

Межгосударственное образовательное учреждение высшего образования
«Белорусско-Российский университет»

УТВЕРЖДАЮ

Первый проректор Белорусско-Российского
университета


Ю.В. Машин

«17» 06 2022г.

Регистрационный № УД-090304/Б. 1. В. 7 1р

ТЕХНОЛОГИИ КОМАНДНОЙ РАЗРАБОТКИ ПРИЛОЖЕНИЙ

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Направление подготовки 09.03.04 Программная инженерия

Направленность (профиль) Разработка программно-информационных систем

Квалификация Бакалавр

	Форма обучения
	Очная
Курс	3
Семестр	5, 6
Лекции, часы	16
Лабораторные занятия, часы	50
Курсовая работа, семестр	6
Экзамен, семестр	5
Контактная работа по учебным занятиям, часы	66
Самостоятельная работа, часы	114
Всего часов / зачетных единиц	180/5

Кафедра-разработчик программы: Программное обеспечение информационных технологий

Составитель: К.В. Захарченков, к.т.н.

Могилев, 2022

Рабочая программа составлена в соответствии с федеральным государственным образовательным стандартом высшего профессионального образования по направлению подготовки 09.03.04 – “Программная инженерия” (уровень бакалавриата), утвержденным приказом № 920 от 19.09.2017г. и учебным планом, утвержденным Рег. № 090304-5 от 25.03.2022.

Рассмотрена и рекомендована к утверждению кафедрой «Программное обеспечение информационных технологий» «08» апреля 2022 г., протокол № 10.

Зав. кафедрой ПОИТ


В. В. Кутузов

Одобрена и рекомендована к утверждению Научно-методическим советом Белорусско-Российского университета

«15» июня 2022 г., протокол № 7.

Зам. председателя
Научно-методического совета


С.А. Сухоцкий

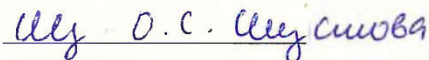
Рецензент:

С.В. Миренков, начальник управления информационных технологий ОАО «Лента»

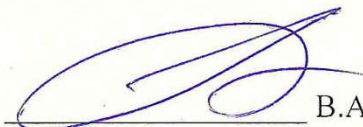
(И.О. Фамилия, должность, ученая степень, ученое звание рецензента)

Рабочая программа согласована:

Ведущий библиотекарь


О.С. Шустова

Начальник учебно-методического
отдела


В.А. Кемова

1 ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

1.1 Цель учебной дисциплины

Основной целью преподавания дисциплины является обзор современных технологий разработки программного обеспечения.

Дисциплина «Технологии командной разработки приложений» обеспечивает приобретение знаний и умений в соответствии с государственным образовательным стандартом, содействует формированию мировоззрения и системного мышления.

1.2 Планируемые результаты изучения дисциплины

В результате освоения учебной дисциплины студент должен **знать:**

- методы проектирования и производства программного продукта, принципы построения, структуры и приемы работы с инструментальными средствами, поддерживающими создание программного обеспечения;

- методы организации работы в коллективах разработчиков программного обеспечения; основные положения метрологии программных продуктов, принципы построения, проектирования и использования средств для измерений характеристик и параметров программ, программных систем и комплексов.

уметь:

- применять методы проектирования и производства программного продукта, принципы построения, структуры и приемы работы с инструментальными средствами, поддерживающими создание программного обеспечения;

владеть:

- различными методами решения задач;
- средствами инженерии программного обеспечения.

1.3 Место учебной дисциплины в системе подготовки студента

Дисциплина относится к блоку 1 «Дисциплины (модули)». Часть Блока 1, формируемая участниками образовательных отношений.

Перечень учебных дисциплин, изучаемых ранее, усвоение которых необходимо для изучения данной дисциплины:

- программирование;
- базы данных;
- информатика.

Перечень учебных дисциплин (циклов дисциплин), которые будут опираться на данную дисциплину:

- архитектура программных систем;
- современные системы программирования;
- интегрированные информационные системы предприятий.

Кроме того, знания, полученные при изучении дисциплины на лабораторных занятиях, будут использованы при прохождении производственной практики, а также при подготовке выпускной квалификационной работы.

1.4 Требования к освоению учебной дисциплины

Освоение данной учебной дисциплины должно обеспечивать формирование следующих компетенций:

Коды формируемых компетенций	Наименования формируемых компетенций
ПК-2	Владение методами контроля проекта и готовностью осуществлять контроль версий

ПК-10	Владение навыками использования различных технологий разработки программного обеспечения
ПК-12	Владение стандартами и моделями жизненного цикла

2 СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Вклад дисциплины в формирование результатов обучения выпускника (компетенций) и достижение обобщенных результатов обучения происходит путём освоения содержания обучения и достижения частных результатов обучения, описанных в данном разделе.

2.1 Содержание учебной дисциплины

Номер тем	Наименование тем	Содержание	Коды формируемых компетенций
1	Процессы жизненного цикла программных систем	Понятие жизненного цикла программных систем Основные процессы ЖЦ ПС Вспомогательные процессы ЖЦ ПО Организационные процессы ЖЦ ПО Взаимосвязь между процессами ЖЦ ПО Модели и стадии ЖЦ ПО Стратегии конструирования ПО Рациональный унифицированный процесс MSF (MicrosoftSolutionFramework)	ПК-2 ПК-12
2	Разработка и анализ требований к программной системе	Понятие требования. Классификации требований Свойства требований Процесс анализа требований Контекст задачи анализа требований Выявление требований	ПК-2 ПК-12
3	Разработка спецификаций программной системы	Понятие спецификаций требований к ПС Принципы спецификаций требований к ПС Модели спецификаций: модели состояний, модели поведения, модели изменения состояний Полнота спецификаций Ясность спецификаций Согласованность спецификаций Верифицируемость спецификаций Осуществимость требований спецификаций	ПК-2 ПК-12
4	Проблемы создания сложных программных систем и основные пути их решения	Проблема сложности программных систем Сложность описания поведения отдельных подсистем Проблемы создания сложных программных систем Методы создания сложных программных систем Структурная декомпозиция Алгоритмическая декомпозиция Нисходящее проектирование Восходящее проектирование Комбинированное проектирование Проблемы создания модулей Структурное проектирование модуля	ПК-2 ПК-12
5	Планирование процесса разработки программной системы	Планирование действий на этапах разработки ПС и интеграционном этапе Планирование работ каждого этапа жизненного цикла программной системы определение окружения жизненного цикла Применение стандартов разработки ПС Составление плана разработки ПС Проработка и проверка плана разработки ПС	ПК-2 ПК-12

6	Организация процесса разработки сложных программных систем	<p>Методология разработки сложных ПС</p> <p>Основные принципы организации работы над проектом</p> <p>Методология Rational Unified Process (RUP)</p> <p>Выбор методов и средств моделирования и проектирования ПС</p> <p>Выбор средств разработки ПС</p> <p>Комплексное тестирование сложных ПС</p> <p>Управление проектами и портфелями</p> <p>Управление требованиями к ПС</p> <p>Управление конфигурациями и изменениями ПС</p> <p>Управление выполнением проекта и документированием</p> <p>IBM SoftwareDevelopmentPlatform – новый подход к разработке программного обеспечения</p> <p>Инструментальные средства IBM Rational</p> <p>IBM Rational – состав пакетов и краткое описание продуктов</p>	ПК-10 ПК-12
7	Критерии, методы и модели оценки качества программных систем	<p>Модель оценки качества программных систем (ПС)</p> <p>Характеристика показателей качества ПС</p> <p>Метрики качества ПС</p> <p>Стандартная оценка значений показателей качества</p> <p>Управление качеством ПС</p> <p>Модели оценки надежности ПС</p> <p>Основные понятия в проблематике надежности ПС</p> <p>Классификация моделей надежности ПС</p> <p>Марковские и пуассоновские модели надежности ПС</p>	ПК-10 ПК-12
8	Испытания, внедрение, эксплуатация и сопровождение программных систем	<p>Проблемы внедрения программных систем и их решение</p> <p>Эксплуатация программных систем</p> <p>Место сопровождения в жизненном цикле программных систем</p> <p>Модификация, усовершенствование и коррекция программных систем в процессе сопровождения</p> <p>Средства и приемы сопровождения</p> <p>Планирование и организация сопровождения.</p> <p>Эксплуатационная документация</p> <p>Инструментальные средства, поддерживающие этап сопровождения</p> <p>Стиль программирования, ориентированный на эффективную поддержку этапа сопровождения</p>	ПК-2 ПК-10

2.2 Учебно-методическая карта учебной дисциплины

№ недели	Лекции (наименование тем)	Часы	Практические (семинарские) занятия	Часы	Лабораторные занятия	Часы	Самостоятельная работа, часы	Форма контроля знаний	Баллы (max)
Модуль 1									
1	1. Процессы жизненного цикла программных систем	2			Лр №1 – Разработка и анализ требований к программной системе	2	3		
2					Лр №1 – Разработка и анализ требований к программной	4	3		

				системе					
3	2. Разработка и анализ требований к программной системе	2		Лр №1 – Разработка и анализ требований к программной системе	2	3	ЗЛР	15	
4				Лр №2 - Спецификации программной системы	4	3			
5	3. Разработка спецификаций программной системы	2		Лр №2 - Спецификации программной системы	2	3			
6				Лр №2 - Спецификации программной системы	4	3	ЗЛР	15	
7	4. Проблемы создания сложных программных систем и основные пути их решения	2		Лр №3 – Испытания программных систем	2	3			
8				Лр №3 – Испытания программных систем	4	3	ПКУ	30	
Модуль 2									
9	5. Планирование процесса разработки программной системы	2		Лр №3 – Испытания программных систем	2	3	ЗЛР	10	
10				Лр №4 - Использование систем автоматизации разработки программ	4	3			
11	6. Организация процесса разработки сложных программных систем	2		Лр №4 - Использование систем автоматизации разработки программ	2	3			
12				Лр №4 - Использование систем автоматизации разработки программ	4	3	ЗЛР	10	
13	7. Критерии, методы и модели оценки качества программных систем	2		Лр №5 - Компонентное программирование	2	2			
14				Лр №5 - Компонентное программирование	4	1			
15	8. Испытания, внедрение, эксплуатация и сопровождение программных систем	2		Лр №5 - Компонентное программирование	2	1			
16				Лр №5 - Компонентное программирование	4	1			
17				Лр №5 - Компонентное	2	1	ЗЛР ПКУ	10 30	

				программиро- вание				
1-17	Выполнение курсового проекта (работы)*					36		
18-20						36	ПА (экзамен)	40
	Итого	16				50	114	100

Принятые обозначения:

Текущий контроль –

ЗЛР – защита лабораторных работ;

ПКУ – промежуточный контроль успеваемости.

ПА - Промежуточная аттестация.

Итоговая оценка определяется как сумма текущего контроля и промежуточной аттестации и соответствует баллам:

Экзамен

Оценка	Отлично	Хорошо	Удовлетворительно	Неудовлетворительно
Баллы	87-100	65-86	51-64	0-50

2.3 Требования к курсовому проекту (курсовой работе)

Целью курсового проектирования является приобретение практических навыков разработки программного обеспечения.

Примерная тематика курсовых проектов (работ) представлена в приложении хранится на кафедре.

Содержание курсовой работы включает две части:

- 1) теоретическая часть – обзор по теме проектирования, исследование актуальных вопросов в данной области, постановка задач, обоснование принятого решения;
- 2) практическая – реализация программной системы и ее отладка.

Примерная тематика курсовых проектов представлена в приложении 1.

На выполнение курсового проекта (работы) отводится 36 часов.

Разбивка этапов курсовой работы, определение количества минимальных и максимальных баллов за каждый из них производится преподавателем. Примерный перечень этапов выполнения курсовой работы и количества баллов за каждый из них представлен в таблице.

№	Этап выполнения	Минимум	Максимум
1	Описание предметной области	6	10
2	Проектирование программного модуля	6	10
3	Разработка документации программного модуля	9	15
4	Разработка программного модуля	9	15
5	Прототипирование интерфейса	3	5
6	Тестирование программного модуля	3	5
	Итого за выполнение курсового проекта	36	60
	Защита курсового проекта	15	40

Итоговая оценка курсового проекта (работы) представляет собой сумму баллов за его выполнение и защиту и выставляется в соответствии со шкалой:

Оценка	Отлично	Хорошо	Удовлетворительно	Неудовлетворительно
--------	---------	--------	-------------------	---------------------

Баллы	87-100	65-86	51-64	0-50
-------	--------	-------	-------	------

3 ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

При изучении дисциплины используется модульно-рейтинговая система оценки знаний студентов. Применение форм и методов проведения занятий при изучении различных тем курса представлено в таблице.

№ п/п	Форма проведения занятия*	Вид аудиторных занятий**			Всего часов
		Лекции	Практические занятия	Лабораторные занятия	
1	Мультимедиа	Лекции №1-8			16
2	С использованием ЭВМ			Лр№1-Лр№5	50
	ИТОГО				66

4 ОЦЕНОЧНЫЕ СРЕДСТВА

Используемые оценочные средства по учебной дисциплине представлены в таблице и хранятся на кафедре.

№ п/п	Вид оценочных средств	Количество комплектов
1	Вопросы к экзамену	1
2	Экзаменационные билеты	1
3	Перечень тем для курсовой работы	1
4	Задания для защиты лабораторных работ	5

5 МЕТОДИКА И КРИТЕРИИ ОЦЕНКИ КОМПЕТЕНЦИЙ СТУДЕНТОВ

5.1 Уровни сформированности компетенций

№ п/п	Уровни сформированности компетенции	Содержательное описание уровня*	Результаты обучения**
ПК-2: Владение методами контроля проекта и готовностью осуществлять контроль версий			
ИПК-2.2: Способен организовать работы по управлению IT-проектом			
1	Пороговый уровень	Знание, понимание методов организации работы по управлению IT-проектом	Знание, понимание требований к программной системе, спецификаций программной системы
2	Продвинутый уровень	Применение, анализ методов организации работы по управлению IT-проектом	Анализ требований к программной системе, спецификаций программной системы
3	Высокий уровень	Синтез, оценка методов организации работы по управлению IT-проектом	Синтез требований к программной системе, спецификаций программной системы
ПК-10: Владение навыками использования различных технологий разработки программного обеспечения			

ИПК-10.1: Применяет современные технологии разработки ПО (структурное, объектно-ориентированное)			
1	Пороговый уровень	Знание технологий разработки ПО	Знание технологий разработки программно-информационных систем
2	Продвинутый уровень	Применение, анализ инструментальных средств и технологий разработки ПО	Применение инструментальных средств планирования и проектирования ПО, технологий разработки программно-информационных систем
3	Высокий уровень	Оценка инструментальных средств и технологий разработки ПО	Оценка инструментальных средств планирования проектирования ПО, технологий разработки программно-информационных систем
ПК-12: Владение стандартами и моделями жизненного цикла			
ИПК-12.1: Использует стандарты и модели жизненного цикла программного обеспечения			
1	Пороговый уровень	Знание стандарты жизненного цикла ПО	Знание стандартов жизненного цикла программно-информационных систем
2	Продвинутый уровень	Применение моделей жизненного цикла при разработке ПО	Применение моделей жизненного цикла ПО при разработке программно-информационных систем
3	Высокий уровень	Оценка моделей жизненного цикла ПО	Оценка моделей жизненного цикла ПО с точки зрения эффективности разработки программно-информационных систем

5.2 Методика оценки знаний, умений и навыков студентов

Результаты обучения	Оценочные средства*
ПК-2: Владение методами контроля проекта и готовностью осуществлять контроль версий	
Знание, понимание требований к программной системе, спецификаций программной системы	Задание для защиты лабораторных работ №№ 1,2,5. Требования к выполнению курсовой работы. Вопросы к экзамену.
Анализ требований к программной системе, спецификаций программной системы	Задание для защиты лабораторных работ №№ 1,2,5. Требования к выполнению курсовой работы. Вопросы к экзамену.
Синтез требований к программной системе, спецификаций программной системы	Задание для защиты лабораторных работ №№ 1,2,5. Требования к выполнению курсовой работы. Вопросы к экзамену.
ПК-10: Владение навыками использования различных технологий разработки программного обеспечения	

Знание технологий разработки программно-информационных систем	Задание для защиты лабораторных работ №№ 4-5. Требования к выполнению курсовой работы. Вопросы к экзамену.
Применение инструментальных средств планирования и проектирования ПО, технологий разработки программно-информационных систем	Задание для защиты лабораторных работ №№ 4-5. Требования к выполнению курсовой работы. Вопросы к экзамену.
Оценка инструментальных средств планирования проектирования ПО, технологий разработки программно-информационных систем	Задание для защиты лабораторных работ №№ 4-5. Требования к выполнению курсовой работы. Вопросы к экзамену.
ПК-12: Использует стандарты и модели жизненного цикла программного обеспечения	
Знание стандартов жизненного цикла программно-информационных систем	Задание для защиты лабораторных работ №№ 1-5. Требования к выполнению курсовой работы. Вопросы к экзамену.
Применение моделей жизненного цикла ПО при разработке программно-информационных систем	Задание для защиты лабораторных работ №№ 1-5. Требования к выполнению курсовой работы. Вопросы к экзамену.
Оценка моделей жизненного цикла ПО с точки зрения эффективности разработки программно-информационных систем	Задание для защиты лабораторных работ №№ 1-5. Требования к выполнению курсовой работы. Вопросы к экзамену.

5.3 Критерии оценки лабораторных работ

Каждая выполненная и защищенная лабораторная работа оцениваются в диапазоне от 12 (7) до 15 (10) баллов. При этом 10 (8) баллов начисляется за выполнение работы и от 2 (1) до 5(2) баллов за оформление отчета и защиту работы в зависимости от качества оформления и уровня знаний студента по тематике работы. Если по окончании модуля лабораторная работа выполнена, но не защищена, то баллы по ней не начисляются и она попадает в разряд задолженности.

5.4 Критерии оценки курсовой работы

Курсовая работа включает шесть разделов, которые входят по три в каждый модуль. Каждый раздел оценивается количеством баллов от 6 до 10.

При этом:

- максимальное количество баллов по разделу начисляется в том случае, если студент выполнил раздел в полном объеме и в соответствии методическими указаниями (МУ), проявил элементы творчества, использовал достаточное количество литературных и нормативных источников, аккуратно и правильно оформил графическую часть и пояснительную записку, вовремя представил материалы раздела руководителю;

- минимальное положительное количество баллов по разделу начисляется в том случае, если студент выполнил раздел в соответствии с МУ, не проявил творчества, использовал явно недостаточное количество источников, допустил ошибки в расчетах или графических материалах, но устранил их, представил материалы раздела с отставанием от графика;

- промежуточные значения положительных баллов начисляются в зависимости от уровня творчества студента, наполнения раздела, качества оформления расчетной и графической частей раздела, сроков представления материалов.

При защите работы количество положительных баллов лежит в диапазоне от 15 до 40. При оценке работы учитывается:

1. Полнота решения всех задач проекта и качество содержания проекта;

2. Самостоятельность решения поставленных задач;
 3. Наличие элементов научных исследований (теоретических и экспериментальных);
 4. Наличие элементов творчества студента;
 5. Оформление графической части;
 6. Оформление пояснительной записки;
 7. Четкость и грамотность сообщения;
 8. Качество и глубина ответов на вопросы.
- Каждый из приведенных пунктов оценивается максимальным количеством баллов
- 5.

5.5 Критерии оценки экзамена

Экзаменационный билет включает 1 теоретический вопрос и 2 практических вопроса. Практический вопрос связан с разработкой программ на языке программирования. Теоретический и практический вопросы выбираются из разных дидактических единиц. Каждый вопрос оценивается положительной оценкой в диапазоне от 6 до 8 баллов. Ответы на вопросы оцениваются по следующим критериям.

Теоретические вопросы:

- 16 баллов – студент глубоко понимает пройденный материал, отвечает четко и всесторонне, умеет оценивать факты, использует научную терминологию, самостоятельно рассуждает, отличается способностью обосновать выводы и разъяснять их в логической последовательности, дает развернутый ответ на поставленный вопрос и четко отвечает на дополнительные вопросы.
- 14 баллов – студент глубоко понимает пройденный материал, отвечает четко и всесторонне, умеет оценивать факты, самостоятельно рассуждает, отличается способностью обосновать выводы и разъяснять их в логической последовательности, но допускает отдельные неточности, в том числе и на дополнительные вопросы.
- 12 баллов – студент хорошо понимает пройденный материал, отвечает правильно, умеет оценивать факты, самостоятельно рассуждает, обосновывает выводы и разъясняет их, но допускает ошибки общего характера.
- 10 баллов – студент понимает пройденный материал, но не может теоретически обосновать некоторые выводы, допускает ошибки общего характера.
- 8 баллов – студент отвечает в основном правильно на поставленный вопрос, но чувствуется механическое заучивание материала, отсутствует логическая последовательность при изложении ответа, не может ответить на дополнительные вопросы.
- 6 баллов – в ответе студента имеются существенные недостатки, материал охвачен «половинчато», в рассуждениях допускаются ошибки
- Ниже 6 баллов – студент имеет общее представление о вопросе, ответ студента правилен лишь частично, при разъяснении материала допускаются серьезные ошибки, отсутствует техническая терминология, не может исправить ошибки с помощью наводящих вопросов;

Практический вопрос:

- 12 баллов – студент правильно и грамотно понимает сущность поставленной задачи, четко поясняет методику ее решения, правильно выбирает технические и программные средства, получает результаты выполнения разработанной программы, умеет правильно составить тестовые задания и их применить, четко отвечает на дополнительные вопросы.
- 10 баллов – студент правильно и грамотно понимает сущность поставленной, поясняет методику решения поставленной задачи, правильно выбирает

технические и программные средства, получает результаты выполнения разработанной программы, но не дает обоснование результатов.

- 8 баллов – студент правильно понимает сущность поставленной задачи, поясняет методику ее решения, но с некоторыми ошибками, правильно выбирает технические и программные средства, получает результаты выполнения разработанной программы, но не дает обоснование результатов.

- 6 баллов – студент в целом правильно понимает сущность поставленной задачи, поясняет методику решения поставленной задачи, но с некоторыми ошибками, в целом правильно выбирает технические и программные средства, не рационально составляет программу для решения поставленной задачи, получает результаты выполнения разработанной программы, но не дает обоснование результатов.

- 3 балла – студент не до конца понимает сущность поставленной задачи, поясняет методику решения поставленной задачи, но с существенными ошибками, не рационально выбирает программные средства, с некоторыми ошибками составляет программу решения задачи, получает результаты выполнения программы, но не дает обоснование результатов.

- Ниже 3 баллов – студент неправильно понимает сущность поставленной задачи, не может пояснить методику решения поставленной задачи, плохо разбирается в технических и программных средствах, не может получить и оценить результаты выполнения программы.

6 МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ОРГАНИЗАЦИИ И ВЫПОЛНЕНИЮ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ

Самостоятельная работа студентов (СРС) направлена на закрепление и углубление освоения учебного материала, развитие практических умений. СРС включает следующие виды самостоятельной работы студентов:

Для СРС рекомендуется использовать источники, приведенные в п. 7.

7 УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

7.1 Основная литература

№ п/п	Библиографическое описание	Гриф	Количество экземпляров / URL
1	Гагарина Л.Г. Технология разработки программного обеспечения : учебное пособие / Л.Г. Гагарина, Е.В. Кокорева, Б.Д. Сидорова-Виснадул ; под ред. Л.Г. Гагариной. - М.: ИД "ФОРУМ" : ИНФРА-М, 2022. – 400 с.	Допущено Учебно-методическим объединением вузов по университетскому политехническому образованию в качестве учебного пособия для студентов вузов, обучающихся по направлениям подготовки 09.04.01 и 09.03.03 «Информатика и вычислительная техника»	https://znanium.com/catalog/product/1699927

7.2 Дополнительная литература

№	Библиографическое описание	Гриф	Количество
---	----------------------------	------	------------

п/п			экземпляров
1	Макконнелл С. Совершенный код. Мастер-класс=Code Complete. Second Edition : пер. с англ. / С. Макконнелл. – СПб. : БХВ ; , 2020. – 896с. : ил.	–	5
2	Паттерны проектирования=Head First Design Patterns / Э. Фримен [и др.] ; пер. с англ. Е. Матвеева. – СПб. : Питер, 2016. – 656с. : ил. – (Head First O`Reilly).	–	5
3	Dennis, A. System Analysis & Design. An Object-Oriented Approach with UML=Системный анализ и проектирование на универсальном языке моделирования / А. Dennis, В. Wixom, D. Tegarden. – 5th ed. – New York : John Wiley & Sons, 2015.	–	1
4	Макаровских, Т. А. Документирование программного обеспечения. В помощь техническому писателю : учеб. пособие / Т. А. Макаровских. – 2-е изд. – М. : ЛЕНАНД, 2015. – 266 с.	Рек. НМС по информатике МО и науки РФ в качестве учеб. пособия для студ.	1
5	Гэртнер М. ATDD - разработка программного обеспечения через приемочные тесты / М. Гэртнер; пер. с англ. Слинкина А. А. - М.: ДМК Пресс, 2013. – 232 с.: ил.	–	2
6	Рихтер, Д. CLR via C#. Программирование на платформе Microsoft .NET Framework 4.5 на языке C# / Д. Рихтер ; пер. с англ. Е. Матвеев. – 4-е изд. – СПб. : Питер, 2016. – 896с. : ил. – (Мастер-класс).	–	1
7	Арлоу Д. UML 2 и Унифицированный процесс. Практический объектно-ориентированный анализ и проектирование / Д. Арлоу; М. : Символ-Плюс, 2015. – 624 с.	–	3
8	Ройс, У. Управление проектами по созданию программного обеспечения : унифицированный подход / У. Ройс ; науч. ред. А. Вендров. – М. : Лори, 2014. – 424с.	–	1
9	Орлов С. А. Программная инженерия. Учебник для вузов. Стандарт третьего поколения / С. А. Орлов – СПб.: Питер, 2016. – 640 с.	Допущено Министерством образования и науки Российской Федерации в качестве учебника для студентов высших учебных заведений, обучающихся по специальности «Программное обеспечение вычислительной техники и автоматизированных систем» направлений подготовки дипломированных специалистов «Информатика и вычислительная техника».	1

7.3 Перечень ресурсов сети Интернет по изучаемой дисциплине

1. *Intuit.ru*

2. *Edx.com*
3. *Diagrams.net*

7.4 Перечень наглядных и других пособий, методических рекомендаций по проведению учебных занятий, а также методических материалов к используемым в образовательном процессе техническим средствам

7.4.1 Методические рекомендации

1) Захарченков К.В., Мрочек Т.В. Технологии командной разработки приложений. Методические рекомендации к лабораторным работам для студентов специальности 09 03 04 «Программная инженерия» дневной формы обучения (электронный вариант).

2) Захарченков К.В., Мрочек Т.В. Технологии командной разработки приложений. Методические рекомендации к курсовому проектированию для студентов специальности 09.03.04 «Программная инженерия» дневной формы обучения (электронный вариант).

7.4.3 Информационные технологии

Тема 1 - Жизненный цикл программных систем. Понятие жизненного цикла программных систем.

Тема 2 - Сложность программных систем

Тема 3 - Качество программных систем. Оценка качества программных систем.

Тема 4 - Разработка и анализ требований к программной системе. Спецификации программной системы

Тема 5 - Проектирование архитектуры и структуры программной системы

Тема 6 - Организация разработки программных систем

Тема 7 - Планирование проектирования программной системы

Тема 8 - Испытания программных систем. Внедрение, эксплуатация и сопровождение

7.4.3 Перечень программного обеспечения, используемого в образовательном процессе

1. *MS Visual Studio 2019 Community Edition (свободно распространяемое)*
2. *Eclipse (свободно распространяемое)*

8 МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

Материально-техническое обеспечение дисциплины содержится в паспорте лаборатории а. 517/2, рег. № паспорта лаборатории № ПУЛ - 4 517/2-21; в паспорте лаборатории а. 518/2, рег. № паспорта лаборатории № ПУЛ - 4 518/2-21.