

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 магистратуры
по направлению подготовки
09.04.01 Информатика и вычислительная техника,
утвержденной первым проректором РУТ (МИИТ)
Тимониным В.С.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Паттерны проектирования

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: 08.06.2022

1. Общие сведения о дисциплине (модуле).

Целями освоения данной дисциплины являются получение базовых, теоретических знаний о паттернах проектирования, практика 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 архитектуру сложных программных продуктов;

ПК-7 - Способен интегрировать программные компоненты в существующие программные продукты.

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 S.O.L.I.D.

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 S.O.L.I.D.

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 №1
Контактная работа при проведении учебных занятий (всего):	54	54
В том числе:		
Занятия лекционного типа	18	18
Занятия семинарского типа	36	36

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 126 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 паттерны проектирования.</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none">- знакомство с паттернами проектирования. Ключевые принципы хорошего кода;- классификация паттернов проектирования. «Банда четырех» (Gang of Four, GoF);- антипаттерны в программировании и проектировании архитектуры;- рефакторинг.
2	<p>Unified Modeling Language.</p> <p>Рассматриваемые вопросы:</p>

№ п/п	Тематика лекционных занятий / краткое содержание
	<ul style="list-style-type: none"> - унифицированный язык моделирования (Unified Modeling Language, UML); - UML диаграммы.
3	<p>Традиционные подходы к организации кода. Порождающие паттерны.</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - фабричный метод (Factory Method); - абстрактная фабрика (Abstract Factory); - одиночка (Singleton); - прототип (Prototype); - строитель (Builder).
4	<p>Традиционные подходы к организации кода. Структурные паттерны.</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - адаптер (Adapter); - декоратор (Decorator); - заместитель (Proxy); - компоновщик (Composite); - мост (Bridge); - приспособленец (Flyweight); - фасад (Facade).
5	<p>Традиционные подходы к организации кода. Паттерны поведения.</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - интерпретатор (Interpreter); - шаблонный метод (Template Method); - итератор (Iterator); - команда (Command); - наблюдатель (Observer); - посетитель (Visitor); - посредник (Mediator); - состояние (State); - стратегия (Strategy); - хранитель (Memento); - цепочка обязанностей (Chain of Responsibility).
6	<p>Дополнительные шаблоны организации объектно-ориентированного кода.</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - 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); - порождающие шаблоны проектирования: Object Pool (Пул объектов); - разделяющие шаблоны проектирования: Filter (Фильтр), Read-Only Interface (Интерфейс, предназначенный только для чтения); - структурные шаблоны проектирования: Dynamic Linkage (Динамическая компоновка), Virtual Proxy (Виртуальный заместитель), Cache Management (Упрамение кэшем); - поведенческие шаблоны проектирования: Command (Команда), Little Language (Малый язык), Snapshot (Моментальный снимок), Null Object (Нулевой объект); - 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).
7	<p>Ключевые принципы хорошего кода.</p> <p>Рассматриваемые вопросы:</p>

№ п/п	Тематика лекционных занятий / краткое содержание
	<ul style="list-style-type: none"> - «Не повторяй себя» (DRY – Don't repeat yourself); - «Делайте вещи проще» (KISS – keep it simple stupid); - «Вам это не понадобится» (YAGNI – You ain't gonna need it); - «Глобальное проектирование прежде всего» (BDUF – Big Design Up Front); - «Избегайте преждевременной оптимизации» (APO – Avoid Premature Optimization); - принципы S.O.L.I.D.: 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).
8	<p>Принципы S.O.L.I.D.</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - принципы S.O.L.I.D.: 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).
9	<p>Антипаттерны архитектуры и проектирования.</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> - ад зависимостей (Dependency hell); - божественный объект (God object); - золотой молоток (Golden hammer); - одиночка (Singleton); - полтерgeist (Poltergeist); - проблема йо йо (Yo-yo problem).

4.2. Занятия семинарского типа.

Лабораторные работы

№ п/п	Наименование лабораторных работ / краткое содержание
1	<p>Unified Modeling Language.</p> <p>В результате выполнения лабораторной работы студент получает навык UML моделирования компонентов архитектуры, алгоритмов и элементов программного кода.</p>
2	<p>Порождающие паттерны.</p> <p>В результате выполнения лабораторной работы студент получает навык применения пораждающих паттернов на типовых кейсах.</p>
3	<p>Структурные паттерны.</p> <p>В результате выполнения лабораторной работы студент получает навык применения структурных паттернов на типовых кейсах.</p>
4	<p>Паттерны поведения.</p> <p>В результате выполнения лабораторной работы студент получает навык применения поведенческих паттернов на типовых кейсах.</p>
5	<p>Принципы S.O.L.I.D.</p> <p>В результате выполнения лабораторной работы студент получает навык применения принципов S.O.L.I.D. на типовых кейсах.</p>

Практические занятия

№ п/п	Тематика практических занятий/краткое содержание
1	Рефакторинг. В результате выполнения практической работы студент получает навык рефакторинга программного кода на типовых кейсах.
2	Порождающие паттерны. В результате выполнения практической работы студент знакомится с порождающими паттернами на типовых кейсах.
3	Структурные паттерны. В результате выполнения практической работы студент знакомится с структурными паттернами на типовых кейсах.
4	Паттерны поведения. В результате выполнения практической работы студент знакомится с поведенческими паттернами на типовых кейсах.
5	Организация объектно-ориентированного кода. В результате выполнения практической работы студент знакомится с подходами организации объектно-ориентированного кода на типовых проектах.
6	Ключевые принципы хорошего кода. В результате выполнения практической работы студент знакомится с принципиальными подходами к программной инженерии профессионального качества.
7	Антипаттерны. В результате выполнения практической работы студент знакомится как с антипаттернами, так и практическими кейсами, в которых паттерны становятся антипаттернами.

4.3. Самостоятельная работа обучающихся.

№ п/п	Вид самостоятельной работы
1	Изучение рекомендованной литературы.
2	Подготовка к лабораторным и практическим занятиям.
3	Выполнение курсового проекта.
4	Подготовка к промежуточной аттестации.
5	Подготовка к текущему контролю.

4.4. Примерный перечень тем курсовых проектов

1. Паттерн «Абстрактная фабрика». Моделирование и анализ кейса до и после рефакторинга.
2. Паттерн «Фасад». Моделирование и анализ кейса до и после рефакторинга.
3. Паттерн «Строитель». Моделирование и анализ кейса до и после рефакторинга.
4. Паттерн «Декоратор». Моделирование и анализ кейса до и после рефакторинга.
5. Паттерн «Наблюдатель». Моделирование и анализ кейса до и после рефакторинга.

6. Паттерн «Посредник». Моделирование и анализ кейса до и после рефакторинга.

7. Паттерн «Стратегия». Моделирование и анализ кейса до и после рефакторинга.

8. Паттерн «Хранитель». Моделирование и анализ кейса до и после рефакторинга.

9. Паттерн «Прокси». Моделирование и анализ кейса до и после рефакторинга.

10. Паттерн «Фильтр». Моделирование и анализ кейса до и после рефакторинга.

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.	https://books.yandex.ru/books/oYgnESD2/read-online?ysclid=mkmkz1uq1q590693192
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.	https://e.lanbook.com/book/368645 (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. — 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	https://e.lanbook.com/book/478178 (дата обращения: 20.01.2026).
4	Ф. Пикус Идиомы и паттерны проектирования в современном C++ :	https://e.lanbook.com/book/140598

	руководство. Москва : ДМК Пресс, 2020. - 452 с. - ISBN 978-5-97060-786-2.	
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.	https://znanium.ru/read?id=473019

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

<http://library.miit.ru/> – электронно-библиотечная система Научно-технической библиотеки МИИТ;

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

<https://www.microsoft.com/ru-ru/learning/training.aspx> – учебные курсы Microsoft.

<https://docs.microsoft.com/ru-ru/azure/architecture/patterns> – документация по продукту Microsoft Azure

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

Microsoft Office

.NET 6

Java 17

Microsoft Visual Studio CE

JetBrains IntelliJ IDEA Community Edition

Браузер с выходом в интернет

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

Для проведения занятий лекционного типа требуются аудитории, оснащенные компьютерной техникой и наборами демонстрационного оборудования.

Для практических занятий – наличие персональных компьютеров вычислительного класса.

9. Форма промежуточной аттестации:

Курсовой проект в 1 семестре.

Экзамен в 1 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.Дутова

Согласовано:

Заведующий кафедрой ЦГУТП

В.Е. Нутович

Председатель учебно-методической
комиссии

Н.А.Клычева