

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



Рабочая программа дисциплины (модуля),
как компонент образовательной программы
высшего образования - программы бакалавриата
по направлению подготовки
10.03.01 Информационная безопасность,
утвержденной первым проректором РУТ (МИИТ)
Тимониным В.С.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Методы разработки программных систем

Направление подготовки: 10.03.01 Информационная безопасность

Направленность (профиль): Безопасность компьютерных систем

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: 4196
Подписал: заведующий кафедрой Желенков Борис
Владимирович
Дата: 18.03.2026

1. Общие сведения о дисциплине (модуле).

Целями освоения дисциплины (модуля) являются:

- изучение методов и средств разработки программных систем;
- изучение этапов разработки программного обеспечения,
- изучение методов и средств, используемых для автоматизации разработки программного обеспечения на каждом этапе жизненного цикла программного обеспечения.

Задачами освоения дисциплины (модуля) являются:

- разработка технических спецификаций на программные компоненты и их взаимодействие;
- проектирование программного обеспечения;
- описание проекта программного комплекса.

2. Планируемые результаты обучения по дисциплине (модулю).

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

ОПК-7 - Способен использовать языки программирования и технологии разработки программных средств для решения задач профессиональной деятельности ;

ПК-2 - способностью применять программные средства системного, прикладного и специального назначения, инструментальные средства, языки и системы программирования для решения профессиональных задач .

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

Уметь:

- выбирать средства реализации требований к программному обеспечению;
- вырабатывать варианты реализации программного обеспечения;
- проводить оценку и обоснование рекомендуемых решений;
- осуществлять коммуникации с заинтересованными сторонами.

Владеть:

- навыками разработки и согласования технических спецификаций на программные компоненты и их взаимодействие с архитектором программного обеспечения;

- навыками распределения заданий между программистами в соответствии с техническими спецификациями; осуществления контроля выполнения заданий;
- навыками осуществления обучения и наставничества;
- навыками формирования и предоставления отчетности в соответствии с установленными регламентами;
- навыками оценки и согласования сроков выполнения поставленных задач.

Знать:

- языки формализации функциональных спецификаций;
- методы и приемы формализации задач; методы и средства проектирования программного обеспечения;
- методы и средства проектирования программных интерфейсов; методы и средства описания данных.

3. Объем дисциплины (модуля).

3.1. Общая трудоемкость дисциплины (модуля).

Общая трудоемкость дисциплины (модуля) составляет 4 з.е. (144 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 №3
Контактная работа при проведении учебных занятий (всего):	80	80
В том числе:		
Занятия лекционного типа	32	32
Занятия семинарского типа	48	48

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 64 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	Унифицированный язык моделирования Рассматриваемые вопросы: - стандарты языка UML (Unified Modeling Language), - диаграммы, виды диаграмм.
2	Диаграмма вариантов использования Рассматриваемые вопросы: - актеры, - варианты использования, - поток событий, - отношения между актерами и вариантами использования.
3	Задача регистрации курсов Рассматриваемые вопросы: - диаграмма вариантов использования, - компоненты на диаграмме вариантов использования, - подсистемы, - примеры описания потоков событий, - подпотоки и альтернативные потоки.
4	Диаграмма классов Рассматриваемые вопросы: - понятие объекта и класса, - графическое изображение класса, - описание атрибутов и операций, - видимость атрибутов и операций, - отношения между классами, - абстрактный класс, - интерфейс, - параметризованный класс, - пакеты и их зависимости, - стереотипы зависимостей.
5	Задача регистрации курсов Рассматриваемые вопросы: - диаграмма классов, - классы-сущности, - граничные классы, - управляющие классы, - ассоциативные классы, - рефлексивные отношения, - примеры описания пакетов.

№ п/п	Тематика лекционных занятий / краткое содержание
6	<p>Диаграмма деятельности</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - действия, поток управления, - элементы ветвления, слияния, - разделения, - соединения, - разделы, - представление состояния объектов, - поток объектов.
7	<p>Диаграмма состояний</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - состояния, - события сигнала, - вызова, изменения, - времени, сигналы и отношения между ними, - переходы между состояниями, - условие перехода, - действия при переходе и в состоянии, - составные состояния.
8	<p>Диаграммы взаимодействия</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - диаграмма последовательности, - объекты, сообщения, - синхронные и асинхронные сообщения, - диаграмма коммуникации, - очередность событий на диаграммах.
9	<p>Диаграммы реализации</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - диаграмма компоненто, - реализация компонентов, - зависимости между компонентами, интерфейсы, - диаграмма развертывания, - стереотипы артефактов и узлов, - связи между узлами и артефактами.
10	<p>Использование UML для проектирования параллельных приложений</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - параллельные действия и события на диаграмме деятельности, - обработка событий, отправка сигналов, - объекты на диаграмме деятельности, - область с прерыванием выполнения действий.
11	<p>Параллельные подавтоматы на диаграмме состояний</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - составное состояние, - вход и выход из ортогонального состояния, - взаимодействие подавтоматов, - псевдосостояние предистории.
12	<p>Разделы диаграммы последовательности</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - альтернативные сценарии,

№ п/п	Тематика лекционных занятий / краткое содержание
	- параллельные фрагменты, - циклы.
13	Жизненный цикл программного обеспечения Рассматриваемые вопросы: - стандарты жизненного цикла: ISO/IEC 12207, - Oracle Unified Method, - ГОСТ34, - Rational Unified Process.
14	Модели жизненного цикла Рассматриваемые вопросы: - каскадная и спиральная модели, - достоинства и недостатки.
15	Гибкие методологии Рассматриваемые вопросы: - Методология Agile, ценности и принципы.
16	Методология SCRAM Методология SCRAM. Рассматриваемые вопросы: - специалисты, - участвующие в разработке продукта, - обзор спринта, - ретроспектива, - практики экстремального программирования.

4.2. Занятия семинарского типа.

Лабораторные работы

№ п/п	Наименование лабораторных работ / краткое содержание
1	Разработка диаграммы вариантов использования на языке UML. Описание актеров и вариантов использования В результате выполнения лабораторной работы студент получает навык проектирования диаграммы вариантов использования в UML Designer
2	Разработка диаграммы вариантов использования на языке UML. Описание отношений В результате выполнения лабораторной работы студент получает навык проектирования диаграммы вариантов использования в UML Designer
3	Описание потоков событий для диаграммы вариантов использования В результате выполнения лабораторной работы студент получает навык описания потоков событий
4	Разработка диаграммы классов на языке UML. Описание атрибутов В результате выполнения лабораторной работы студент получает навык проектирования диаграммы классов в UML Designer
5	Разработка диаграммы классов на языке UML. Описание операций В результате выполнения лабораторной работы студент получает навык проектирования диаграммы классов в UML Designer

№ п/п	Наименование лабораторных работ / краткое содержание
6	Разработка диаграммы классов на языке UML. Описание отношений В результате выполнения лабораторной работы студент получает навык проектирования диаграммы классов в UML Designer
7	Разработка диаграмм деятельности на языке UML. Последовательность действий В результате выполнения лабораторной работы студент получает навык проектирования диаграммы деятельности в UML Designer
8	Разработка диаграмм деятельности на языке UML. Обработка событий В результате выполнения лабораторной работы студент получает навык описания событий на диаграмме деятельности в UML Designer
9	Разработка диаграмм деятельности на языке UML. Дорожки В результате выполнения лабораторной работы студент получает навык проектирования взаимодействия между подразделениями и актерами на диаграмме деятельности в UML Designer
10	Разработка диаграмм деятельности на языке UML. Объекты В результате выполнения лабораторной работы студент получает навык изображения объектов и их состояний на диаграмме деятельности в UML Designer
11	Разработка диаграммы состояний на языке UML. Состояния В результате выполнения лабораторной работы студент получает навык проектирования диаграммы состояний в UML Designer
12	Разработка диаграммы состояний на языке UML. Переходы между состояниями В результате выполнения лабораторной работы студент получает навык проектирования диаграммы состояний в UML Designer
13	Разработка диаграммы состояний на языке UML. Условия при выполнении переходов В результате выполнения лабораторной работы студент получает навык проектирования диаграммы состояний в UML Designer
14	Разработка диаграммы состояний на языке UML. Функции при выполнении переходов В результате выполнения лабораторной работы студент получает навык проектирования диаграммы состояний в UML Designer
15	Разработка диаграммы состояний на языке UML. Деятельность ENTRY внутри состояния В результате выполнения лабораторной работы студент получает навык проектирования диаграммы состояний в UML Designer
16	Разработка диаграммы состояний на языке UML. Деятельность DO внутри состояния В результате выполнения лабораторной работы студент получает навык проектирования диаграммы состояний в UML Designer
17	Разработка диаграммы состояний на языке UML. Деятельность EXIT внутри состояния В результате выполнения лабораторной работы студент получает навык проектирования диаграммы состояний в UML Designer
18	Разработка диаграмм последовательности на языке UML. Описание объектов В результате выполнения лабораторной работы студент получает навык проектирования диаграммы последовательности в UML Designer
19	Разработка диаграмм последовательности на языке UML. Взаимодействие объектов В результате выполнения лабораторной работы студент получает навык проектирования диаграммы последовательности в UML Designer

№ п/п	Наименование лабораторных работ / краткое содержание
20	Разработка диаграмм последовательности на языке UML. Создание и уничтожение объектов В результате выполнения лабораторной работы студент получает навык проектирования диаграммы последовательности в UML Designer
21	Разработка диаграмм последовательности на языке UML. Рекурсивные вызовы В результате выполнения лабораторной работы студент получает навык проектирования диаграммы последовательности в UML Designer
22	Разработка диаграмм последовательности на языке UML. Описание разделов В результате выполнения лабораторной работы студент получает навык проектирования диаграммы последовательности с использованием разделов в UML Designer
23	Разработка диаграммы компонентов на языке UML В результате выполнения лабораторной работы студент получает навык проектирования диаграммы компонентов в UML Designer
24	Разработка диаграммы размещения на языке UML В результате выполнения лабораторной работы студент получает навык проектирования диаграммы размещения в UML Designer

4.3. Самостоятельная работа обучающихся.

№ п/п	Вид самостоятельной работы
1	Изучение документации по языку UML и системе UML Designer
2	Анализ и дополнительная проработка лекционного материала
3	Подготовка к лабораторным работам
4	Изучение учебной литературы из приведенных источников
5	Выполнение курсовой работы.
6	Подготовка к промежуточной аттестации.
7	Подготовка к текущему контролю.

4.4. Примерный перечень тем курсовых работ

Разработать информационную систему и ее проект на языке UML.

1. Система совместной подготовки документов
2. Электронная таблица
3. Система тестирования
4. Экспертная система диагностики заболевания пациента
5. Система обмена сообщениями
6. Интернет-магазин. Заказ товаров
7. Интернет-магазин. Доставка товаров
8. Научно-исследовательский институт

9. Магазин и склад
10. Железная дорога
11. Нагрузка кафедры
12. Ресторан
13. Методическая работа преподавателя
14. Больница. Медосмотр.
15. Завод. Выпуск изделий
16. Аренда велосипедов
17. Аэродром. Выполнение тренировочных полетов
18. Новостной сайт
19. Авиабилеты
20. Аэропорт
21. Банк
22. Расписание занятий на семестр
23. Приемная комиссия
24. Сессия
25. Библиотека
26. Турфирма

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

№ п/п	Библиографическое описание	Место доступа
1	М. А. Давыдовский, М. Н. Никольская. Проектирование программной системы в UML Designer. М. : РУТ(МИИТ), 2019. - 129 с.	https://elibrary.ru/download/elibrary_41662371_91637704.pdf , https://reader.lanbook.com/book/175651#1 (дата обращения: 05.04.2025). - Текст: электронный.
2	Титов А. Н., Тагиева Р. Ф. Введение в Tkinter. Разработка графических интерфейсов в Python: учебно-методическое пособие. - Казанский	https://e.lanbook.com/book/412466 (дата обращения: 16.03.2026). - Текст: электронный.

	национальный исследовательский технологический университет, 2023. – 100 с. ISBN 978-5-7882-3340-6	
3	Федькова Н.А. Современные технологии разработки программного обеспечения. – Брянск : Издательство Брянский ГАУ, 2022. — 58 с.	https://reader.lanbook.com/book/305087#2 (дата обращения: 05.04.2025). - Текст: электронный.
4	Унгер А.Ю. Объектно-ориентированное программирование: учебник . – МИРЭА - Российский технологический университет, 2023.-102 с. ISBN 978-5-7339-2051-1	https://reader.lanbook.com/book/398276 (дата обращения: 05.04.2025). - Текст: электронный.

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

- Научная электронная библиотека (<http://elibrary.ru>)
- Электронно-библиотечная система Научно-технической библиотеки МИИТ (<http://library.miit.ru>)
- Материалы по информационным технологиям (www.citforum.ru)
- Сайт системы автоматизированного проектирования UML Designer (<http://www.uml designer.org>)
- Стандарты языка UML (<http://www.omg.org/spec/UML>)

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

- Microsoft Windows
- Microsoft Office
- Интернет-браузер (Yandex и др.)
- Язык программирования Java,
- Язык программирования C++,
- UML Designer (лицензия – свободно распространяемое ПО).
- Eclipse (лицензия – свободно распространяемое ПО)

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

Учебная аудитория для проведения учебных занятий (занятий лекционного типа, практических занятий, лабораторных работ, курсового проектирования (выполнения курсовых работ), текущего контроля и промежуточной аттестации):

- компьютер преподавателя, проектор, экран проекционный, рабочие станции студентов, маркерная доска.

Аудитория подключена к сети «Интернет»

9. Форма промежуточной аттестации:

Зачет в 3 семестре.

Курсовая работа в 3 семестре.

10. Оценочные материалы.

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

Авторы:

доцент, доцент, к.н. кафедры
«Вычислительные системы, сети и
информационная безопасность»

М.А. Давыдовский

Согласовано:

Заведующий кафедрой ВССиИБ

Б.В. Желенков

Председатель учебно-методической
комиссии

Н.А. Андриянова