

**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)**

**Разработка корпоративных приложений**

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).

Целями освоения данной дисциплины являются получение базовых, теоретических знаний о разработке корпоративных приложений с использованием паттернов проектирования, практики их применения и реализации на языках высокого уровня.

Основной задачей изучения дисциплины является формирование базовых представлений, знаний и умений о ключевых принципах организации кода, порождающих, структурных паттернах и паттернах поведения, шаблонах организации объектно-ориентированного кода, об антипаттернах, причинах их возникновения, связанных с ними проблемами и путями решения.

### 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-7** - Способен разрабатывать программные продукты под разные платформы для корпоративного рынка.

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

#### **Знать:**

- ключевые принципы хорошего кода;
- UML диаграммы и основные компоненты;
- различные типы паттернов и практики их применения;
- принципы SOLID.

#### **Уметь:**

- применять техники рефакторинга в существующем коде;
- применять традиционные подходы к организации кода;
- применять традиционные принципы хорошего кода.

#### **Владеть:**

- навыками рефакторинга с применением различных типов паттернов;
- навыками применения принципов SOLID.

### 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):	64	64
В том числе:		
Занятия лекционного типа	32	32
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 44 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	Введение в паттерны проектирования. Рассматриваемые вопросы: - знакомство с паттернами проектирования. Ключевые принципы хорошего кода; - классификация паттернов проектирования. «Банда четырех» (Gang of Four, GoF); - антипаттерны в программировании и проектировании архитектуры; - рефакторинг.

№ п/п	Тематика лекционных занятий / краткое содержание
2	<p>Unified Modeling Language.</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- унифицированный язык моделирования (Unified Modeling Language, UML);</li> <li>- UML диаграммы.</li> </ul>
3	<p>Традиционные подходы к организации кода. Порождающие паттерны.</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- фабричный метод (Factory Method);</li> <li>- абстрактная фабрика (Abstract Factory);</li> <li>- одиночка (Singleton);</li> <li>- прототип (Prototype);</li> <li>- строитель (Builder).</li> </ul>
4	<p>Традиционные подходы к организации кода. Структурные паттерны.</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- адаптер (Adapter);</li> <li>- декоратор (Decorator);</li> <li>- заместитель (Proxy);</li> <li>- компоновщик (Composite);</li> <li>- мост (Bridge);</li> <li>- приспособленец (Flyweight);</li> <li>- фасад (Facade).</li> </ul>
5	<p>Традиционные подходы к организации кода. Паттерны поведения.</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- интерпретатор (Interpreter);</li> <li>- шаблонный метод (Template Method);</li> <li>- итератор (Iterator);</li> <li>- команда (Command);</li> <li>- наблюдатель (Observer);</li> <li>- посетитель (Visitor);</li> <li>- посредник (Mediator);</li> <li>- состояние (State);</li> <li>- стратегия (Strategy);</li> <li>- хранитель (Memento);</li> <li>- цепочка обязанностей (Chain of Responsibility).</li> </ul>
6	<p>Дополнительные шаблоны организации объектно-ориентированного кода.</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- основные шаблоны проектирования: Delegation (Делегирование), Interface (Интерфейс), Abstract Superclass (Абстрактный суперкласс), Interface and Abstract Class (Интерфейс и абстрактный класс), Immutable (Неизменный), Marker Interface (Маркер-интерфейс);</li> <li>- порождающие шаблоны проектирования: Object Pool (Пул объектов);</li> <li>- разделяющие шаблоны проектирования: Filter (Фильтр), Read-Only Interface (Интерфейс, предназначенный только для чтения);</li> <li>- структурные шаблоны проектирования: Dynamic Linkage (Динамическая компоновка), Virtual Proxy (Виртуальный заместитель), Cache Management (Управление кэшем);</li> <li>- поведенческие шаблоны проектирования: Command (Команда), Little Language (Малый язык), Snapshot (Моментальный снимок), Null Object (Нулевой объект);</li> <li>- шаблоны проектирования для конкурирующих операций: Single Threaded Execution (Однопоточное выполнение), Lock Object (Объект блокировки), Guarded Suspension (Охраняемая приостановка), Balking (Отмена), Scheduler (Планировщик), ReadjWrite Lock (Блокировка чтения/записи), Producer/Consumer (Производитель-потребитель), Two-Phase Termination (Двухфазное завершение), Double Buffering (Двойная буферизация), Asynchronous Processing (Асинхронная обработка), Future (Будущее).</li> </ul>

№ п/п	Тематика лекционных занятий / краткое содержание
7	<p><b>Ключевые принципы хорошего кода.</b></p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- «Не повторяй себя» (DRY – Don’t repeat yourself);</li> <li>- «Делайте вещи проще» (KISS – keep it simple stupid);</li> <li>- «Вам это не понадобится» (YAGNI – You ain’t gonna need it);</li> <li>- «Глобальное проектирование прежде всего» (BDUF – Big Design Up Front);</li> <li>- «Избегайте преждевременной оптимизации» (APO – Avoid Premature Optimization);</li> <li>- принципы SOLID: принцип единственной ответственности (single responsibility principle), принцип открытости/закрытости (open-closed principle), принцип подстановки Барбары Лисков (Liskov substitution principle), принцип разделения интерфейсов (interface segregation principle), принцип инверсии зависимостей (dependency inversion principle).</li> </ul>
8	<p><b>Принципы SOLID.</b></p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- принципы SOLID: принцип единственной ответственности (single responsibility principle), принцип открытости/закрытости (open-closed principle), принцип подстановки Барбары Лисков (Liskov substitution principle), принцип разделения интерфейсов (interface segregation principle), принцип инверсии зависимостей (dependency inversion principle).</li> </ul>
9	<p><b>Антипаттерны архитектуры и проектирования.</b></p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- ад зависимостей (Dependency hell);</li> <li>- божественный объект (God object);</li> <li>- золотой молоток (Golden hammer);</li> <li>- одиночка (Singleton);</li> <li>- полтергейст (Poltergeist);</li> <li>- проблема йо йо (Yo-yo problem).</li> </ul>

## 4.2. Занятия семинарского типа.

### Лабораторные работы

№ п/п	Наименование лабораторных работ / краткое содержание
1	<p><b>Unified Modeling Language.</b></p> <p>В результате выполнения лабораторной работы студент получает навык UML моделирования компонентов архитектуры, алгоритмов и элементов программного кода.</p>
2	<p><b>Рефакторинг.</b></p> <p>В результате выполнения лабораторной работы студент получает навык рефакторинга программного кода на типовых кейсах.</p>
3	<p><b>Порождающие паттерны.</b></p> <p>В результате выполнения лабораторной работы студент получает навык применения порождающих паттернов на типовых кейсах.</p>
4	<p><b>Структурные паттерны.</b></p> <p>В результате выполнения лабораторной работы студент получает навык применения структурных паттернов на типовых кейсах.</p>
5	<p><b>Паттерны поведения.</b></p> <p>В результате выполнения лабораторной работы студент получает навык применения поведенческих паттернов на типовых кейсах.</p>
6	<p><b>Организация объектно-ориентированного кода.</b></p> <p>В результате выполнения лабораторной работы студент знакомится с подходами организации объектно-ориентированного кода на типовых проектах.</p>

№ п/п	Наименование лабораторных работ / краткое содержание
7	Ключевые принципы хорошего кода. В результате выполнения лабораторной работы студент знакомится с принципиальными подходами к программной инженерии профессионального качества.
8	Антипаттерны. В результате выполнения лабораторной работы студент знакомится как с антипаттернами, так и практическими кейсами, в которых паттерны становятся антипаттернами.
9	Организация объектно-ориентированного кода. В результате выполнения лабораторной работы студент знакомится с подходами организации объектно-ориентированного кода на типовых проектах.
10	Принципы SOLID. В результате выполнения лабораторной работы студент получает навык применения принципов SOLID на типовых кейсах.

#### 4.3. Самостоятельная работа обучающихся.

№ п/п	Вид самостоятельной работы
1	Работа с лекционным материалом.
2	Работа с литеартурой.
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	Э. Гамма, Р. Хелм, Р. Джонсон, Дж. Влиссидес Паттерны объектно-ориентированного проектирования. Санкт-Петербург : Питер, 2021. - 448 с. - ISBN 978-5-4461-1595-2.	<a href="https://books.yandex.ru/books/oYgnESD2/read-online?ysclid=mkmkz1uq1q590693192">https://books.yandex.ru/books/oYgnESD2/read-online?ysclid=mkmkz1uq1q590693192</a>
2	Унгер, А. Ю. Паттерны проектирования на C++ : учебное пособие / А. Ю. Унгер. — Москва : РТУ МИРЭА, 2023. — 74 с. — ISBN 978-5-7339-1753-5.	<a href="https://e.lanbook.com/book/368645">https://e.lanbook.com/book/368645</a> (дата обращения: 20.01.2026)
3	«Баланов, А. Н. Комплексное руководство по разработке: от мобильных приложений до веб-технологий : учебное пособие для вузов / А. Н. Баланов. — 2-е изд., стер. — Санкт-Петербург : Лань, 2025. — 412 с.	<a href="https://e.lanbook.com/book/478178">https://e.lanbook.com/book/478178</a> (дата обращения: 20.01.2026).

	— ISBN 978-5-507-53193-6.» (Баланов, А. Н. Комплексное руководство по разработке: от мобильных приложений до веб-технологий : учебное пособие для вузов / А. Н. Баланов. — 2-е изд., стер. — Санкт-Петербург : Лань, 2025. — ISBN 978-5-507-53193-6	
4	Ф. Пикус Идиомы и паттерны проектирования в современном C++ : руководство. Москва : ДМК Пресс, 2020. - 452 с. - ISBN 978-5-97060-786-2.	<a href="https://e.lanbook.com/book/140598">https://e.lanbook.com/book/140598</a>
5	Эффективное моделирование бизнес-процессов с применением UML: стратегии и инструменты : учебное пособие / Д. В. Шлаев, А. А. Сорокин, С. В. Аникуев, Ю. В. Орел. – Ставрополь : Ставропольский государственный аграрный университет, 2024. - 109 с. - Текст : электронный.	<a href="https://znanium.ru/read?id=473019">https://znanium.ru/read?id=473019</a>

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

Электронно-библиотечная система Научно-технической библиотеки РУТ(МИИТ) (<http://library.miit.ru/>)

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

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

Браузер Microsoft Internet Explorer или его аналоги

Пакет офисных программ Microsoft Office или его аналоги

.NET 6

Java 17

Microsoft Visual Studio CE

JetBrains IntelliJ IDEA Community Edition

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 ЦТУТП

В.Е. Нутович

Председатель учебно-методической  
комиссии

Н.А. Андриянова