МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ,
МИНИСТЕРСТВО НАУКИ, ВЫСШЕГО ОБРАЗОВАНИЯ И ИННОВАЦИЙ
КЫРГЫЗСКОЙ РЕСПУБЛИКИ

МОО ВО Кыргызско-Российский Славянский университет
имени первого Президента Российской Федерации Б.Н. Ельцина



УТВЕРЖДАЮ
директор Политехнического института
_____ Д.В. Зуев
3 сентября 2026 г.

Процессы управления качеством программного обеспечения

рабочая программа дисциплины (модуля)

Закреплена за **Высшая школа Цифровых технологий**

Учебный план g090404_26_1 пи_оаирп.plx
Направление подготовки 09.04.04 - РФ, 710400 - КР Программная инженерия
Магистерская программа "Основы анализа и разработки приложений с большими объемами распределенных данных"

Квалификация **магистр**

Форма обучения **очная**

Общая трудоемкость **3 ЗЕТ**

Часов по учебному 108

в том числе:

аудиторные занятия 38

самостоятельная 69,9

Виды контроля в семестрах:
экзамен 2

Распределение часов дисциплины по семестрам

Семестр (<Курс>.<Семестр на курсе>)	**2 (1.2)**		Итого	
Недель	18			
Вид занятий	УП	РП	УП	РП
Лекции	16	16	16	16
Практические	22	22	22	22
Контактная работа в период теоретического	0,1	0,1	0,1	0,1
В том числе инт.	8	8	8	8
В том числе в форме практ.подготовки	22	22	22	22
Итого ауд.	38	38	38	38
Контактная работа	38,1	38,1	38,1	38,1
Сам. работа	69,9	69,9	69,9	69,9
Итого	108	108	108	108

Программу составил(и):
ст. преп., Гайдамако В.В.;к.т.н. , доцент, Мусина И.Р. _____

Рецензент(ы):
д.т.н., профессор, Верзунов С.Н. _____

Согласовано РОП:
_____*Лыченко Н.М.*

Рабочая программа дисциплины
Процессы управления качеством программного обеспечения

разработана в соответствии с ФГОС 3++:
Федеральный государственный образовательный стандарт высшего образования - магистратура по направлению подготовки 09.04.04 Программная инженерия (приказ Минобрнауки России от 19.09.2017 г. № 932)

составлена на основании учебного плана:

Направление подготовки 09.04.04 - РФ, 710400 - КР Программная инженерия
Магистерская программа "Основы анализа и разработки приложений с большими объемами распределенных данных"
утвержденного учёным советом вуза от 26.06.2026 протокол № 11.

Рабочая программа одобрена на заседании Высшей школы цифровых технологий

Протокол от 02.09.2026 г. № 1
Срок действия программы: 2024-2028 уч.г.
Руководитель подразделения: директор ВШ ЦТ к.т.н., доцент Хмелева И.В.

ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ
к рабочей программе дисциплины (модуля)

Процессы управления качеством программного обеспечения

_____ *наименование дисциплины*

№ п/п	Основание внесения изменения	Раздел/ пункт	Содержание изменения
1.			
2.			
3.			
4.			
5.			

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ	
1.1	- формирование понимания процессов разработки качественного программного обеспечения;
1.2	- приобретение навыков в организации и управлении процессом верификации ПО;
1.3	- формирование теоретических знаний и практических навыков в области обеспечения качества ПО;
1.4	- систематизация знаний о способах достижения «хорошего» уровня тестированности программ;

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ООП	
Цикл (раздел) ООП:	Б1.О
2.1	**Требования к предварительной подготовке обучающегося:**
2.1.1	Знания об управлении версиями и изменениями программных компонентов, автоматизации сборки, тестирования и развертывания ПО, организации CI/CD-процессов, управлении конфигурациями программных сред и обеспечении воспроизводимости программных поставок.
2.2	**Дисциплины и практики, для которых освоение данной дисциплины (модуля) необходимо как предшествующее:**
2.2.1	Выполнение и защита выпускной квалификационной работы
2.2.2	Преддипломная практика
2.2.3	Верификация распределенных алгоритмов и систем

3. КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)	
ОПК-8: Способен осуществлять эффективное управление разработкой программных средств и проектов.	
Знать:	
Уровень 1	- базовые научные положения и принципы программной инженерии сопровождения и эволюции ПС; - научные основы различных методов адаптации, модификации и тестирования ПО; -специальные программные продукты, реализующие эти методы и технологии.
Уметь:	
Уровень 1	- собирать такую информацию о работе функционирующего ПО, которая позволяет определить конкретные задачи для его модификации и/или адаптации с учётом изменившихся условий эксплуатации; - применять адекватные методы и средства для реализации прикладных задач эволюции ПО в конкретной предметной области
Владеть:	
Уровень 1	- навыками самостоятельного анализа работы сопровождаемого ПО и формулировки конкретных задач по его модификации и адаптации (дальнейшей эволюции ПО); - методами и средствами для выполнения необходимых процессов и работ по разработке, тестированию и сопровождению программных систем; - методами и средствами оценки качества ПО на всех этапах его разработки и сопровождения.

В результате освоения дисциплины обучающийся должен

3.1	**Знать:**
3.1.1	- модели качества программ;
3.1.2	-основные источники рисков в программном проекте;
3.1.3	- виды верификации ПО и их особенности.
3.2	**Уметь:**
3.2.1	- применять модели качества программ;
3.2.2	- применять методы и модели жизненного цикла ПО на практике;
3.2.3	- организовывать процесс верификации программ;
3.2.4	- оценивать полноту тестирования программ различных типов.
3.3	**Владеть:**
3.3.1	- навыком организации и управления процессами обеспечения качества программного обеспечения;
3.3.2	- различными видами автоматического тестирования;

3.3.3	- навыками оценки качества программ;
3.3.4	- формализованными методами разработки тестов и оценки их полноты.

4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

Код занятия	Наименование разделов и тем /вид занятия/	Семестр / Курс	Часов	Компетен-ции	Литература	Инте ракт.	Пр. подг.	Примечание
	Раздел 1. Введение в процессы разработки качественного ПО. Оценка качества программ.							
1.1	Практическая работа №1.Оценка атрибутов качества: функциональности, эффективности, надежности, удобства сопровождения, удобства использования, переносимости (мобильности). Практическая работа №2. Оценка производительности и профилирование программы. Практическая работа №3. Расчет метрик кода в интегрированной среде программирования и их интерпретация. /Пр/	2	6	ОПК-8	Л1.2Л2.1 Л2.2 Л2.3 Э2			
1.2	Проработка лекционного материала. Выполнение пр. работы №1 /Ср/	2	8	ОПК-8	Л1.2Л2.1 Л2.2 Л2.3 Э2			
1.3	Проработка лекционного материала. Выполнение пр. работы №2. /Ср/	2	8	ОПК-8	Л1.2Л2.1 Л2.2 Л2.3 Э2			
1.4	Проработка лекционного материала. Выполнение пр. работы №3. /Ср/	2	12	ОПК-8	Л1.2Л2.1 Л2.2 Л2.3 Э2			
1.5	Определение качества программного обеспечения (ПО). Стандарты качества ПО. Наборы метрик для оценки атрибутов качества. Задачи обеспечения качества. Методы контроля (повышения) качества ПО. Требования к к ПО. Методы верификации. Методологии создания ПО. Причины недостаточного качества. Стандарты IEEE 830 и IEEE 1233. /Лек/	2	4	ОПК-8	Л1.2Л2.1 Л2.2 Л2.3 Э2	2		Обсуждение опыта студентов по проверке качества ПО
	Раздел 2. Современные методологии создания ПО. Верификация программ.							

2.1	«Тяжеловесных» методологии разработки ПО. Краткая характеристика достоинств, недостатков и ограничений каждой из моделей. Принципы Agile разработки ПО. Классификация видов тестирования. Функциональное тестирование. Структурное тестирование: тестирование потока управления, потока данных, тестирование ветвей, тестирование циклов. Инспектирование кода Основные принципы организации процесса тестирования. Взаимодействие в команде. Управление рисками. Структура плана испытаний ПО, основные компоненты плана. Тестовые сценарии. Методика отбора тестовых сценариев. Контроль качества наборов тестов. /Лек/	2	6	ОПК-8	Л1.1Л2.1 Л2.2 Л2.3 Э2	2		Обсуждение опыта студентов тестирования ПО
2.2	Практическая работа №4. Разработка плана тестирования с использованием шаблонов. Практическая работа №5. Разработка тестовых сценариев Практическая работа №6. Структурное тестирование. Создание Unit- тестов. Практическая работа №7. Тестирование базы Данных /Пр/	2	4	ОПК-8	Л1.1Л2.1 Л2.2 Л2.3 Э2			
2.3	Проработка лекционного материала. Выполнение пр. работ №4 и №5. /Ср/	2	8	ОПК-8	Л1.1Л2.1 Л2.2 Л2.3 Э2			
2.4	Выполнение пр. работ №6 и №7 /Ср/	2	8					
	Раздел 3. Техники тестирования. Документирование результатов							

3.1	Мутационное тестирование. Дымовое тестирование (smoke testing). Тестирование новой функции (new feature testing). Подтверждающее тестирование. Регрессионное тестирование. Интуитивное тестирование (ad hoc testing). Критерии завершения тестирования. Нефункциональное тестирование. Рекомендации по описанию дефектов. Степени критичности дефектов. Багтрекинговые системы. Отладка. /Лек/	2	6	ОПК-8	Л1.1Л2.1 Л2.2 Л2.3 Э2			
3.2	Практическая работа №8. Нагрузочное и стрессовое тестирование. Практическая работа №9. Оценивание покрытия кода тестами. /Пр/	2	6	ОПК-8	Л1.1Л2.1 Л2.2 Л2.3 Э2	2		Работа в парах "Разработчик-Тестировщик"
3.3	Практическая работа №10 Описание дефектов. Отладка /Пр/	2	6			2		
3.4	Проработка лекционного материала. выполнение практических работ №8 и №9 /Ср/	2	14	ОПК-8	Л1.1Л2.1 Л2.2 Л2.3 Э2			
3.5	Проработка лекционного материала. Выполнение пр. работы №10. /Ср/	2	11,9	ОПК-8	Л1.1Л2.1 Л2.2 Л2.3 Э2			
3.6	Подготовка к экзамену. /КрТО/	2						
3.7	Сдача экзамена /КрТО/	2	0,1					

5. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

5.1. Контрольные вопросы и задания

Контрольные вопросы для проверки ЗНАТЬ:
1. Определение качества программного обеспечения.
2. Процессы, необходимые для создания качественного ПО
3. Методологии создания ПО
4. Принципы Agile разработки ПО.
5. Модели жизненного цикла ПО.
6. Стандарты качества ПО.
7. Метрики для оценки атрибутов качества.
8. Задачи обеспечения качества.
9. Методы контроля (повышения) качества ПО.
8. Требования к ПО. Задачи извлечения и анализа требований. Необходимые свойства требований.
9. Требования к надежности.
10. Требования к эффективности.
11. Требования к удобству сопровождения
12. Требования к удобства использования,
13 Требования к переносимости (мобильности).
14. Причины недостаточного качества.
15. Методы верификации ПО: экспертиза, статический анализ, динамический анализ, формальные методы, синтетические методы верификации..
16. Основные принципы организации процесса тестирования.
17. Планирование верификации и аттестации: V – модель планирования разработки ПО.
18. Схема двухэтапного процесса тестирования.

19. Структура плана испытаний ПО, основные компоненты плана.
20. Тестовые сценарии. Методика отбора тестовых сценариев.
21. Контроль качества наборов тестов.
22. Взаимодействие в команде при тестировании.
23.Управление рисками.
24. Инструменты и инфраструктура при тестировании .
25. Как проводить отладку программы?

Контрольные задачи для проверки УМЕТЬ:
1. Измерять атрибуты качества ПО и интерпретировать результаты измерения.
2. Использовать автоматизированные средства для расчёта метрик ПО.
3. Планировать и организовывать процесс верификации и валидации ПО .
4. Для проекта разработки ПО определять наиболее подходящую модель жизненного цикла

Контрольные задания для проверки ВЛАДЕТЬ:
1. Средствами автоматизированного тестирования ПО
2. Средствами автоматизированного расчета показателей качества ПО.

5.2. Темы курсовых работ (проектов)

не предусмотрены

5.3. Фонд оценочных средств

Перечь заданий к практическим работам.
Практическая работа №1.Оценка атрибутов качества: функциональности, эффективности, надежности, удобства сопровождения, удобства использования, переносимости (мобильности).
1.Имеется web-приложения, расположенного по адресу http://www.klerk.ru
2. Для данного приложения дать оценку следующим характеристикам:
2.1)удобство использования;
2.2) планировка и дизайн;
2.3) соответствие содержания назначению;
2.4) графика;
2.5) цвета;
2.6) оформление текста;
2.7) устойчивость к ошибкам;
2.8) независимость от платформы и особенности ОС.
3. Результаты представить в виде распечатанного отчета.

Практическая работа №2. Расчет производительности и профилирование программы.
1. Выбрать программу с открытым кодом.
2. Определить реальную производительность и используемую память. Сравнить с требуемыми метриками (из ТЗ).
3. Результаты представить в виде распечатанного отчета со скриншотами

Практическая работа №3. Расчет метрик кода в интегрированной среде программирования и их интерпретация.
1Выбрать программу с открытым кодом
2. В интегрированной среде программирования (можно использовать MS VS любой версии, Net Beans или Eclipse) рассчитать метрики кода:
2.1.Цикломатическую сложность
2.2.глубину наследования
2.3.связанность модулей (классов)
2.4..количество строк кода
2.5.индекс удобства поддержки
3. Провести интерпретацию результатов
4. Представить отчет со скриншотами.

Практическая работа №4. Разработка плана тестирования с использованием шаблонов.
1.Выбрать программу с открытым кодом;
2. Задан шаблон разработки плана верификации программы;
3. Разработать план тестирования программы.
4. Оформить отчет.

Практическая работа №5. Разработка тестовых сценариев
1 Выбрать программу с заданным техническим заданием;
2. Создать тестовые сценарии, используя для формирования данных методы разбиения по областям эквивалентности и граничных значений

5. Составить отчет с выводом об эффективности выбранных сценариев.

Практическая работа №6. Структурное тестирование. Создание Unit- тестов.
1. В качестве тестируемой программы взять ту же программу, что и в практической работе №5
2.Провести тестирование в выбранной интегрированной среде программирования.
3. Показать процент покрытия кода тестами.
4. Составить отчет с выводом об эффективности проведенного модульного тестирования.

Практическая работа №7. Тестирование базы данных
1. Выбрать базу данных (желательно на SQL SERVER)
2. Протестируйте функции, триггеры, хранимые процедуры
3. Составьте отчет.

Практическая работа №8. Стрессовое и нагрузочное тестирование
1. В качестве тестируемой программы взять ту же программу, что и в практической работе №5
2. Провести нагрузочное тестирование в выбранной среде.
3. Составить отчет по результатам проведенного тестирования с нарастающей нагрузкой.

Практическая работа №9. Оценивание покрытия кода тестами
1. Выбрать приложение для тестирования.
2. Провести Unit-тестирование.
3. С помощью готового инструментария в IDE оценить процент покрытия кода.
4. Составить отчет.

Практическая работа №10.Описание дефектов
1. Выбрать приложение для тестирования.
2. Провести тестирование
3. Описать дефекты в виде таблицы.
Задания к самостоятельным работам и контрольные вопросы

Самостоятельная работа №1. Оценка атрибутов качества: функциональности, эффективности, надежности, удобства сопровождения, удобства использования, переносимости (мобильности).
1.Выберите web-приложение.
2.Для выбранного приложения дайте оценку следующим метрикам качества программы:
2.1)удобство использования;
2.2) планировка и дизайн;
2.3) соответствие содержания назначению;
2.4) графика;
2.5) цвета;
2.6) оформление текста;
2.7) устойчивость к ошибкам;
2.8) независимость от платформы и особенности ОС.
3. Оформите отчет.
Ответьте на вопросы:
1. Изложите модель качества ПО по ISO 9126.
2.Перечислите метрики качества ПО.

Самостоятельная работа №2. Расчет производительности и профилирование программы.
1. Выбрать программу с открытым кодом.
2. Определить производительность и используемую память. Сравнить с требуемыми метриками (из ТЗ).
2. Результаты представить в виде распечатанного отчета со скриншотами.
3. Ответьте на вопросы:
3.1. Что дают результаты тестирования производительности разработчику?
3.2. Какие пути программы являются «горячими» (hot paths) ?
3.2 Как можно улучшить показатели производительности и памяти для выбранной программы?

Самостоятельная работа №3. Расчет метрик кода в интегрированной среде программирования и их интерпретация
1.Рассчитайте метрики кода для программы нахождения корня нелинейного уравнения с использованием, как минимум двух методов поиска.
2. Ответьте на вопросы:
2.1 Что показывают метрики кода в MS VS?
2.2 Как знание метрик кода может улучшить программу?

Самостоятельная работа №4. Разработка плана тестирования с использованием шаблонов
1.Разработайте тестовый план для программы из сам. работы №3
2.Проведите тестирование согласно плану.

3. Ответьте на вопросы:
3.1.На каком этапе разработки ПО строится план тестирования?
3.2 Что такое TTD?
3.3 Менеджер проекта решил для оценки специалиста в качестве исходных данных воспользоваться отчетами о результатах инспектирования программы. В отчетах содержится информация о том, кто совершил и кто обнаружил ошибки в программе. Этичны ли действия? Этично ли заранее проинформировать персонал об этом? Как это решение может повлиять на процесс инспектирования?

Самостоятельная работа №5. Разработка тестовых сценариев.
1.Разработайте набор тестов для функции вычисления наибольшего общего делителя двух положительных целых чисел: int NOD(int a, int b).
2.Ответьте на вопрос.
“Два тестовых примера принадлежат одному классу эквивалентности если:”
Варианты ответов (возможен выбор нескольких вариантов)
1)получен один и тот же результат;
2)получена одинаковая реакция системы
3)они построены по одному тест-требованию
4)от них ожидается получить одинаковую реакцию системы

Самостоятельная работа №6. Структурное тестирование. Создание Unit- тестов.
1.Проведите модульное тестирование для задачи из сам. работы №5.
2. Покажите процент покрытия кода тестами.
3. Проанализируйте результаты тестирования.
4. Ответьте на вопросы:
4.1. Насколько эффективны Unit-тесты?
4.2. Почему исчерпывающее тестирование невозможно?

Самостоятельная работа №7. Тестирование базы данных
1.Разработайте базы данных для системы учета товаров на складе
2.Проведите тестирование этой базы данных.
3.Один из подходов, широко используемых при тестировании ПО, состоит в тестировании системы до тех пор, пока не будут израсходованы все средства, выделенные на тестирование. Затем система передается заказчику. Обсудите этичность такого подхода.

Самостоятельная работа №8. Нагрузочное тестирование
1.Проведите нагрузочное тестирование для задачи из сам. работы №5.
2.Составьте отчет по тестированию.
3.Ответьте на вопросы:
3.1 Как правильно проводить нагрузочное тестирование?
3.2 Что дают результаты нагрузочного тестирования для разработчика?
3.3 До каких пор следует тестировать программу, чтобы удостовериться, что она соответствует требованиям производительности?

Самостоятельная работа №9. Оценивание покрытие кода
1. Проведите модульное тестирование выбранного приложения.
2. Оцените в процентном соотношении покрытие кода тестами.
3. Ответьте на вопрос:
Может ли процент покрытия кода служить основным критерием полноты проведенного тестирования.
4. Составьте отчет по проведенному виду тестирования.

Самостоятельная работа №10.Документирование дефектов
1. Проведите тестирование приложения.
2. Составьте таблицу с описанием дефектов.
3. Проведите анализ дефектов, разбив их по категориям.

5.4. Перечень видов оценочных средств

Практическая работа; Самостоятельная работа. Шкалы оценивания представлены в Приложении 1.

6. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

6.1. Рекомендуемая литература

6.1.1. Основная литература

	Авторы, составители	Заглавие	Издательство, год
Л1.1	Куликов С.С.	Тестирование программного обеспечения. Базовый курс: 3-издание. Учебное пособие	Минск: Четыре четверти, Электронная версия 3.2.7 от 07.07.2025.

	Авторы, составители	Заглавие	Издательство, год
Л1.2	Проскуряков, А. В.	Качество и тестирование программного обеспечения. Метрология программного обеспечения : учебное пособие	Таганрог :Издательство Южного федерального университета, 2022. ISBN 978-5-9275-4044-0

6.1.2. Дополнительная литература

	Авторы, составители	Заглавие	Издательство, год
Л2.1	Коцюба И. Ю., Чунаев А. В., Шиков А. Н.	Методы оценки и измерения характеристик информационных систем: Учебное пособие	Санкт-Петербург: Университет ИТМО 2016
Л2.2	Mickey Gousset, Brian Keller, Martin Woodward.	Professional Application Lifecycle Management with Visual Studio	John Wiley & Sons, Inc 2012
Л2.3	Афанасьева Т. В., Афанасьев А. Н.	Основы управления качеством программных средств: Учебное пособие	Ульяновск: Ульяновский государственный технический университет 2017

6.2. Перечень ресурсов информационно-телекоммуникационной сети "Интернет"

Э1	ГОСТ Р ИСО/МЭК 25010-2015 Информационные технологии. Системная и программная инженерия. Требования и оценка качества систем и программного обеспечения (SQuaRE). Модели качества систем и программных продуктов	https://allgosts.ru/35/080/gost_r_iso!mek_25010-2015
Э2	ГОСТ Р 56920-2024 Системная и программная инженерия. Тестирование программного обеспечения. Общие положения	https://allgosts.ru/35/080/gost_r_56920-2024

6.3. Перечень информационных и образовательных технологий

6.3.1 Компетентностно-ориентированные образовательные технологии

6.3.1.1	Изучение дисциплины студентами осуществляется в форме лекций, практических занятий в аудиторных условиях (компьютерные классы) и в процессе самостоятельной работы, контроля знаний.
6.3.1.2	Теоретическая информация, по мере возможности, представляется в виде компьютерных презентаций с использованием мультимедийных средств.
6.3.1.3	Практические занятия проводятся в компьютерных классах, оснащенных персональными компьютерами с необходимыми параметрами и с установленным необходимым программным обеспечением. Используется Интернет для получения дополнительной информации. Используется дискуссионный метод проведения занятий, где студенты могут высказать свое мнение по обсуждаемой проблеме.
6.3.1.4	Защита практических работ проводится в виде собеседования с преподавателем по теории и программной реализации работы.

6.3.2 Перечень информационных справочных систем и программного обеспечения

6.3.2.1	Любая интегрированная среда программирования с возможностью создания тестового проекта. Предпочтительна
6.3.2.2	MS VISUAL STUDIO CODE, Selenium, Katalon
6.3.2.3	http://msdn.microsoft.com; https://docs.seleniumhq.org/; https://katalon.com/katalon-studio

7. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

7.1	Учебная лаборатория программно-технического обеспечения:
7.2	ПК - 10 шт;
7.3	сервер - 1;
7.4	ПК преподавателя - 1.
7.5	Локальная сеть Высшей школы Цифровых технологий.
7.6	Интернет со скоростью 70 Мбит/сек.
7.7	Зона WI-FI.
7.8	Интерактивная доска, проектор, обычная доска, 50 посадочных мест

8. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ)

1.Технологическая карта дисциплины представлена в Приложении 2.
2.Текущий контроль осуществляется в течение семестра в виде защиты практических работ.
Методические указания по выполнению практических работ представлены в электронной папке преподавателя (локальная сеть Высшей школы Цифровых технологий). Выполнение всех практических работ завершается оформлением отчета, в котором приводится теория по теме работы, отчеты в виде скриншотов результатов, таблиц и графиков.
3.Рубежный контроль осуществляется в виде сдачи самостоятельных работ.

Вопросы по контрольным работам представлены в разделе РПД Фонд оценочных средств. Результаты самостоятельных работ предоставляются виде распечатанных отчетов.

4.Для допуска к экзамену студент в течении семестра должен сдать все практические и самостоятельные работы.