

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



Рабочая программа дисциплины (модуля),  
как компонент образовательной программы  
специализированного высшего образования  
по направлению подготовки  
11.04.02 Инфокоммуникационные технологии и  
системы связи,  
утвержденной первым проректором РУТ (МИИТ)  
Тимониным В.С.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)**

**Java-разработка. Виртуальные машины**

Направление подготовки: 11.04.02                      Инфокоммуникационные  
технологии и системы связи

Направленность (профиль): Инфокоммуникационные и нейросетевые  
технологии передачи и анализа больших  
данных

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: 167783  
Подписал: руководитель образовательной программы  
Киселёва Анастасия Сергеевна  
Дата: 11.06.2026

## 1. Общие сведения о дисциплине (модуле).

Целью освоения учебной дисциплины является формирование у обучающихся компетенций в соответствии с требованиями образовательного стандарта и в освоении принципов разработки программного обеспечения на языке Java с использованием виртуальных машин для создания кроссплатформенных приложений и эффективного управления ресурсами.

Задачи дисциплины включают углубленное изучение продвинутых возможностей языка Java, таких как функциональное программирование и работа с потоками данных. Студенты должны освоить архитектуру виртуальной машины Java, включая механизмы управления памятью и оптимизации производительности, а также научиться применять паттерны проектирования для создания масштабируемых и поддерживаемых приложений. Важной задачей также является изучение интеграции с современными фреймворками и инструментами разработки, такими как Spring и Maven, для повышения эффективности работы в команде и улучшения качества кода.

## 2. Планируемые результаты обучения по дисциплине (модулю).

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

**ПК-2** - Способен осуществлять разработку и внедрение специального программного обеспечения цифровой обработки сигналов, цифрового программного управления на языках высокого и низкого уровней.

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

### **Знать:**

- особенности языка программирования Java, включая его синтаксис, особенности и новые возможности, а также архитектуру виртуальной машины Java (JVM) и ее влияние на производительность приложений;
- методики сбора, анализа и обработки статистической информации инфокоммуникационных систем;
- правила и закономерности личной и деловой устной и письменной коммуникации.

### **Уметь:**

- разрабатывать и оптимизировать многопоточные приложения, эффективно управлять памятью и ресурсами, а также применять различные

паттерны проектирования для создания масштабируемых и поддерживаемых решений;

- проводить исследования характеристик телекоммуникационного оборудования и оценки качества предоставляемых услуг;

- применять на практике коммуникативные технологии, методы и способы делового общения для академического и профессионального взаимодействия.

#### **Владеть:**

- навыками с работы современными фреймворками, такими как Spring и Hibernate, интеграции их в свои проекты, а также использовать инструменты сборки и управления зависимостями, такие как Maven или Gradle, для автоматизации процессов разработки;

- навыками проведения экспериментальных работ по проверке достижимости технических характеристик, радиоэлектронной аппаратуры;

- методикой межличностного делового общения на русском и иностранном языках, с применением профессиональных языковых форм, средствами современных коммуникативных технологий.

### 3. Объем дисциплины (модуля).

#### 3.1. Общая трудоемкость дисциплины (модуля).

Общая трудоемкость дисциплины (модуля) составляет 6 з.е. (216 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 №2
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 168 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	<b>Расширенные возможности Java.</b> Рассматриваемые вопросы: - новые функции языка (Java 8 и выше); - параметризованные типы и дженерики; - обработка исключений: best practices.
2	<b>Архитектура и производительность JVM.</b> Рассматриваемые вопросы: - внутренние механизмы JVM (классовая загрузка, выполнение); - оптимизация производительности (JIT-компиляция, HotSpot); - сборка мусора: алгоритмы и настройки.
3	<b>Многопоточность и асинхронное программирование.</b> Рассматриваемые вопросы: - продвинутые концепции многопоточности (ThreadLocal, Fork/Join); - параллельные потоки и Stream API; - асинхронное программирование с использованием CompletableFuture.
4	<b>Сетевое программирование и протоколы.</b> Рассматриваемые вопросы: - основы работы с сокетами в Java; - создание RESTful API с использованием Spring Boot; - введение в WebSocket и его применение.
5	<b>Работа с базами данных и ORM.</b> Рассматриваемые вопросы: - глубокое понимание Hibernate: кэширование, транзакции; - оптимизация запросов и работа с SQL; - миграции баз данных с Flyway или Liquibase.
6	<b>Разработка микросервисов.</b> Рассматриваемые вопросы: - архитектура микросервисов: принципы и паттерны; - сервисная коммуникация: REST vs. gRPC; - управление конфигурацией и секретами (Spring Cloud Config, HashiCorp Vault).

№ п/п	Тематика лекционных занятий / краткое содержание
7	Тестирование и обеспечение качества кода. Рассматриваемые вопросы: - юнит-тестирование с JUnit и Mockito; - интеграционное тестирование и тестирование производительности; - практики тестирования в Agile-среде.
8	DevOps и CI/CD в Java-разработке. Рассматриваемые вопросы: - основы DevOps: философия и инструменты; - настройка пайплайнов CI/CD с Jenkins или GitLab CI; - контейнеризация приложений с Docker.

## 4.2. Занятия семинарского типа.

### Практические занятия

№ п/п	Тематика практических занятий/краткое содержание
1	Работа с новыми возможностями Java. Рассматриваемые вопросы: - лямбда-выражения и функциональные интерфейсы; - streams API: фильтрация, сортировка и агрегация данных; - параметризованные типы и дженерики.
2	Оптимизация производительности JVM. Рассматриваемые вопросы: - профилирование приложений с JVisualVM; - настройка параметров JVM для повышения производительности; - алгоритмы сборки мусора: настройка и тестирование.
3	Многопоточность и параллелизм. Рассматриваемые вопросы: - реализация многопоточных приложений с использованием ExecutorService; - использование Concurrent Collections; - создание и управление потоками с помощью Fork/Join Framework.
4	Разработка RESTful API. Рассматриваемые вопросы: - создание RESTful сервисов с Spring Boot; - обработка ошибок и валидация входящих данных; - документирование API с использованием Swagger.
5	Работа с базами данных. Рассматриваемые вопросы: - использование Hibernate для ORM; - оптимизация запросов и индексация; - реализация миграций с Flyway или Liquibase.
6	Тестирование кода. Рассматриваемые вопросы: - написание юнит-тестов с JUnit и Mockito; - интеграционное тестирование с использованием Spring Test; - настройка тестирования производительности с JMeter.
7	Архитектура микросервисов. Рассматриваемые вопросы: - реализация микросервисной архитектуры с Spring Cloud;

№ п/п	Тематика практических занятий/краткое содержание
	- использование API Gateway и Service Discovery; - настройка конфигурации и управления секретами.
8	<b>Контейнеризация приложений.</b> Рассматриваемые вопросы: - создание Docker-контейнеров для Java-приложений; - настройка Docker Compose для многоконтейнерных приложений; - развертывание приложений в Kubernetes.
9	<b>CI/CD.</b> Рассматриваемые вопросы: - настройка пайплайнов CI/CD с Jenkins; - автоматизация развертывания с использованием Ansible; - интеграция тестов в процесс CI/CD.
10	<b>Безопасность приложений.</b> Рассматриваемые вопросы: - основы безопасности в веб-приложениях (OWASP Top Ten); - реализация аутентификации и авторизации с Spring Security; - защита от уязвимостей: XSS, CSRF, SQL-инъекции.
11	<b>Работа с кэшированием.</b> Рассматриваемые вопросы: - использование кэша в приложениях с Spring Cache; - реализация распределенного кэша с Redis; - настройка кэширования запросов к базе данных.
12	<b>Логирование и мониторинг.</b> Рассматриваемые вопросы: - настройка логирования с использованием SLF4J и Logback; - мониторинг приложений с Prometheus и Grafana; - анализ логов с ELK Stack (Elasticsearch, Logstash, Kibana).
13	<b>Проектирование API.</b> Рассматриваемые вопросы: - создание и тестирование GraphQL API; - использование gRPC для высокопроизводительных сервисов; - реализация версионирования API.
14	<b>Работа с асинхронным программированием</b> Рассматриваемые вопросы: - использование CompletableFuture для асинхронных задач; - реализация обработки событий с использованием Reactive Streams; - применение Project Reactor для создания реактивных приложений.

#### 4.3. Самостоятельная работа обучающихся.

№ п/п	Вид самостоятельной работы
1	Самостоятельное изучение и конспектирование отдельных тем учебной литературы, связанных с разделами дисциплины.
2	Работа с лекционным материалом.
3	Подготовка к практическим занятиям.
4	Выполнение курсового проекта.
5	Подготовка к промежуточной аттестации.

#### 4.4. Примерный перечень тем курсовых проектов

Курсовой проект на тему: "Система управления IoT-устройствами".  
Исходные данные выбираются согласно вариантам:

##### 1. Умные лампы

Количество устройств: 100 умных ламп

Данные об их состоянии: 70 включено, 30 выключено

История событий: 2000 событий (включение/выключение) за месяц

Данные о пользователях: 50 зарегистрированных пользователей

Настройки устройства: 60 ламп с индивидуальными настройками яркости

Данные сенсоров: 0 сенсоров

Геолокация: 20 ламп с активной геолокацией

Триггеры событий: 10 триггеров для автоматизации (например, включение по расписанию)

Взаимодействие с пользователем: 500 взаимодействий за месяц

##### 2. Умные термостаты

Количество устройств: 75 термостатов

Данные об их состоянии: 60 активных, 15 в режиме ожидания

История событий: 1500 событий (изменение температуры) за месяц

Данные о пользователях: 40 пользователей

Настройки устройства: 70 термостатов с индивидуальными настройками

Данные сенсоров: 75 сенсоров температуры

Геолокация: 50 термостатов с активной геолокацией

Триггеры событий: 15 триггеров для автоматизации (например, по расписанию)

Взаимодействие с пользователем: 600 взаимодействий за месяц

##### 3. Умные камеры

Количество устройств: 50 камер

Данные об их состоянии: 45 включено, 5 выключено

История событий: 3000 событий (запись видео) за месяц

Данные о пользователях: 30 пользователей

Настройки устройства: 20 камер с индивидуальными настройками

Данные сенсоров: 50 сенсоров движения

Геолокация: 30 камер с активной геолокацией

Триггеры событий: 5 триггеров (например, запись при обнаружении движения)

Взаимодействие с пользователем: 800 взаимодействий за месяц

#### 4. Умные замки

Количество устройств: 40 замков

Данные об их состоянии: 35 открыто, 5 закрыто

История событий: 1000 событий (открытие/закрытие) за месяц

Данные о пользователях: 20 пользователей

Настройки устройства: 30 замков с индивидуальными настройками

Данные сенсоров: 0 сенсоров

Геолокация: 15 замков с активной геолокацией

Триггеры событий: 8 триггеров (например, открытие по приближению)

Взаимодействие с пользователем: 300 взаимодействий за месяц

#### 5. Умные датчики дыма

Количество устройств: 60 датчиков

Данные об их состоянии: 50 активных, 10 в режиме ожидания

История событий: 500 событий (срабатывание) за месяц

Данные о пользователях: 25 пользователей

Настройки устройства: 55 датчиков с индивидуальными настройками

Данные сенсоров: 60 сенсоров дыма

Геолокация: 10 датчиков с активной геолокацией

Триггеры событий: 3 триггера (например, сигнализация при срабатывании)

Взаимодействие с пользователем: 200 взаимодействий за месяц

#### 6. Умные фитнес-трекеры

Количество устройств: 120 фитнес-трекеров

Данные об их состоянии: 100 активных, 20 в режиме ожидания

История событий: 2500 событий (измерение активности) за месяц

Данные о пользователях: 80 пользователей  
Настройки устройства: 90 трекеров с индивидуальными настройками  
Данные сенсоров: 120 сенсоров (пульс, шаги, сожженные калории)  
Геолокация: 100 трекеров с активной геолокацией  
Триггеры событий: 10 триггеров (например, уведомление о достижении цели)

Взаимодействие с пользователем: 1200 взаимодействий за месяц

#### 7. Умные холодильники

Количество устройств: 30 холодильников

Данные об их состоянии: 25 активных, 5 в режиме ожидания

История событий: 800 событий (изменение температуры) за месяц

Данные о пользователях: 15 пользователей

Настройки устройства: 20 холодильников с индивидуальными настройками

Данные сенсоров: 30 сенсоров температуры и влажности

Геолокация: 5 холодильников с активной геолокацией

Триггеры событий: 6 триггеров (например, уведомление о высоких температурах)

Взаимодействие с пользователем: 400 взаимодействий за месяц

#### 8. Умные системы полива

Количество устройств: 50 систем полива

Данные об их состоянии: 40 активных, 10 в режиме ожидания

История событий: 1200 событий (включение/выключение полива) за месяц

Данные о пользователях: 25 пользователей

Настройки устройства: 35 систем с индивидуальными настройками

Данные сенсоров: 50 сенсоров влажности почвы

Геолокация: 30 систем с активной геолокацией

Триггеры событий: 12 триггеров (например, автоматический полив по расписанию)

Взаимодействие с пользователем: 500 взаимодействий за месяц

#### 9. Умные колонки

Количество устройств: 90 колонок

Данные об их состоянии: 70 активных, 20 в режиме ожидания

История событий: 3000 событий (воспроизведение музыки, запросы) за месяц

Данные о пользователях: 60 пользователей

Настройки устройства: 80 колонок с индивидуальными настройками звука

Данные сенсоров: 0 сенсоров

Геолокация: 10 колонок с активной геолокацией

Триггеры событий: 15 триггеров (например, воспроизведение музыки по голосовой команде)

Взаимодействие с пользователем: 2000 взаимодействий за месяц

10. Умные системы безопасности

Количество устройств: 45 систем безопасности

Данные об их состоянии: 35 активных, 10 в режиме ожидания

История событий: 900 событий (срабатывание сигнализации, уведомления) за месяц

Данные о пользователях: 25 пользователей

Настройки устройства: 30 систем с индивидуальными настройками

Данные сенсоров: 45 сенсоров (движения, открытия дверей)

Геолокация: 20 систем с активной геолокацией

Триггеры событий: 7 триггеров (например, уведомления при срабатывании)

Взаимодействие с пользователем: 600 взаимодействий за месяц

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

№ п/п	Библиографическое описание	Место доступа
1	Строганкова, Н. В. Шаблоны программных платформ языка Java : учебное пособие / Н. В. Строганкова, К. В. Касьяненко, А. В. Хозяинов ; составители практикума входят: использование. — Москва : РТУ МИРЭА, 2021. — 83 с.	<a href="https://e.lanbook.com/book/182466">https://e.lanbook.com/book/182466</a>

2	Пономарчук, Ю. В. Программирование на языке Java : учебное пособие / Ю. В. Пономарчук, И. В. Кузнецов. — Хабаровск : ДВГУПС, 2021. — 103 с.	<a href="https://e.lanbook.com/book/259451">https://e.lanbook.com/book/259451</a>
3	Федоричев, Л. А. Реализация многопоточности в языке Java : учебное пособие для вузов / Л. А. Федоричев, О. В. Букунова. — 2-е изд., стер. — Санкт-Петербург : Лань, 2025. — 72 с. — ISBN 978-5-507-52722-9.	<a href="https://e.lanbook.com/book/457502">https://e.lanbook.com/book/457502</a>

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

Информационный портал Научная электронная библиотека eLIBRARY.RU ([www.elibrary.ru](http://www.elibrary.ru));

Единая коллекция цифровых образовательных ресурсов (<http://window.edu.ru>);

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

Поисковые системы «Яндекс» для доступа к тематическим информационным ресурсам;

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

Электронно-библиотечная система [ibooks.ru](http://ibooks.ru) — <http://ibooks.ru/>;

Электронно-библиотечная система «УМЦ» — <http://www.umczdt.ru/>;

Электронно-библиотечная система «BOOK.ru» — <http://www.book.ru/>;

Электронно-библиотечная система «ZNANIUM.COM» — <http://www.znanium.com/>

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

1. Операционная система windows microsoft office 2003 и выше;

2. Браузер Internet Explorer 8.0 и выше с установленным Adobe Flash player версии 10.3 и выше;

3. Adobe acrobat.

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.

9. Форма промежуточной аттестации:

Курсовой проект во 2 семестре.

Экзамен во 2 семестре.

10. Оценочные материалы.

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

Авторы:

старший преподаватель кафедры  
«Системы управления транспортной  
инфраструктурой»

А.Н. Малых

Согласовано:

Директор

Д.В. Паринов

Руководитель образовательной  
программы

А.С. Киселёва

Председатель учебно-методической  
комиссии

Д.В. Паринов