения компетенций

Код компетенции	Наименование компетенции	Код и наименование индикатора достижения	Планируемый результат обучения по дисциплине	Код показателя освоения
ПКС-6	Способность обосновать принимаемые проектные решения, осуществлять постановку и выполнение экспериментов по проверке их корректности и эффективности	ИД-3 <sub>ПКС-6</sub> владеет навыками по постановке и выполнению экспериментов по проверке их корректности и эффективности	Знать: – технологии и инструменты тестирования программных продуктов, принципы управления тестированием.	З(ПКС-6)1
			Уметь: – проектировать и проводить тесты для разработанного приложения, составлять тестовые таблицы и планы тестирования.	У(ПКС-6)1
			Владеть: – навыками отладки программ на алгоритмических языках программирования, использования различных методов ручного и автоматического тестирования.	В(ПКС-6)1

## 3. Место дисциплины в структуре образовательной программы

В структуре образовательной программы «Тестирование программного обеспечения» является дисциплиной части, формируемая участниками образовательных отношений.

Курс дисциплины «Тестирование программного обеспечения» продолжает цикл дисциплин разработки программных средств, опирается на знания в области информационных технологий и программирования, полученные на дисциплинах «Теория систем и системный анализ», «Теория принятия решений», «Разработка и анализ требований», «Теория графов», «Вычислительная математика», «Исследование операций»

и во время прохождения технологической (проектно-технологической) практики.

Знания, полученные обучающимися в процессе изучения дисциплины «Тестирование программных приложений», позволят им выполнять на высоком уровне лабораторные и практические работы, курсовое и дипломное проектирование. Дисциплина «Тестирование программных приложений» является базовой дисциплиной для прохождения производственной (преддипломной) практики и выполнения дипломного проектирования.

#### 4. Содержание дисциплины

##### 4.1 Тематический план дисциплины

Наименование разделов и тем	Всего часов / з.е.	Аудиторные занятия	Контактная работа по видам учебных занятий			Самостоятельная работа	Формы текущего контроля <sup>1</sup>	Итоговый контроль знаний по дисциплине
			Лекции	практические занятия	Лабораторные работы			
1	2	3	4	5	6	7	8	9
Очная форма обучения								
<b>Раздел 1. Тестирование ПО в ЖЦ ПО</b>	<b>50</b>	<b>24</b>	<b>10</b>	<b>0</b>	<b>14</b>	<b>26</b>	Опрос	
Основные понятия и определения технологии тестирования ПО	14	6	2		4	8	Опрос, РЗ	
Тестирования в жизненном цикле ПО	16	8	4		4	8	Опрос, РЗ	
Документирование тестирования	20	10	4		6	10	Опрос, РЗ	
<b>Раздел 2. Автоматизация тестирования и отладки ПО</b>	<b>0</b>	<b>0</b>					Опрос	
Нагрузочное тестирование	0	0					Опрос, РЗ	
Автоматизированное функциональное тестирование	<b>58</b>	<b>26</b>	<b>10</b>		<b>16</b>	<b>32</b>	Опрос, РЗ	
Методологии разработки программных средств	15	7	3		4	8	Опрос, РЗ	
Методы отладки программного обеспечения	14	6	2		4	8	Опрос, РЗ	
<b>Экзамен</b>	<b>36</b>							<b>36</b>
<b>Всего</b>	<b>144/4</b>	<b>50</b>	<b>20</b>	<b>0</b>	<b>30</b>	<b>58</b>		<b>36</b>
Заочная форма обучения								
<b>Раздел 1. Тестирование ПО в ЖЦ ПО</b>	<b>51</b>	<b>10</b>	<b>2</b>	<b>0</b>	<b>8</b>	<b>41</b>	Опрос	
Основные понятия и определения технологии тестирования ПО	6	1	1			5	Опрос, РЗ	
Тестирования в жизненном цикле ПО	23	5	1		4	18	Опрос, РЗ	
Документирование тестирования	22	4			4	18	Опрос, РЗ	
<b>Раздел 2. Автоматизация тестирования и отладки ПО</b>	<b>0</b>	<b>0</b>					Опрос	
Нагрузочное тестирование	0	0					Опрос, РЗ	
Автоматизированное функциональное тестирование	<b>84</b>	<b>10</b>	<b>2</b>		<b>8</b>	<b>74</b>	Опрос, РЗ	
Методологии разработки программных средств	21	3	1		2	18	Опрос, РЗ	
Методы отладки программного обеспечения	21	1	1			20	Опрос, РЗ	
<b>Экзамен</b>	<b>20</b>	<b>2</b>			<b>2</b>	<b>18</b>		
<b>Всего</b>	<b>144/4</b>	<b>20</b>	<b>4</b>	<b>0</b>	<b>16</b>	<b>115</b>		<b>9</b>

<sup>1</sup>Т – тестирование, КС – кейс-стади, РЗ – решение задач, ПЗ – практические задания, Д - доклады

## **4.2 Содержание дисциплины**

Понятия правильности и надежности программ. Роль документирования в проверке правильности ПО. Стандарты качества ПО. Тестирование и верификация. Место тестирования в жизненном цикле ПО. Основные принципы и стадии тестирования, связь тестирования и качества разрабатываемого программного обеспечения. Стратегии тестирования.

Принципы, стратегии и этапы тестирования. Ручное тестирование и автоматизация процесса тестирования. Уровни и типы тестирования. Регрессионное тестирование.

Стандарты документирования ПО. Проектная, общесистемная, пользовательская и внутренняя документация. Документирование и анализ ошибок. Сертификация систем качества. Документирование тестирования.

Отладка программного обеспечения: классификация ошибок, методы отладки, методы и средства получения дополнительной информации об ошибке, общая методика отладки.

### **Раздел 1. Тестирование ПО в ЖЦ ПО**

#### **Тема 1.1** Основные понятия и определения технологии тестирования ПО

Качество программного средства. Правильность и надежность ПО.

Роль документирования в проверке правильности ПО. Стандарты качества ПО. Тестирование, верификация и валидация. Цели тестирования. Место тестирования в жизненном цикле ПО.

Обеспечение функциональности и надежности программного средства. Защитное программирование

#### **Тема 1.2** Тестирования в жизненном цикле ПО

Эволюция программного средства. Жизненный цикл и процессы этапа разработки программного обеспечения. Этапы тестирования. Стратегия тестирования. Классификация ошибок. Градация серьезности дефекта. Градация приоритета дефекта. Уровни тестирования. Принципы тестирования.

Тестирование методами черного и белого ящиков. Виды и типы тестирования.

Системное и регрессионное тестирование. Автоматизация тестирования. Инструменты отладки программного обеспечения. Методы отладки программного обеспечения.

Стандарты в тестировании.

V-образная модель обобщенного ЖЦ ТП. Планирование тестирования ПО. Циклы тестирования ПО: полный цикл тестирования и частный цикл тестирования.

#### **Тема 1.3** Документирование тестирования

Управление тестированием.

Тестовая документация назначение, структура, особенности. Формирование и отслеживание реализации плана тестирования.

Техника граничных значений. Таблица принятия решений. Метод уникальных пар.

Рабочая и проектная документация тестировщика. Тест план. Чек-лист. Наборы тестовых сценариев. Баг-репорт. Отчет о тестировании.

Инструменты тестирования. Автокликеры. Postman. SoapUI. JMetр. Selenium.

Особенности тестирования веб-приложений

Системы документирования дефектов в области ПО.

#### **Лабораторная 1.1**

Цель: Запись текстов программ на алгоритмическом языке высокого уровня.

*Рассматриваемые вопросы:*

1. правила хорошего стиля программирования.
2. тест-план.

3. Запись текстов программ на алгоритмическом языке высокого уровня.

*Задания:*

- составить тест-план тестирования.
- написать программу на конкретном алгоритмическом языке программирования для решения конкретной задачи, используя приемы и методы программирования.

#### ***Лабораторная 1.2***

Цель: 1) изучение классификации тестов по видам, типам и областям; 2) получение навыков определения тестов для различных видов, типов и областей тестирования на примерах настольных или web-приложений.

*Рассматриваемые вопросы:*

1. Классификация тестов.
2. Типы и области тестирования.

*Задания:*

- выполнить тестирование приложения или сервиса согласно варианту работы.
- указать, какие тесты необходимы для покрытия различных видов, типов и областей тестирования.

#### ***Лабораторная 1.3***

Цель: тестирование программ методами «белого ящика».

*Рассматриваемые вопросы:*

- Подходы к тестированию.
- Метод покрытия операторов. Метод покрытия решений (покрытия переходов). Метод покрытия условий. Метод комбинаторного покрытия условий.

*Задания:*

- построить алгоритм решения задачи в виде схемы программы
- написать программу, реализующую заданный преподавателем алгоритм обработки данных.
- записать тесты, которые позволят пройти по выбранным путям алгоритма.
- протестировать разработанную программу

#### ***Лабораторная 1.4***

Цель: тестирование программ методами «черного ящика».

*Рассматриваемые вопросы:*

- Оформление документации процесса тестирования.
- Классификация характерных ошибок.
- Тестовый сценарий, кейс-тест.
- Тестовые случаи.

*Задания:*

- разработать набор тестовых сценариев (как позитивных, так и негативных) для программы.
- составить модель состояний для любой своей разработки ПС.
- разработать набор тестовых сценариев.
- протестировать выбранное по полученному набору сценариев

**Самостоятельная работа по разделу.** Работа с конспектом лекций и рекомендованной литературой. Защита лабораторных работ.

## **Раздел 2. Автоматизация тестирования и отладки ПО**

### ***Тема 2.1*** Нагрузочное тестирование

Терминология нагрузочного тестирования. Цели и этапы нагрузочного тестирования. Обзор программ нагрузочного тестирования веб-сервисов

### ***Тема 2.2*** Автоматизированное функциональное тестирование

Преимущества и недостатки. Применение автоматизации. Уровни автоматизации

тестирования. Архитектура тестов.

**Тема 2.3** Методологии разработки программных средств  
Инструменты автоматизации тестирования.

**Тема 2.4** Методы отладки программного обеспечения  
Отладка (debugging). Трудоемкость процесса отладки. Машинные методы отладки  
Принципы и виды отладки программного средства. Заповеди отладки  
программного средства. Автономная отладка программного средства. Комплексная  
отладка программного средства.

Средства и методы отладки. Инструментальные средства отладки.

Метод "грубой силы". Метод индукции. Метод дедукции. Инверсное  
прослеживание логики программы.

### **Лабораторная 2.1**

Цели: Тестирование и анализ кода в NetBeans IDE Features.

*Рассматриваемые вопросы:*

- Тесты.
- Платформа тестирования.
- Анализ кода

*Задания:*

- создавать тесты с помощью мастеров и шаблонов для популярных платформ тестирования JUnit и TestNG
- выполнить анализ кода.
- получить статический анализ кода в редакторе Java IDE NetBeans.

**Самостоятельная работа по разделу.** Работа с конспектом лекций и рекомендованной литературой. Защита лабораторных работ.

## **5 Учебно-методическое обеспечение для самостоятельной работы обучающихся**

### **5.1. Внеаудиторная самостоятельная работа студентов**

В целом внеаудиторная самостоятельная работа студента при изучении курса включает в себя следующие виды работ:

- проработка (изучение) материалов лекций;
- чтение и проработка рекомендованной основной и дополнительной литературы;
- подготовка к практическим занятиям;
- поиск и проработка материалов из Интернет-ресурсов, научных публикаций;
- выполнение домашних заданий в форме практических заданий, докладов и рефератов;
- выполнение курсового проектирования;
- подготовка презентаций для иллюстрации результатов курсового проектирования, докладов;
- подготовка к текущему(индивидуальные опросы) и итоговому (промежуточная аттестация) контролю знаний по дисциплине (экзамен).

Основная доля самостоятельной работы студентов приходится на подготовку к лабораторным занятиям, тематика которых полностью охватывает содержание курса. Самостоятельная работа по подготовке к лекционным занятиям предполагает умение работать с первичной информацией.

*Самостоятельная работа по разделу 1:*

Работа с конспектом лекций и рекомендованной литературой (1-3 и

дополнительная).

Подготовка материалов к контрольному опросу по изученным темам, практических занятиях, диалогах с преподавателем и участниками проверки знаний дисциплинарного модуля.

*Самостоятельная работа по разделу 2:*

Работа с конспектом лекций и рекомендованной литературой (1-3 и дополнительная).

Подготовка материалов к контрольному опросу по изученным темам, практических занятиях, диалогах с преподавателем и участниками проверки знаний дисциплинарного модуля.

## **6 Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине**

Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине «Разработка программных приложений» представлен в приложении к рабочей программе дисциплины и включает в себя:

- перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы;
- описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания;
- типовые контрольные задания или материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций;
- методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций.

### **Вопросы для проведения промежуточной аттестации по дисциплине (экзамен)**

1. Качество программного средства. Правильность и надежность ПО.
2. Роль документирования в проверке правильности ПО.
3. Тестирование, верификация и валидация. Цели тестирования. Место тестирования в жизненном цикле ПО
4. Этапы тестирования. Стратегия тестирования.
5. Классификация ошибок. Градация серьезности дефекта. Градация приоритета дефекта.
6. Уровни тестирования. Принципы тестирования
7. Тестирование методами черного и белого ящиков.
8. Виды и типы тестирования.
9. Системное и регрессионное тестирование.
10. Автоматизация тестирования.
11. Инструменты отладки программного обеспечения.
12. Методы отладки программного обеспечения.
13. Управление тестированием.
14. Тестовая документация назначение, структура, особенности.
15. Формирование и отслеживание реализации плана тестирования.
16. Техника граничных значений.
17. Таблица принятия решений.
18. Метод уникальных пар.
19. Рабочая и проектная документация тестировщика. Тест план. Чек-лист. Наборы



тестовых сценариев. Баг-репорт. Отчет о тестировании.

20. Инструменты тестирования. Автокликеры. Postman. SoapUI. JMetр. Selenium.
21. Особенности тестирования веб-приложений
22. Системы документирования дефектов в области ПО.
23. Нагрузочное тестирование. Цели и этапы нагрузочного тестирования.
24. Автоматизированное функциональное тестирование. Преимущества и недостатки. Применение автоматизации. Уровни автоматизации тестирования.
25. Методологии разработки программных средств.
26. Методы отладки программного обеспечения. Метод "грубой силы". Метод индукции. Метод дедукции. Инверсное прослеживание логики программы.

## **7 Рекомендуемая литература**

### ***7.1 Основная литература***

1. Вендров А.М. Проектирование программного обеспечения экономических информационных систем: учебник - М. Финансы и статистика, 2000 (2002). - 352 с
2. Вендров А.М. Практикум по проектированию программного обеспечения экономических информационных систем: учеб. пособие. - М. Финансы и статистика, 2004 (2006). - 192 с.

### ***7.2 Дополнительная литература:***

3. Соммервилл, Иан. Инженерия программного обеспечения, 6-е издание.: Пер. с англ. – М.: Издательский дом Вильямс, 2002. – 624 с.
4. Стандартизация разработки программных средств : [Учеб. пособие для вузов по специальности 351400 "Прикладная информатика (в экономике)"] / В. А. Благодатских, В. А. Волнин, К. Ф. Посакалов ; Под ред. О. С. Разумова. - Москва : Финансы и статистика, 2003. - 284 с.
5. Иванова Г.С., Технология программирования: Учебник для вузов. – М.: Изд-во МГТУ им. Н.Э Баумана, 2002. – 320 с.

## **8. Перечень ресурсов информационно-телекоммуникационной сети «Интернет»**

1. Научная электронная библиотека eLIBRARY.RU – крупнейший российский информационный портал в области науки, технологии, медицины и образования, содержащий рефераты и полные тексты : [Электронный ресурс]. - Режим доступа URL:[www.elibrary.ru](http://www.elibrary.ru).
2. Российское образование. Федеральный портал: [Электронный ресурс]. - Режим доступа URL: <http://www.edu.ru>.
3. Software Engineering Body of Knowledge (SWEBOOK) [Электронный ресурс] // The IEEE Computer Society. – USA, Washington, (2001-) – Режим доступа URL: <https://www.computer.org/education/bodies-of-knowledge/software-engineering> (дата обращения: 15.01.2019).
4. Стандарты и регламенты [Электронный ресурс] // Федеральное агентство по техническому регулированию и метрологии РФ – М.: Режим доступа URL: <https://www.gost.ru/portal/gost/home/standarts> (дата обращения: 15.01.2019).
5. Электронный фонд правовой и нормативно-технической документации [Электронный ресурс] // М.: АО «Кодекс». – Режим доступа URL: <http://docs.cntd.ru/> (дата обращения: 15.01.2019).
6. Microsoft Docs - Документация Майкрософт для пользователей, разработчиков и ИТ-специалистов – [Электронный ресурс] // Microsoft, М.: (2006-). – Режим доступа URL:

<https://docs.microsoft.com/ru-ru/> (дата обращения: 15.01.2019).

7. Docs & Support NetBeans – Документация Apache – [Электронный ресурс] // Apache: (2006-). – Режим доступа URL: [https://netbeans.org/features/java/testing\\_ru.html](https://netbeans.org/features/java/testing_ru.html) (дата обращения: 15.01.2019).

8. Документации, обзоры, публикации по продуктам – [Электронный ресурс] // Компания «Интерфейс», М.: (2006-). – Режим доступа URL: <http://www.interface.ru/iservices/catalog.asp?catId=160> (дата обращения: 15.01.2019).

9. Распределённая система управления версиями Git. – [Электронный ресурс] // Режим доступа URL: <https://git-scm.com/> (дата обращения: 15.01.2019).

10. Липаев В.В. Программная инженерия: Комплекс учебников и монографий. – [Электронный ресурс] // Виртуальный компьютерный музей, М.: (2010-). – Режим доступа URL: <http://www.computer-museum.ru/books/lipaev/> (дата обращения: 15.01.2019).

11. Липаев В.В. Тестирование компонентов и комплексов программ. Учебник. – М.: СИНТЕГ, 2010. – 400 с. – [Электронный ресурс] // Виртуальный компьютерный музей, М.: (2010-). – Режим доступа URL: [http://www.computer-museum.ru/books/lipaev/lip\\_testirovanie\\_komp.pdf](http://www.computer-museum.ru/books/lipaev/lip_testirovanie_komp.pdf) (дата обращения: 15.01.2019).

1. Курсы для приобретения знаний и навыков, необходимых для успешной современной карьеры. // Oracle – Режим доступа URL: <https://academy.oracle.com/ru/solutions-curriculum-full.html> (дата обращения: 15.01.2019).

2. Компонентный подход в программировании : курс [Электронный ресурс] // сост. Кулямин В. – М.: НОУ ИНТУИТ, 2011. Режим доступа URL: <https://www.intuit.ru/studies/courses/64/64/info> (дата обращения: 15.01.2019).

3. Программирование на Java: курс [Электронный ресурс] // сост. Вязовик Н. – М.: НОУ ИНТУИТ, 2011. Режим доступа URL: <https://www.intuit.ru/studies/courses/16/16/info> (дата обращения: 15.01.2019).

4. Углубленное программирование на Java : курс [Электронный ресурс] // сост. Чибриков В. – М.: НОУ ИНТУИТ, 2011. Режим доступа URL: <https://www.intuit.ru/studies/courses/3711/953/info> (дата обращения: 15.01.2019).

5. Программирование на языке высокого уровня C# : курс [Электронный ресурс] // сост. Павловская Т. – М.: НОУ ИНТУИТ, 2011. Режим доступа URL: <https://www.intuit.ru/studies/courses/629/485/info> (дата обращения: 15.01.2019).

6. Методы и средства инженерии программного обеспечения : курс [Электронный ресурс] // сост. Лаврищева Е., Петрухин В. – М.: НОУ ИНТУИТ, 2011. Режим доступа URL: <https://www.intuit.ru/studies/courses/2190/237/info> (дата обращения: 15.01.2019).

7. Язык программирования Java и среда NetBeans : курс [Электронный ресурс] // сост. Монахов В. – М.: НОУ ИНТУИТ, 2011. Режим доступа URL: <https://www.intuit.ru/studies/courses/569/425/info> (дата обращения: 15.01.2019).

8. Верификация программного обеспечения: курс [Электронный ресурс] // сост. Налютин Н.Ю., Сеницын С.В. – М.: НОУ ИНТУИТ, 2011. Режим доступа URL: <https://www.intuit.ru/studies/courses/1040/209/info> (дата обращения: 15.01.2019).

9. Объектно-ориентированное программирование и программная инженерия: курс [Электронный ресурс] // сост. Meyer, Bertrand. – М.: НОУ ИНТУИТ, 2011. Режим доступа URL: <https://www.intuit.ru/studies/courses/2309/609/info> (дата обращения: 15.01.2019).

10. Технологии командной разработки программного обеспечения информационных систем : курс [Электронный ресурс] // сост. Долженко А. – М.: НОУ ИНТУИТ, 2011. Режим доступа URL: <https://www.intuit.ru/studies/courses/4806/1054/info> (дата обращения: 15.01.2019).

11. Технологии разработки современных информационных систем на платформе Microsoft.NET: курс [Электронный ресурс] // сост. Павлова Е. – М.: НОУ ИНТУИТ, 2011. Режим доступа URL: <https://www.intuit.ru/studies/courses/1162/285/info> (дата обращения: 15.01.2019).

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

Методика преподавания данной дисциплины предполагает чтение лекций, проведение практических занятий, групповых и индивидуальных консультаций по отдельным вопросам дисциплины. Предусмотрена самостоятельная работа студентов, а также прохождение аттестационных испытаний промежуточной аттестации (экзамен).

Лекции посвящаются рассмотрению наиболее важных концептуальных вопросов: основным вопросам программной инженерии, инструментам и методам программной инженерии; навыками составления технической документации на различных этапах жизненного цикла информационной системы; разработки программного обеспечения.

Целью проведения практических занятий является закрепление знаний студентов, полученных ими в ходе изучения дисциплины на лекциях и самостоятельно. Во время практических занятий выполняются лабораторные работы; на них разбираются конкретные ситуации по изучаемой теме. Для подготовки к занятиям практического типа студенты выполняют проработку рабочей программы, уделяя особое внимание целям и задачам, структуре и содержанию дисциплины, конспектирование источников и работу с конспектом лекций, просмотр рекомендуемой литературы.

При изучении дисциплины используются интерактивные методы обучения, такие как:

### **1. Лекция:**

– лекция-визуализация — подача материала осуществляется средствами технических средств обучения с кратким комментированием демонстрируемых визуальных материалов (презентаций).

### **2. Практические занятия:**

– лабораторные работы — это вид учебной работы, в рамках которого осуществляется тот или иной эксперимент, направленный на получение результатов, имеющих значение с точки зрения успешного освоения студентами учебной программы.

## **10 Курсовой проект (работа)**

Учебным планом не предусмотрено выполнение курсовой работы по дисциплине.

## **11 Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационно-справочных систем**

### **11.1 Перечень информационных технологий, используемых при осуществлении образовательного процесса**

- электронные образовательные ресурсы, представленные в п. 8 рабочей программы;
- использование слайд-презентаций;
- изучение межгосударственных стандартов ЕСПД на официальном сайте Росстандарта;
- интерактивное общение с обучающимися и консультирование посредством электронной почты.

### **11.2 Перечень программного обеспечения, используемого при осуществлении образовательного процесса**

При освоении дисциплины используется лицензионное программное обеспечение:

- пакет Microsoft Office;
- Microsoft Visio;
- Microsoft Project;

- Microsoft Visual Studio;
- Java.

### ***11.3 Перечень информационно-справочных систем***

- справочная правовая система Консультант-плюс <http://www.consultant.ru/online>
- справочно-правовая система по законодательству Российской Федерации Гарант <http://www.garant.ru/online>

## **12 Материально-техническое обеспечение дисциплины**

Для проведения занятий лекционного типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации используются учебные аудитории 7-206, 7-401, 7-402, 7-501 с комплектом учебной мебели.

Для самостоятельной работы обучающихся, в том числе для курсового проектирования, используются кабинеты 7-401, 7-402, 7-501, 7-520; каждый кабинет оборудован:

- комплектом учебной мебели,
- компьютерами с доступом в информационно-телекоммуникационную сеть «Интернет» и в электронную информационно-образовательную среду организации,
- техническими средствами обучения для представления учебной информации: аудиторная доска, мультимедийное оборудование (ноутбук, проектор), наглядными пособиями.