

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
09.03.01 Информатика и вычислительная техника,
утвержденной первым проректором РУТ (МИИТ)
Тимониным В.С.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Основы проектирования ПО

Направление подготовки: 09.03.01 Информатика и вычислительная техника

Направленность (профиль): Технологии разработки 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: 01.09.2025

1. Общие сведения о дисциплине (модуле).

Целью курса «Основы проектирования ПО» является обучение студентов базовым принципам проектирования программного обеспечения. В рамках курса учащиеся получают знания основных принципов, концепций и методов проектирования программного обеспечения.

На практических занятиях у обучающихся формируются навыки применением принципов проектирования ПО, применения принципов SOLID, шаблонов проектирования. Особое внимание уделяется понятию тестируемости и поддерживаемости программного обеспечения при его проектировании.

Задачей дисциплины является обучить практическим навыкам организации сбора, обработки и управления данными и информацией для ведения процесса проектирования и ознакомить с актуальными российскими стандартами в области информационных технологий

2. Планируемые результаты обучения по дисциплине (модулю).

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

ПК-6 - Способен проектировать разрабатываемый программный продукт на основе принятых методологий и практик для корпоративного рынка.;

ПК-8 - Способен разрабатывать программные продукты используя современные методологии и практики для корпоративного рынка.

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

Знать:

- жизненный цикл программного обеспечения;
- модели жизненного цикла ПО;
- основные принципы архитектуры ПО;
- виды программного обеспечения и особенности их архитектуры;
- различные виды моделей и диаграмм в UML;
- принципы SOLID (SRP, OCP, LSP, ISP, DIP);
- базовые шаблоны проектирования и антипаттерны;
- архитектурные шаблоны проектирования и стили, такие как MVC, MVP/MVVM, MVI;

Уметь:

- анализировать требования заказчика и определять их осуществимость;
- использовать инструменты проектирования ПО для создания моделей и диаграмм;
- применять принципы абстракции, инкапсуляции, декомпозиции и другие методы проектирования ПО;
- проектировать с учетом принципов SOLID (SRP, OCP, LSP, ISP, DIP);
- применять базовые шаблоны проектирования для решения типовых задач;
- учитывать требования к тестируемости и поддерживаемости программного обеспечения при его проектировании.

Владеть:

- навыками проектирования программного обеспечения на основе требований заказчика;
- навыками применения архитектурных шаблонов и стилей при проектировании и разработки программного обеспечения;
- навыками проектирования и разработки тестируемого и поддерживаемого программного обеспечения.

3. Объем дисциплины (модуля).

3.1. Общая трудоемкость дисциплины (модуля).

Общая трудоемкость дисциплины (модуля) составляет 3 з.е. (108 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 №7
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 60 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 проектирование ПО.</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none">- жизненный цикл программного обеспечения и роль проектирования;- модели жизненного цикла;- требования и осуществимость;- классификация требований заказчика;- понятие качества программного продукта.
2	<p>Введение в архитектуру ПО.</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none">- понятие архитектуры;- виды программного обеспечения и особенности их архитектуры;- инструменты проектирования ПО;- модели и практики проектирования ПО.
3	<p>Проектирование, моделирование и UML.</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none">- виды моделей, моделирование архитектуры с использованием UML;- диаграммы классов;- диаграммы пакетов;- диаграммы объектов;- диаграммы компонентов.
4	<p>Методы проектирования ПО.</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none">- абстракция, инкапсуляция, декомпозиция (модульность);- иерархическая организация, типизация, повторное использование;- проектирование по контракту;- основные принципы проектирования, проектирование с учетом будущих изменений;- типовые причины перепроектирования;- проектирование в соответствии с интерфейсом, а не с реализацией.
5	<p>SOLID.</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none">- Single Responsibility Principle (SRP);- Open-Closed Principle (OCP);- Liskov Substitution Principle (LSP);

№ п/п	Тематика лекционных занятий / краткое содержание
	<ul style="list-style-type: none"> - Interface Segregation Principle (ISP); - Dependency Inversion Principle (DIP).
6	<p>Шаблоны проектирования.</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - понятие шаблонов проектирования; - базовые порождающие шаблоны проектирования; - базовые структурные шаблоны проектирования; - базовые поведенческие шаблоны проектирования; - «антипаттерны».
7	<p>Архитектурные шаблоны проектирования и стили.</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - понятие архитектурных шаблонов проектирования и архитектурного стиля; - шаблон Model-View-Controller (MVC); - шаблон Model-View-Presenter/Model-View-ViewModel (MVP/MVVM); - шаблон MVI.
8	<p>Проектирование с учетом тестируемости и поддерживаемости программного обеспечения.</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - тестируемость, методы разработки тестируемого программного обеспечения; - Test-Driven Development (TDD); - роль модульного и интеграционного тестирования при разработке программного обеспечения; - роль систем контроля версий; - высокое зацепление (High Cohesion); - низкая связанность (Low Coupling).

4.2. Занятия семинарского типа.

Практические занятия

№ п/п	Тематика практических занятий/краткое содержание
1	<p>Модели жизненного цикла ПО.</p> <p>В результате выполнения практической работы студент получает навык анализа и выбора подходящей модели жизненного цикла ПО для конкретного проекта.</p>
2	<p>Анализ требований заказчика.</p> <p>В результате выполнения практической работы студент получает навык анализа и классификации требований заказчика для дальнейшего проектирования ПО.</p>
3	<p>Моделирование архитектуры программного продукта с использованием UML.</p> <p>В результате выполнения практической работы студент получает навык создания диаграмм классов, пакетов, объектов и компонентов с использованием UML.</p>
4	<p>SOLID.</p> <p>В результате выполнения практической работы студент получает навык проектирования гибкой и расширяемой архитектуры с использованием SOLID-принципов.</p>
5	<p>Шаблоны проектирования.</p> <p>В результате выполнения практической работы студент получает навык выбора и применения подходящего шаблона проектирования для решения конкретной задачи.</p>
6	<p>Архитектурные стили.</p> <p>В результате выполнения практической работы студент получает навык создания и применения архитектурных шаблонов проектирования и стилей.</p>

№ п/п	Тематика практических занятий/краткое содержание
7	Проектирование с учетом тестируемости программного обеспечения. В результате выполнения практической работы студент получает навык проектирования тестируемого программного обеспечения и создания модульных и интеграционных тестов.
8	Проектирование с учетом поддерживаемости программного обеспечения. В результате выполнения практической работы студент получает навык создания чистого кода, устранения дублирования и улучшения читаемости кода.
9	Основные принципы проектирования. В результате выполнения практической работы студент получает навык проектирования в соответствии с интерфейсом и перепроектирования с учетом будущих изменений.
10	Методы проектирования ПО. В результате выполнения практической работы студент получает навык работы с различными методами проектирования ПО.
11	Test-Driven Development (TDD). В результате выполнения практической работы студент получает навык использования Test-Driven Development (TDD).
12	Высокое зацепление, низкая связность. В результате выполнения практической работы студент изучает принципы объектно-ориентированного программирования, направленные на улучшение качества кода и его поддержки.

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	Подготовка к промежуточной аттестации.
5	Подготовка к текущему контролю.

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

№ п/п	Библиографическое описание	Место доступа
1	Кумагина, Е. А. Модели жизненного цикла и технологии проектирования программного обеспечения : учебно-методическое пособие / Е. А. Кумагина, Е. А. Неймарк. — Нижний Новгород : ННГУ им. Н. И. Лобачевского, 2016. — 41 с. — Текст : электронный Учебно-методическое издание	https://e.lanbook.com/book/153391 (дата обращения: 16.04.2025)
2	Нафикова, А. Р. Объектно-ориентированный анализ и проектирование программного обеспечения на языке UML : учебное пособие / А.	https://e.lanbook.com/book/219221 (дата обращения: 16.04.2025)

	Р. Нафикова. — Уфа : БГПУ имени М. Акмуллы, 2022. — 118 с. — ISBN 978-5-907475-48-9. — Текст : электронный Учебное пособие	
3	Дэвис, К. Шаблоны проектирования для облачной среды : руководство / К. Дэвис ; перевод с английского Д. А. Беликова.. — Москва : ДМК Пресс, 2020. — 388 с. — ISBN 978-5-97060-807-4. — Текст : электронный	https://e.lanbook.com/book/140593 (дата обращения: 16.04.2025)
4	Крючкова, Е. Н. Объектно-ориентированное программирование: Архитектурное проектирование и паттерны программирования : учебно-методическое пособие / Е. Н. Крючкова, С. М. Старолетов. — Барнаул : АлтГТУ, 2020. — 180 с. — Текст : электронный Учебно-методическое издание	https://e.lanbook.com/book/292790 (дата обращения: 16.04.2025)

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

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

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

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

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

Браузер с доступом в интернет

Пакет офисных приложений

Drawio Desktop

IntelliJ IDEA Community Edition

Visual Studio Code

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

Учебные аудитории для проведения учебных занятий, оснащенные компьютерной техникой и наборами демонстрационного оборудования.

Для практических занятий – наличие персональных компьютеров вычислительного класса.

9. Форма промежуточной аттестации:

Зачет в 7 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»

Е.А. Заманов

Согласовано:

Заведующий кафедрой ЦГУТП

В.Е. Нутович

Председатель учебно-методической
комиссии

Н.А. Андриянова