

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



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

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Архитектура программного обеспечения

Направление подготовки: 09.03.01 Информатика и вычислительная техника

Направленность (профиль): IT-сервисы и технологии обработки данных на транспорте (Российско-Китайская программа)

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: 937226
Подписал: руководитель образовательной программы
Проневич Ольга Борисовна
Дата: 15.06.2026

1. Общие сведения о дисциплине (модуле).

Краткая аннотация дисциплины (модуля) (как правило, описываются основные цели и задачи дисциплины(модуля).

2. Планируемые результаты обучения по дисциплине (модулю).

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

ОПК-6 - Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения;

ПК-3 - Способен осуществлять разработку требований и проектирование программного обеспечения.

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

Знать:

существующие языки программирования и их методологии, принципы разработки процедурных программ и подпрограмм.

Уметь:

формализовывать алгоритмическую и функциональную структуру программного обеспечения; разрабатывать блок схемы алгоритмов, подпрограмм.

Уметь:

основными инструментами разработки, отладки, тестирования и распространения программного обеспечения.

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 №5

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 116 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 архитектуру ПО. Рассматриваемые вопросы: - Этапы развития технологий проектирование - Обзор современных архитектур
2	Различные подходы к разработке программного обеспечения. Рассматриваемые вопросы: - Жизненный цикл программного продукта. - Разработка технического задания.
3	Гибкие методологии и архитектура ПО. Рассматриваемые вопросы: - Jobs To Be Done. Технические средства и инструменты обеспечивающие процесс проектирования ПО.
4	Архитектура программного обеспечения. - Архитектура программного обеспечения при объектноориентированном подходе к программированию. - UML. - Принципы ООП.
5	Паттерны программирования. Рассматриваемые вопросы: - Основные принципы

№ п/п	Тематика лекционных занятий / краткое содержание
6	Паттерны Рассматриваемые вопросы: - Паттерны Фабричный метод и Фабрика. - Паттерн Репозиторий.
7	Поведенческие паттерны.
8	Архитектурные паттерны.

4.2. Занятия семинарского типа.

Практические занятия

№ п/п	Тематика практических занятий/краткое содержание
1	Анализ нефункциональных требований и выбор архитектурного стиля Рассматриваемые вопросы: - Классификация нефункциональных требований (масштабируемость, отказоустойчивость, производительность) - Влияние требований к задержкам и пропускной способности на архитектуру - Матрица принятия решений: когда выбирать монолит, микросервисы или serverless - Создание архитектурных сценариев (по методологии ATAM)
2	Проектирование монолитной архитектуры: слои и модули Рассматриваемые вопросы: - Сравнение строгой слоистой (strict layers) и ослабленной слоистой (relaxed layers) архитектур - Реализация сквозного функционала (логирование, безопасность) без нарушения слоев - Антипаттерны: «божественный объект», «газированная архитектура» - Практическое разделение проекта на уровни: Presentation > Business > Persistence
3	Микросервисная архитектура: декомпозиция по доменам Рассматриваемые вопросы: Методы идентификации границ сервисов (по доменам DDD, по бизнес-возможностям) - Стратегии декомпозиции: по поддоменам и по командам/запросам - Проблема распределенных транзакций (Saga, двухфазная фиксация) - Практикум: разбиение гипотетической системы на 5–7 микросервисов
4	Коммуникации в распределенных системах Рассматриваемые вопросы: - Синхронное vs асинхронное взаимодействие: HTTP/gRPC vs message broker - Паттерны: API Gateway, Backend for Frontend (BFF), агрегатор данных - Проблемы: сетевые задержки, частичные отказы, идемпотентность запросов - Реализация retry с экспоненциальной задержкой и circuit breaker
5	Управление данными в архитектуре: CQRS и Event Sourcing Рассматриваемые вопросы: - Разделение команд и запросов (CQRS): простой и расширенный подходы - Event Sourcing: хранение состояния как последовательности событий - Снапшоты, воспроизведение событий, eventual consistency - Кейсы: когда CQRS/ES дает преимущества, а когда — излишняя сложность
6	Архитектурные паттерны отказоустойчивости Рассматриваемые вопросы: - Паттерны: Retry, Timeout, Circuit Breaker, Bulkhead, Fallback - Реализация Health Checks и обработки частичных отказов

№ п/п	Тематика практических занятий/краткое содержание
	- Graceful degradation и feature flags - Моделирование отказа одного компонента и восстановление системы
7	Оценка и рефакторинг архитектуры: технический долг Рассматриваемые вопросы: - Метрики связности и сцепления (coupling & cohesion): CBO, LCOM, D - Архитектурные антипаттерны: Big Ball of Mud, Golden Hammer, Vendor Lock-in - Матрица рисков изменений и инструменты анализа (Structure101, jQAssistant) - Практикум: рефакторинг плохих архитектурных решений на примере кода
8	Документирование архитектуры и архитектурные решения (ADR) Рассматриваемые вопросы: - Структура ADR (Architecture Decision Record) и виды архитектурных документов - Диаграммы: C4 model (Context, Container, Component, Code) - Подход ARC42: шаблон для описания архитектуры - Практикум: написание ADR для выбора базы данных или стиля взаимодействия

4.3. Самостоятельная работа обучающихся.

№ п/п	Вид самостоятельной работы
1	Работа с лекционным материалом
2	Работа с литературой
3	Подготовка к промежуточной аттестации.
4	Подготовка к текущему контролю.

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

№ п/п	Библиографическое описание	Место доступа
1	Голуб, Н. Р. Программное обеспечение ПК : учебное пособие / Н. Р. Голуб, В. В. Морозова, Р. М. Чудинский. — Воронеж : ВГПУ, 2025. — 84 с. — ISBN 978-5-907961-20-3.	https://e.lanbook.com/book/498539
2	Девницын, И. Н. Системное программное обеспечение : учебное пособие / И. Н. Девницын, Т. В. Гавриленко. — Сургут : СурГУ, 2024. — 44 с.	https://e.lanbook.com/book/422306

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

<https://habr.com/ru> - база знаний в виде статей, обзоров

<https://journal.tinkoff.ru/short/ai-for-all/> - база данных нейронных сетей

<https://vc.ru/services/916617-luchshie-neyroseti-bolshaya-podborka-iz-top-200-ii-generatorov-po-kategoriyam> - база данных нейронных сетей

<https://github.com/abalmumcu/bert-rest-api> - профессиональная платформа для командой работы над проектов (нейронная сеть bert)

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

<https://proglib.io/p/raspoznavanie-obektov-s-pomoshchyu-yolo-v3-na-tensorflow-2-0-2020-11-08> - профессиональная библиотека программистов

https://yandex.cloud/ru/blog/posts/2022/12/andrey-berger-and-yandex-cloud?utm_referrer=https%3A%2F%2Fyandex.ru%2F – библиотека профессиональных статей разработчиков Яндекс

<https://yandex.cloud/ru/blog> - библиотека профессиональных статей разработчиков Яндекс

<https://tproger.ru/translations/opencv-python-guide> - библиотека основных команд OpenCV

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

Офисный пакет приложений MicrosoftOffice

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

Компьютер преподавателя

Компьютеры студентов

экран для проектора, маркерная доска

Проектор

9. Форма промежуточной аттестации:

Экзамен в 5 семестре.

10. Оценочные материалы.

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

Авторы:

руководитель образовательной
программы

О.Б. Проневич

Согласовано:

Директор

Д.В. Паринов

Руководитель образовательной
программы

О.Б. Проневич

Председатель учебно-методической
комиссии

Д.В. Паринов