

**Федеральное государственное автономное образовательное учреждение  
высшего образования  
«Санкт-Петербургский политехнический университет Петра Великого»**

---

УТВЕРЖДАЮ  
Директор ИКНК  
\_\_\_\_\_ Д.П. Зегжда  
«17» июня 2025 г.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)**

**«Основы процесса разработки качественного программного продукта и его метрология»**

Разработчик	Высшая школа программной инженерии
Направление (специальность) подготовки	09.03.04 Программная инженерия
Наименование ООП	09.03.04_01 Технология разработки и сопровождения качественного программного продукта
Квалификация (степень) выпускника	<b>бакалавр</b>
Образовательный стандарт	<b>СУОС</b>
Форма обучения	<b>Очная</b>

СОГЛАСОВАНО	Соответствует СУОС
Руководитель ОП	Утверждена протоколом заседания
_____ А.В. Петров	высшей школы "ВШПИ" от «21» мая 2024 г. № №1

РПД разработал:  
Старший преподаватель Л.П. Котлярова

# **1. Цели и планируемые результаты изучения дисциплины**

## **Цели освоения дисциплины**

1. Дисциплина «Основы процесса разработки качественного программного продукта и его метрология» является дисциплиной по выбору вариативной части профессионального цикла для направления бакалавриата 09.03.04 «Программная инженерия» и содержит теоретическую информацию вместе с практикумом и курсовым проектированием.
2. Целью изучения дисциплины является формирование понимания совокупности процессов современных методологий создания качественных программных продуктов, выработки умения ориентироваться в существующих и появляющихся новых методологиях с тем, чтобы обоснованно выбирать методологии для промышленной разработки программных продуктов в зависимости от целей разработки, используя при этом метрическую информацию. Важной задачей дисциплины является приобретение практических навыков определения модели процессов разработки программного учебного проекта, критериев эффективности этапов разработки, освоение элементов планирования программных проектов с учетом метрических показателей в рамках курсового проектирования.

## **Результаты обучения выпускника**

<b>Код</b>	<b>Результат обучения (компетенция) выпускника ООП</b>
<b>ПК-11</b>	<b>Способен проводить тестирование и верификацию программного обеспечения</b>
ИД-3 ПК-11	Использует современный инструментарий для фиксации дефекта и отслеживает его состояние в соответствии с жизненным циклом дефекта
<b>ПК-12</b>	<b>Способен выполнять разработку, модификацию и тестирование программного обеспечения в рамках согласованных процессов программного проекта</b>
ИД-1 ПК-12	Выполняет необходимые действия по прогнозированию и оценке отдельных работ проекта, связанных с разработкой и модификацией программного обеспечения
ИД-2 ПК-12	Определяет возможные задержки выполнения работ проекта, определяет факторы, влияющие на задержку и информирует руководство проекта или организации
ИД-3 ПК-12	Выполняет необходимые действия по прогнозированию и оценке отдельных работ проекта, связанных с тестированием программного обеспечения

## **Планируемые результаты изучения дисциплины**

### **знания:**

- Знает основные метрики для слежения за результатами процесса тестирования и основные понятия, связанные с инструментами хранения и обработки записей об ошибках
- Знает принципы оценки сроков выполнения работ проекта, связанных с разработкой или модификацией программного обеспечения
- Знает принципы формулирования критериев выполнения работ, связанных с разработкой и модификацией программного обеспечения
- Знает принципы оценки сроков выполнения работ проекта, связанных с тестированием программного обеспечения

### **умения:**

- Умеет настраивать параметры инструментов среды разработки для обеспечения измерения покрытия исполняемого кода тестами, а также соотносить тесты и требования, тесты и исходный код
- Умеет составлять прогнозы выполнения работ проекта, связанных с разработкой или модификацией программного обеспечения (на примере конкретного проекта)
- Умеет отслеживать ход выполнения работ, связанных с разработкой и модификацией программного обеспечения, и определять необходимость эскалирования проблем
- Умеет составлять прогнозы выполнения работ проекта, связанных с тестированием программного обеспечения (на примере конкретного проекта)

### **навыки:**

- Владеет навыком работы с выбранным инструментом для записей об ошибках в программном продукте и навыком применения методов подсчета и анализа метрик, связанных с измерением покрытия программного кода тестами для конкретного проекта
- Владеет навыками работы с одним из инструментов для прогнозирования сроков и трудоемкости работ разработки программного обеспечения
- Владеет навыками определения факторов, влияющих на задержку работ проекта по отношению к первоначальным прогнозам
- Владеет навыками работы с инструментами, применяемыми при процессе тестирования программного обеспечения

## **2. Место дисциплины в структуре ООП**

В учебном плане дисциплина «Основы процесса разработки качественного программного продукта и его метрология» относится к модулю «Модуль цифровых компетенций» / «Технологии и процессы разработки программного обеспечения».

Изучение дисциплины базируется на результатах освоения следующих дисциплин:

- Архитектура программных систем
- Основы проектной деятельности
- Введение в профессиональную деятельность

### **3. Распределение трудоёмкости освоения дисциплины по видам учебной работы и формы текущего контроля и промежуточной аттестации**

#### **3.1. Виды учебной работы**

Виды учебной работы	Трудоемкость по семестрам
	Очная форма
Лекционные занятия	14
Практические занятия	8
Самостоятельная работа	37
Часы на контроль	16
Промежуточная аттестация (экзамен)	11
Курсовое проектирование	22
<b>Общая трудоемкость освоения дисциплины</b>	108, ач
	3, зет

#### **3.2. Формы текущего контроля и промежуточной аттестации**

Формы текущего контроля и промежуточной аттестации	Количество по семестрам
	Очная форма
<b>Текущий контроль</b>	
Контрольные, шт.	2
Курсовые работы, шт.	1
<b>Промежуточная аттестация</b>	
Экзамены, шт.	1

### **4. Содержание и результаты обучения**

#### **4.1 Разделы дисциплины и виды учебной работы**

№ раздела	Разделы дисциплины, мероприятия текущего контроля	Очная форма		
		Лек, ач	Пр, ач	СР, ач

1.	Введение в процессы разработки качественного программного обеспечения	1	0	1
2.	Современные методологии разработки ПО			
2.1.	Традиционные методологии разработки ПО на примерах водопадной, инкрементальной, Rational Unified Process и CMMI моделях	2	0	1
2.2.	Особенности разработки ПО при использовании методологий Agile (гибкие, быстрые методы); жизненный цикл проектов, принципы Agile разработки; обзор методологий Agile и их распространение	2	1	1
2.3.	Обзор технологической цепочки процессов Agile на примере метода Scrum	2	1	1
2.4.	Принципы бережливой разработки ПО. Применение метрических данных для управления процессом разработки и качества ПО. Встраивание качества в программный продукт	1	1	1
2.5.	Проблемы при внедрении Agile методов и практик. Преимущества, ограничения, области применения методологий Agile. Метрики по применению Agile методов в промышленном производстве ПО	1	1	1
2.6.	Контрольная работа №1	0	1	1
2.7.	Курсовая работа	0	0	22
3.	Основы планирования разработки программным проектом и управления рисками			
3.1.	Обзор планирования разработки программных продуктов; стратегическое и краткосрочное планирование, процессы и цели планирования на примере CMMI методологии	1	0	2
3.2.	Введение в управление рисками; количественный и качественный анализ рисков; планирование процесса управления рисками; мониторинг и обработка рисков	1	1	1
3.3.	Контрольная работа №2	0	1	1
4.	Методы проверки кода, отслеживание проблем			
4.1.	Формальные инспекции: фазы проведения, роли, особенности процесса. Облегченные методики обзора кода: описание процесса, преимущества и недостатки каждой методики	1	0	1
4.2.	Отслеживание проблем: отчеты о проблемах, управление проблемами, классификация проблем, связывание проблем, исправлений и тестов	1	1	1

5.	Технологии коллективной разработки ПО			
5.1.	Принципы коллективной разработки ПО, Wiki технология	1	0	2
<b>Итого по видам учебной работы:</b>		14	8	37
Экзамены, ач				16
<b>Часы на контроль, ач</b>				16
<b>Курсовое проектирование</b>				22
<b>Промежуточная аттестация (экзамен)</b>				11
<b>Общая трудоёмкость освоения: ач / зет</b>				108 / 3

## 4.2. Содержание разделов и результаты изучения дисциплины

Раздел дисциплины	Содержание
<b>1. Введение в процессы разработки качественного программного обеспечения</b>	Введение в процессы разработки качественного программного обеспечения (ПО). Программная инженерия - определения, цели. Программная инженерия как дисциплина обучения; перечень областей знаний, входящих в ПИ, место данной дисциплины в этом перечне и связь дисциплины с остальными – основными и смежными – областями знаний. Стремительный рост программных продуктов и областей их внедрения; основные проблемы, связанные с разработкой ПО.
<b>2. Современные методологии разработки ПО</b>	
<b>2.1. Традиционные методологии разработки ПО на примерах водопадной, инкрементальной, Rational Unified Process и CMMI моделях</b>	Обзор наиболее употребительных моделей жизненного цикла разработки ПО (водопадная, инкрементная, спиральная Боэма, итеративная). Основные сведения о широко распространенных ресурсоёмких методологиях разработки ПО, таких как Rational Unified Process и CMMI. Терминология, используемая при описании моделей жизненного цикла разработки ПО, специфические определения для моделей RUP, CMMI и др. Краткая характеристика достоинств, недостатков и ограничений каждой из моделей.
<b>2.2. Особенности разработки ПО при использовании методологий Agile (гибкие, быстрые методы); жизненный цикл проектов, принципы Agile разработки; обзор методологий Agile и их распространение</b>	Основные понятия и определения для Agile-проектов. Принципы Agile разработки. Обзор наиболее распространенных Agile методов (экстремальное программирование, ASD, Scrum и др.). Лучшие практики отдельных методов.
<b>2.3. Обзор технологической цепочки процессов Agile на примере метода Scrum</b>	Основные понятия и определения для Agile Scrum. Особенности процессов разработки, перечень задач, входы и выходы в виде проектных артефактов каждого из процессов, описание собираемых метрик для управления разработкой ПО в Scrum-проекте.

<p><b>2.4. Принципы бережливой разработки ПО. Применение метрических данных для управления процессом разработки и качества ПО. Встраивание качества в программный продукт</b></p>	<p>Краткая характеристика каждого принципа бережливого производства ПО (Lean Software Development). Бережливое производство на примере компании Google. Внесение изменений в процесс Agile-проекта на основании анализа метрических данных. Разбор известных метрик Agile-проектов. Анализ качества создаваемого программного продукта. Постоянная интеграция программного продукта, постоянное тестирование (Test Driven Development).</p>
<p><b>2.5. Проблемы при внедрении Agile методов и практик. Преимущества, ограничения, области применения методологий Agile. Метрики по применению Agile методов в промышленном производстве ПО</b></p>	<p>Обзор известных проблем при внедрении Agile методов в проектах. Анализ преимуществ, ограничений и недостатков этих методов. Области применения Agile методов. Анализ метрик по применению Agile методов в промышленном производстве ПО, используя доступные источники информации.</p>
<p><b>2.6. Контрольная работа №1</b></p>	<p>Контрольная работа по отдельным темам разделов 1 и 2, где в частности разбираются вопросы: описание особенностей отмеченных процессов разработки программного продукта, пояснения перечня задач, объяснение входов и выходов каждого процесса.</p>
<p><b>2.7. Курсовая работа</b></p>	<p>Курсовая работа проводится по предложенным вариантам для закрепления учебного материала по разделам 2, 3 и 4, а также для развития профессиональных умений и повышения творческого потенциала студентов. Курсовая работа выполняется студентами с использованием методических указаний, а также консультируясь с преподавателем относительно отдельных этапов работы.</p>
<p><b>3. Основы планирования разработки программным проектом и управления рисками</b></p>	
<p><b>3.1. Обзор планирования разработки программных продуктов; стратегическое и краткосрочное планирование, процессы и цели планирования на примере CMMI методологии</b></p>	<p>Особенности долгосрочного и краткосрочного планирования проектов. Примеры инструментов планирования и контроля изменений. Различия в планировании для Agile проектов по сравнению с проектами, разработанными в рамках водопадной, RUP и CMMI моделей. Специфика определений, связанная с планированием проектов разработки ПО с использованием различных методологий. Планирование в курсовой работе: формирование планового списка задач по методу Scrum (Product Backlog), назначение / изменение приоритетов задач, анализ невыполненных задач одного цикла разработки и перенос их для выполнения в последующих циклах.</p>

<b>3.2. Введение в управление рисками; количественный и качественный анализ рисков; планирование процесса управления рисками; мониторинг и обработка рисков</b>	<p>Отличительные признаки проектных рисков; психология риска; приемлемость риска; жизненный цикл риска. Разница в управлении рисками и управлении проектом.</p> <p>Риски в курсовой работе: определение рисков для учебного проекта, их ранжирование в соответствии с тяжестью последствий их наступления для проекта, регулярное отслеживание (мониторинг), присваивание рискам приоритетов, создание упрощенного плана управления рисками проекта.</p>
<b>3.3. Контрольная работа №2</b>	<p>Контрольная работа по отдельным темам раздела 3, где в частности разбираются вопросы: идентификация рисков и описание их атрибутов, планирование задач в проектах, использующих Scrum методологию.</p>
<b>4. Методы проверки кода, отслеживание проблем</b>	
<b>4.1. Формальные инспекции: фазы проведения, роли, особенности процесса.</b>  <b>Облегченные методики обзора кода: описание процесса, преимущества и недостатки каждой методики</b>	<p>Общие сведения об обзورах и инспекциях как методов обеспечения качества ПО. Детальный разбор процесса формальных инспекций по методу M.Fagan, включая описание фаз процесса, ролей участников, преимуществ и недостатков данного метода. Обзор существующих в промышленном производстве ПО облегченных методов обзора кода и документов, разбор их преимуществ и недостатков. Инструменты специального назначения.</p>
<b>4.2. Отслеживание проблем: отчеты о проблемах, управление проблемами, классификация проблем, связывание проблем, исправлений и тестов</b>	<p>Обзор процесса отслеживания найденных проблем в программном проекте. Содержание отчета о проблемах и сопутствующая информация. Управление проблемами, базы данных для хранения записей о проблемах. Возможность проследить связь между проблемами, исправлениями и тестами.</p>
<b>5. Технологии коллективной разработки ПО</b>	
<b>5.1. Принципы коллективной разработки ПО, Wiki технология</b>	<p>Wiki технология как средство коллективной разработки, хранения, структуризации информации. Изучение базовой функциональности TWiki как одного из инструментов, поддерживающего Wiki. При использовании дистанционного обучения данный раздел выполняется в часы самостоятельной работы на одном из доступном для студентов wiki-сайте.</p>

## 5. Образовательные технологии

В изучении дисциплины используются такие образовательные технологии, как лекции, практические занятия. При этом используются следующие аспекты современных образовательных технологий: лекционный материал подается с применением презентаций;

предполагается интерактивное обсуждение отдельных тем курса; предусмотрено промежуточное аудиторное или онлайн тестирование по темам курса с целью контроля уровня знаний студентов. Часть практических занятий реализуются в интерактивной форме с элементами индивидуального обучения и коллективного разбора результатов заданий. При необходимости в отдельные периоды времени применяется электронное обучение, дистанционные образовательные технологии.

## **6. Лабораторный практикум**

Не предусмотрено

## **7. Практические занятия**

№ раздела	Наименование практических занятий (семинаров)	Трудоемкость, ач
		Очная форма
1.	Характерные особенности модели процессов [X] создания программного обеспечения с пояснением для разработки какого типа программных продуктов рекомендуется применять данную модель. В качестве [X] предлагается рассмотреть одну из следующих моделей: V-образную модель разработки ПО, Rational Unified Process (RUP), CMMI, итеративную модель разработки ПО, одну из Agile методологий	1
2.	Метрические данные для управления процессом разработки и качества ПО. Анализ преимуществ и недостатков формальных инспекций и облегченных методик обзора кода. Применение одной из методик обзора кода для учебного примера.	1
3.	Процессы и цели планирования на примере одной-двух ключевых процессных областей CMMI методологии	1
4.	Процесс отслеживание проблем в ПО: отчеты о проблемах: управление проблемами: классификация проблем: связывание проблем, исправлений и тестов.	1
5.	Освоение Wiki технологии для коллективной разработки, хранения, структуризации данных и информации. Изучение основных возможностей инструмента TWiki как одного из инструментов, реализующих Wiki технологию (при использовании дистанционных образовательных технологий, данное задание выполняется в часы самостоятельной работы).	1
6.	Контрольные работы (2 шт.) с вопросами по пройденным темам, включая обсуждение отдельных ошибок и предложений улучшения процессов разработки ПО.	2
7.	Занятие с разбором отдельных пунктов курсовой работы и выборочным анализом промежуточных результатов отдельных студентов с целью предотвратить типовые ошибки в работах обучающихся	1
<b>Итого часов</b>		8

## **8. Организация и учебно-методическое обеспечение самостоятельной работы**

Для закрепления знаний, умений и получения новых навыков работы с гибкими моделями разработки программного обеспечения проводится курсовое проектирование. Именно на него расходуется основная часть времени для самостоятельной работы. Курсовая работа выполняется по одному из вариантов заданий, предложенных преподавателем.

Для реализации курсовой работы будет использоваться теоретический материал, полученный студентами во время самостоятельной работы, лекций и практических занятий. Процесс выполнения курсового задания и правила оформления отчета описаны в методических указаниях для курсовой работы. Выполнение курсовой работы проводится в часы самостоятельных занятий для закрепления учебного материала, а также для развития профессиональных умений и повышения творческого потенциала студентов. Обсуждение промежуточных результатов работы и консультации по выполнению отдельных пунктов работы проводится на практическом занятии и в виде вопросов-ответов по мере выполнения курсовой работы.

Примерное распределение времени самостоятельной работы студентов

Вид самостоятельной работы	Примерная трудоемкость, ач
	Очная форма
<b>Текущая СР</b>	
работа с лекционным материалом, с учебной литературой	7
опережающая самостоятельная работа (изучение нового материала до его изложения на занятиях)	0
самостоятельное изучение разделов дисциплины	1
выполнение домашних заданий, домашних контрольных работ	3
подготовка к лабораторным работам, к практическим и семинарским занятиям	1
подготовка к контрольным работам, коллоквиумам	2
<b>Итого текущей СР:</b>	14
<b>Творческая проблемно-ориентированная СР</b>	
выполнение расчётно-графических работ	0
выполнение курсового проекта или курсовой работы	22
поиск, изучение и презентация информации по заданной проблеме, анализ научных публикаций по заданной теме	1
работа над междисциплинарным проектом	0
исследовательская работа, участие в конференциях, семинарах, олимпиадах	0
анализ данных по заданной теме, выполнение расчётов, составление схем и моделей на основе собранных данных	0
<b>Итого творческой СР:</b>	23
<b>Общая трудоемкость СР:</b>	37

## 9. Учебно-методическое обеспечение дисциплины

### 9.1. Адрес сайта курса

<https://dl.spbstu.ru/course/view.php?id=1856>

## **9.2. Рекомендуемая литература**

### **Основная литература**

<b>№</b>	<b>Автор, название, место издания, издательство, год (годы) издания</b>	<b>Год изд.</b>	<b>Источник</b>
1	Котлярова Л.П. Организация и планирование разработки качественного программного продукта, 2012. URL: <a href="http://elib.spbstu.ru/dl/local/2469.pdf">http://elib.spbstu.ru/dl/local/2469.pdf</a>	2012	ЭБ СПбПУ
2	Брукс Ф. Мифический человеко-месяц или как создаются программные системы : Санкт-Петербург: Символ-Плюс, 2003.	2003	ИБК СПбПУ

### **Дополнительная литература**

<b>№</b>	<b>Автор, название, место издания, издательство, год (годы) издания</b>	<b>Год изд.</b>	<b>Источник</b>
1	A guide to the project management body of knowledge (PMBOK guide): Newtown Square, Pa.: PMI, 2008.	2008	ИБК СПбПУ

### **Ресурсы Интернета**

1. CMMI for Development, Version 1.3: <http://www.sei.cmu.edu/library/abstracts/reports/10tr033.cfm>
2. ГОСТ Р ИСО/МЭК 12207-2010 Информационная технология. Системная и программная инженерия. Процессы жизненного цикла программных средств: <http://protect.gost.ru/document1.aspx?control=31&baseC=6&page=0&month=1&year=-1&search=%D0%93%D0%9E%D0%A1%D0%A2%20%D0%A0%20%D0%98%D0%A1%D0%9E/%D0%9C%D0%AD%D0%9A%2012207&id=176990>
3. Основы Scrum метода: <https://www.scrumalliance.org/about-scrum/overview>
4. AgileDays - конференции по гибкому управлению: <https://agiledays.ru/>

## **9.3. Технические средства обеспечения дисциплины**

Свободно распространяемая версия TWiki, инсталлируется на институтский сервер (или доступный студентам wiki-сайт, где можно редактировать временные тексты). Свободно распространяемые программные инструменты для статической и динамической проверки кода. Наличие MS Office 2016 (и более новые версии) или LibreOffice или P-7 Офис, а также браузеров, совместимых с версией TWiki.

## **10. Материально-техническое обеспечение дисциплины**

Для успешного проведения практических занятий необходимо использование компьютерного класса, имеющего не менее 10 компьютеров, оснащенных необходимым программным обеспечением и локальной сети с сервером на 8-10 рабочих мест. Для проведения лекций и консультаций необходим класс с наличием мультимедийного проектора.

При использовании дистанционных образовательных технологий и электронного обучения необходимы программные инструменты, одобренные приказами университета и института.

## **11. Критерии оценивания и оценочные средства**

### **11.1. Критерии оценивания**

Для дисциплины «Основы процесса разработки качественного программного продукта и его метрология» формой аттестации является экзамен. Дисциплина реализуется с применением системы индивидуальных достижений.

#### **Текущий контроль успеваемости**

Максимальное значение персонального суммарного результата обучения (ПСРО) по приведенной шкале - 100 баллов

Максимальное количество баллов приведенной шкалы по результатам прохождения двух точек контроля - 80 баллов.

Подробное описание правил проведения текущего контроля с указанием баллов по каждому контрольному мероприятию и критериев выставления оценки размещается в СДО в навигационном курсе дисциплины.

#### **Промежуточная аттестация по дисциплине**

Максимальное количество баллов по результатам проведения аттестационного испытания в период промежуточной аттестации – 20 баллов приведенной шкалы.

Промежуточная аттестация по дисциплине проводится в соответствии с расписанием.

**1. Форма проведения экзамена:** устное собеседование по билетам; каждый билет содержит 3 вопроса. Количество вопросов может быть уменьшено с учетом портфолио студента (результаты выполненных заданий и контрольных в течение семестра). Портфолио включает самостоятельные работы и тесты, выполненные в течение семестра. Получение положительных оценок за все (или не менее 75%) предусмотренных программой заданий/тестов является основанием проведения промежуточной аттестации по дисциплине.

**Критерии оценки:** максимальная оценка за прохождение промежуточной аттестации – 5.

Оценка и общие требования к оценке:

5 - даны полные, аргументированные ответы с примерами на все вопросы билета и на дополнительные вопросы, при их наличии; продемонстрировано глубокое понимание теоретического материала.

4 - ответы полные, но имеются незначительные неточности, а также незначительные затруднения с ответами на дополнительные вопросы по теме.

3 - ответы фрагментарны, основные положения раскрыты поверхностно, затрудняется ответить на дополнительные вопросы по теме вопросов билета.

2 - не раскрыты ключевые аспекты вопросов и дополнительных вопросов, обнаружены серьезные пробелы в базовом теоретическом материале; данная оценка выставляется также, если студент после начала экзамена отказался его сдавать или нарушил правила сдачи экзамена.

Дополнительные параметры:

Время ответа: 15-18 мин. для экзамена.

Разрешается использование:

- личных конспектов,
- материалов дисциплины, предоставленных преподавателем.

Запрещается:

- подсказки третьих лиц.

При использовании дистанционного режима зачет и экзамен принимаются с использованием средств Интернета способом, приближенном к обычному.

**2. Форма проведения защиты курсовой работы:** на отдельном занятии по расписанию.

**Критерии оценки:** для оценки "зачтено" необходимо выполнить все пункты задания курсовой работы и получить определённое количество баллов.

Отдельные пункты работы оцениваются в баллах на основании полноты и качества выполнения данных пунктов:

качество выполнения пунктов 1-3 – бизнес-требования, ресурсы проекта и вехи проекта (от 0 до 3 баллов);

качество выполнения пунктов 4-5 – график и риски программного проекта (от 0 до 5 баллов);

качество выполнения пунктов 6-7 – критерии сдачи/приемки и предлагаемые метрики проекта (от 0 до 4 баллов);

качество оформления итогового отчета курсовой работы (от 0 до 3 баллов).

Оценка «зачтено» за курсовую работу выставляется обучающемуся, если он набрал от 8 до 15 баллов, и за каждый оцениваемый набор пунктов получил не менее 2х баллов.

Результаты промежуточной аттестации, определяются на основе баллов, набранных в рамках применения, СИД

Баллы по приведенной шкале в рамках применения СИД (ПСРО+ ПА)	Оценка по результатам промежуточной аттестации
	Экзамен/диф.зачет/зачет
0 - 60 баллов	Неудовлетворительно/не зачтено
61 - 75 баллов	Удовлетворительно/зачтено
76 - 89 баллов	Хорошо/зачтено
90 и более	Отлично/зачтено

## 11.2. Оценочные средства

Оценочные средства по дисциплине представлены в фонде оценочных средств, который является неотъемлемой частью основной образовательной программы и размещается в электронной информационно-образовательной среде СПбПУ на портале etk.spbstu.ru

## 12. Методические рекомендации по организации изучения дисциплины

Методические рекомендации по каждому разделу приводятся в слайдах лекционного материала в качестве дополнительных слайдов, со ссылкой на рекомендуемую литературу для самостоятельного изучения, и предварительного списка вопросов или заданий для самостоятельной работы.

Успешное изучение дисциплины требует выполнения всех учебных заданий и практических заданий, ознакомления с основной и дополнительной литературой.

Разбор наиболее часто встречающихся ошибок при выполнении самостоятельных заданий происходит с помощью преподавателя в часы аудиторных занятий или в виде вопросов-ответов.

Рекомендуется изучать материал по мере его подачи на занятиях. Если тема или отдельные вопросы по теме вызывают затруднения, необходимо постараться найти ответы на них, используя рекомендуемую литературу. Если студенту самостоятельно не удалось разобраться в материале, необходимо сформулировать вопросы и обратится за помощью к преподавателю на ближайшей занятии или задать вопрос, используя предоставленный преподавателем способ

коммуникации с ним. Обучающемуся необходимо регулярно отводить время для повторения пройденного материала, проверяя свои знания, умения и навыки по контрольным вопросам. Практические занятия и курсовая работа составляют важную часть профессиональной подготовки студентов. Они направлены на подтверждение теоретических положений и формирование учебных и профессиональных практических умений.

Для более хорошего усваивания материала, особенно плохо успевающими студентами, предлагается добавить дополнительные часы (1 час в неделю) для факультативных занятий.

### **13. Адаптация рабочей программы для лиц с ОВЗ**

Адаптированная программа разрабатывается при наличии заявления со стороны обучающегося (родителей, законных представителей) и медицинских показаний (рекомендациями психолого-медицинской-педагогической комиссии). Для инвалидов адаптированная образовательная программа разрабатывается в соответствии с индивидуальной программой реабилитации.