

МИНИСТЕРСТВО ТРАНСПОРТА РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«РОССИЙСКИЙ УНИВЕРСИТЕТ ТРАНСПОРТА»
(РУТ (МИИТ))



Рабочая программа дисциплины (модуля),
как компонент образовательной программы
специализированного высшего образования
по направлению подготовки
09.04.01 Информатика и вычислительная техника,
утвержденной первым проректором РУТ (МИИТ)
Тимониным В.С.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Управление качеством ПО

Направление подготовки: 09.04.01 Информатика и вычислительная техника

Направленность (профиль): Технологии проектирования программного обеспечения

Форма обучения: Очная

Рабочая программа дисциплины (модуля) в виде
электронного документа выгружена из единой
корпоративной информационной системы управления
университетом и соответствует оригиналу

Простая электронная подпись, выданная РУТ (МИИТ)
ID подписи: 5665
Подписал: заведующий кафедрой Нутович Вероника
Евгеньевна
Дата: 01.09.2026

1. Общие сведения о дисциплине (модуле).

Цель дисциплины «Управление качеством ПО» состоит в том, чтобы научить студентов практическим навыкам организации и управления процессом обеспечения качества программного обеспечения на всех этапах.

Задачи данной дисциплины:

- изучение основных задач процесса обеспечения качества программного обеспечения;
- изучение основных принципов организации процесса обеспечения качества программного обеспечения;
- приобретений практических навыков работы с инструментами для разных видов тестирования;
- изучение методов организации процесса обеспечения качества программного обеспечения.

2. Планируемые результаты обучения по дисциплине (модулю).

Перечень формируемых результатов освоения образовательной программы (компетенций) в результате обучения по дисциплине (модулю):

ПК-4 - Способен осуществлять руководство процессом обеспечения качества разрабатываемого программного продукта для корпоративного рынка.

Обучение по дисциплине (модулю) предполагает, что по его результатам обучающийся будет:

Знать:

- задачи процесса обеспечения качества программного обеспечения;
- принципы организации процесса обеспечения качества программного обеспечения;
- виды тестирования и их особенности;
- методы обеспечения процесса управления качеством программного обеспечения;
- отличие и взаимосвязь Quality Assurance (QA), Quality Control (QC) и Testing.

Уметь:

- использовать инструменты тестирования для обеспечения качества программного обеспечения;

- следовать принципам обеспечения качества программного обеспечения для решения задач процесса управления качеством программного обеспечения;
- следовать методам организации процесса обеспечения качества программного обеспечения;
- тестировать производительность, безопасность, юзабилити, совместимость и др;
- делать выбор и внедрение инструментов.

Владеть:

- навыком организации и управления процессами обеспечения качества программного обеспечения с использованием методов обеспечения качества программного обеспечения;
- навыками использования инструментов тестирования для обеспечения качества программного обеспечения;
- навыками использования техник оценки рисков;
- навыками использования метрик для принятия решений и отчетности;
- навыками применения базовых методов анализа корневых причин для выявления первоисточников повторяющихся или критичных дефектов.

3. Объем дисциплины (модуля).

3.1. Общая трудоемкость дисциплины (модуля).

Общая трудоемкость дисциплины (модуля) составляет 5 з.е. (180 академических часа(ов)).

3.2. Объем дисциплины (модуля) в форме контактной работы обучающихся с педагогическими работниками и (или) лицами, привлекаемыми к реализации образовательной программы на иных условиях, при проведении учебных занятий:

Тип учебных занятий	Количество часов	
	Всего	Семестр №2
Контактная работа при проведении учебных занятий (всего):	48	48
В том числе:		
Занятия лекционного типа	16	16
Занятия семинарского типа	32	32

3.3. Объем дисциплины (модуля) в форме самостоятельной работы обучающихся, а также в форме контактной работы обучающихся с педагогическими работниками и (или) лицами, привлекаемыми к реализации

образовательной программы на иных условиях, при проведении промежуточной аттестации составляет 132 академических часа (ов).

3.4. При обучении по индивидуальному учебному плану, в том числе при ускоренном обучении, объем дисциплины (модуля) может быть реализован полностью в форме самостоятельной работы обучающихся, а также в форме контактной работы обучающихся с педагогическими работниками и (или) лицами, привлекаемыми к реализации образовательной программы на иных условиях, при проведении промежуточной аттестации.

4. Содержание дисциплины (модуля).

4.1. Занятия лекционного типа.

№ п/п	Тематика лекционных занятий / краткое содержание
1	<p>Введение в управление качеством программного обеспечения (SQM). Основные процессы управления качеством программного обеспечения.</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - что такое качество ПО, аспекты качества (ISO 25010); - цели и задачи SQM, важность качества в современном мире; - отличие и взаимосвязь Quality Assurance (QA), Quality Control (QC) и Testing; - экономика качества ПО; - роль различных участников команды в обеспечении качества; - цикл Деминга (PDCA) применительно к SQM; - планирование качества (Quality Planning); - обеспечение качества (Quality Assurance); - контроль качества (Quality Control); - улучшение качества (Quality Improvement).
2	<p>Планирование качества. Обеспечение качества (Quality Assurance). Контроль качества (Quality Control).</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - определение заинтересованных сторон и их ожиданий; - сбор и анализ требований к качеству (функциональные и нефункциональные); - определение стандартов и политик качества проекта/организации; - оценка рисков качества; - разработка плана управления качеством (Quality Management Plan): структура и содержание; - цели и задачи QA: предотвращение дефектов; - анализ требований и спецификаций (Reviews); - статические методы QA: инспекции, сквозные просмотры (walkthroughs), аудиты; - статический анализ кода и документации (Static Code Analysis); - обеспечение качества процессов разработки; - метрики процессов QA; - цели и задачи QC: выявление дефектов; - тестирование как основной метод контроля качества; - виды и уровни тестирования в контексте контроля качества;

№ п/п	Тематика лекционных занятий / краткое содержание
	<ul style="list-style-type: none"> - анализ результатов контроля и обратная связь; - управление дефектами как часть QC.
3	<p>Нефункциональное тестирование как элемент контроля качества. Управление рисками качества.</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - тестирование производительности, безопасности, юзабилити, совместимости и др; - как нефункциональные требования трансформируются в активности контроля качества; - метрики нефункционального качества; - инструменты для нефункционального тестирования (обзор в контексте QC); - что такое риски качества, источники рисков в разработке ПО; - процесс управления рисками: идентификация, анализ, оценка, реагирование, мониторинг; - техники оценки рисков (например, матрица "вероятность/воздействие"); - риск-ориентированное тестирование как подход к контролю качества; - управление рисками на разных этапах жизненного цикла.
4	<p>Метрики качества программного обеспечения.</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - что такое метрики и зачем они нужны в SQM; - классификация метрик (продуктовые, процессные, проектные); - ключевые метрики качества продукта (количество дефектов, плотность дефектов, время отклика, надежность, покрытие тестами и др.); - ключевые метрики процесса (эффективность ревью, скорость устранения дефектов и др.); - сбор, анализ и интерпретация метрик; - использование метрик для принятия решений и отчетности.
5	<p>Управление конфигурацией в контексте SQM. Улучшение качества (Quality Improvement).</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - цели и задачи управления конфигурацией: обеспечение целостности и стабильности; - ключевые аспекты: идентификация элементов конфигурации, контроль версий, контроль изменений, сборка, управление релизами; - как управление конфигурацией поддерживает качество ПО; - инструменты управления конфигурацией; - цели и задачи улучшения качества; - анализ корневых причин дефектов (RCA); - разработка и реализация корректирующих и предупреждающих действий (CAPA); - модели и подходы к улучшению процессов; - Цикл улучшения качества.
6	<p>SQM в гибких методологиях (Agile).</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - особенности управления качеством в Scrum, Kanban; - принципы Agile, способствующие качеству (работающая программа важнее документации, взаимодействие, адаптация к изменениям); - роль SQM-активностей в итеративном процессе (планирование качества в спринте, непрерывная интеграция и поставка, автоматизация, демонстрация, ретроспектива); - понятие "Done" (Definition of Done) как критерий качества.
7	<p>SQM и DevOps.</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - интеграция разработки, тестирования и эксплуатации; - культура качества в DevOps: ответственность всех, автоматизация "везде"; - концепция Continuous Integration/Continuous Delivery/Continuous Deployment (CI/CD) и качество;

№ п/п	Тематика лекционных занятий / краткое содержание
	- мониторинг качества в процессе эксплуатации (Production Monitoring); - инструменты для CI/CD и мониторинга.
8	Инструменты поддержки управления качеством ПО. Рассматриваемые вопросы: - категории инструментов: управление требованиями, управление тестами, баг-трекинг, автоматизация тестирования, статический анализ, управление конфигурацией, мониторинг, системы отчетности; - обзор популярных инструментов в каждой категории с точки зрения их вклада в SQM; - выбор и внедрение инструментов

4.2. Занятия семинарского типа.

Практические занятия

№ п/п	Тематика практических занятий/краткое содержание
1	Анализ требований с точки зрения качества. В рамках выполнения практических работ студент получит навыки анализа функциональных и нефункциональных требований, идентификации потенциальных проблем качества на раннем этапе (QA активность), а также формулирования четких и измеримых критериев приемки.
2	Разработка разделов плана управления качеством. В рамках выполнения практических работ студент получит навыки структурирования и наполнения ключевых разделов Плана управления качеством для гипотетического проекта, включая определение стандартов, подходов к QA/QC, управлению рисками и метрикам.
3	Оценка рисков качества и разработка мер реагирования. В рамках выполнения практических работ студент получит навыки идентификации, анализа и оценки потенциальных рисков проекта с точки зрения их влияния на качество ПО, а также предложит стратегии и конкретные меры по их снижению или предотвращению.
4	Проведение статического анализа документации и кода. В рамках выполнения практических работ студент получит навыки участия в обзорах документации (например, требований или проектных решений) или использования простых инструментов статического анализа кода для выявления потенциальных дефектов на этапе разработки (QA активность).
5	Анализ результатов контроля качества. В рамках выполнения практических работ студент получит навыки анализа данных, полученных в результате тестирования (статистика дефектов, покрытие тестами и др.), для оценки текущего состояния качества продукта и процесса.
6	Проведение анализа корневых причин (RCA). В рамках выполнения практических работ студент научится применять базовые методы анализа корневых причин для выявления первоисточников повторяющихся или критичных дефектов, формулировать выводы для улучшения процессов.
7	Применение метрик для оценки качества и процессов. В рамках выполнения практических работ студент получит навыки расчета и интерпретации ключевых метрик качества (например, плотность дефектов, скорость обнаружения/устранения дефектов, покрытие тестами) для оценки эффективности процессов QA/QC и принятия решений.
8	Работа с инструментами управления дефектами и тестами в контексте SQM. В рамках выполнения практических работ студент получит навыки использования возможностей BTS/TMS для отслеживания дефектов, управления тест-кейсами, сбора метрик и создания отчетов о качестве.

№ п/п	Тематика практических занятий/краткое содержание
9	Применение принципов управления конфигурацией для обеспечения стабильности. В рамках выполнения практических работ студент получит представление о том, как использование систем контроля версий и процессов управления изменениями помогает поддерживать стабильность и качество кодовой базы и релизов.
10	Мониторинг качества на этапах эксплуатации и обратная связь. В рамках выполнения практических работ студент научится рассматривать методы сбора обратной связи от пользователей, анализа данных мониторинга продакшена (ошибки, производительность) для выявления проблем качества после релиза и планирования улучшений.
11	Разработка предложений по улучшению процесса. В рамках выполнения практических работ студент, основываясь на анализе метрик и корневых причин, разработает конкретные, измеримые предложения по улучшению одного из процессов разработки/тестирования с целью повышения качества ПО.

4.3. Самостоятельная работа обучающихся.

№ п/п	Вид самостоятельной работы
1	Работа с лекционным материалом.
2	Работа с литературой.
3	Текущая подготовка к занятиям.
4	Выполнение курсовой работы.
5	Подготовка к промежуточной аттестации.
6	Подготовка к текущему контролю.

4.4. Примерный перечень тем курсовых работ

1. Исследование методов и инструментов управления качеством программного обеспечения.

2. Анализ проблем, связанных с качеством программного обеспечения, и разработка решений для их устранения.

3. Исследование процесса тестирования программного обеспечения и разработка методов для его оптимизации.

4. Анализ влияния архитектуры программного обеспечения на его качество и разработка рекомендаций для улучшения архитектуры.

5. Исследование методов управления рисками в разработке программного обеспечения и разработка рекомендаций для минимизации рисков.

6. Анализ процесса управления изменениями в программном обеспечении и разработка методов для его оптимизации.

7. Исследование методов управления качеством программного обеспечения в Agile-разработке.

8. Исследование методов автоматизации тестирования программного обеспечения и разработка рекомендаций для оптимизации процесса.

9. Анализ влияния процесса управления конфигурацией на качество программного обеспечения и разработка рекомендаций для его улучшения.

10. Исследование процесса контроля качества в разработке мобильных приложений и разработка методов для его оптимизации.

11. Анализ влияния процесса управления требованиями на качество программного обеспечения и разработка рекомендаций для его улучшения.

12. Исследование методов управления качеством программного обеспечения в интеграционной разработке.

13. Анализ проблем, связанных с безопасностью программного обеспечения, и разработка решений для их устранения.

14. Исследование методов управления качеством программного обеспечения в разработке больших проектов.

15. Анализ влияния процесса управления релизами на качество программного обеспечения и разработка рекомендаций для его улучшения.

16. Анализ влияния процесса обеспечения качества на удовлетворенность пользователей программным обеспечением и разработка рекомендаций для его улучшения.

17. Исследование методов тестирования безопасности программного обеспечения и разработка рекомендаций для повышения его защищенности.

18. Исследование методов тестирования производительности программного обеспечения и разработка рекомендаций для его оптимизации.

19. Исследование методов тестирования интерфейсов программного обеспечения и разработка рекомендаций для повышения их удобства использования.

20. Анализ влияния процесса управления конфиденциальностью на качество программного обеспечения и разработка рекомендаций для его улучшения.

21. Исследование методов тестирования функциональности программного обеспечения и разработка рекомендаций для повышения его надежности.

22. Анализ влияния процесса управления доступностью на качество программного обеспечения и разработка рекомендаций для его улучшения.

23. Исследование методов тестирования совместимости программного обеспечения и разработка рекомендаций для его оптимизации.

24. Исследование методов тестирования многопоточных приложений и разработка рекомендаций для повышения их производительности.

25. Исследование методов тестирования интеграции программного обеспечения и разработка рекомендаций для повышения его совместимости.

5. Перечень изданий, которые рекомендуется использовать при освоении дисциплины (модуля).

№ п/п	Библиографическое описание	Место доступа
1	Дукельский, К. В. Управление качеством программного обеспечения : учебное пособие / К. В. Дукельский, И. Б. Бондаренко. — Санкт-Петербург : СПбГУТ им. М.А. Бонч-Бруевича, 2021. — 52 с. — Текст : электронный Учебное пособие	https://e.lanbook.com/book/279632 (дата обращения: 17.04.2025)
2	Семахин, А. М. Методы верификации и оценки качества программного обеспечения : учебное пособие / А. М. Семахин. — Курган : КГУ, 2018. — 150 с. — ISBN 978-5-4217-0461-4. — Текст : электронный Учебное пособие	https://e.lanbook.com/book/177908 (дата обращения: 17.04.2025)
3	Сорока, Е. Г. Управление качеством программного продукта : учебное пособие для вузов / Е. Г. Сорока. — Санкт-Петербург : Лань, 2021. — 100 с. — ISBN 978-5-8114-7519-3. — Текст : электронный Учебное пособие	https://e.lanbook.com/book/176878 (дата обращения: 17.04.2025)

6. Перечень современных профессиональных баз данных и информационных справочных систем, которые могут использоваться при освоении дисциплины (модуля).

Официальный сайт РУТ (МИИТ) (<https://www.miit.ru/>).

Научно-техническая библиотека РУТ (МИИТ) (<http://library.miit.ru>).

Образовательная платформа «Юрайт» (<https://urait.ru/>).

Электронно-библиотечная система издательства «Лань» (<http://e.lanbook.com/>).

Электронно-библиотечная система ibooks.ru (<http://ibooks.ru/>).

7. Перечень лицензионного и свободно распространяемого программного обеспечения, в том числе отечественного производства, необходимого для освоения дисциплины (модуля).

Selenium

JMeter

Appium

Bombardier

LoadRunner

Cucumber

Пакет офисных приложений

Браузер с доступом в интернет

8. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю).

Для лекционных занятий – наличие проектора и экрана.

Для практических занятий – наличие персональных компьютеров вычислительного класса.

9. Форма промежуточной аттестации:

Курсовая работа во 2 семестре.

Экзамен во 2 семестре.

10. Оценочные материалы.

Оценочные материалы, применяемые при проведении промежуточной аттестации, разрабатываются в соответствии с локальным нормативным актом РУТ (МИИТ).

Авторы:

старший преподаватель кафедры
«Цифровые технологии управления
транспортными процессами»

И.С. Разживайкин

Согласовано:

Заведующий кафедрой ЦТУТП

В.Е. Нутович

Председатель учебно-методической
комиссии

Н.А. Андриянова