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 директором ФГАОУ ВО РУТ
(МИИТ) Париновым Д.В.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Разработка серверной части веб-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 ФГАОУ ВО РУТ (МИИТ)
ID подписи: 937226
Подписал: директор центра Проневич Ольга Борисовна
Дата: 12.11.2026

1. Общие сведения о дисциплине (модуле).

Дисциплина посвящена разработке серверной части веб-приложений на Java и Spring MVC для корпоративных и транспортных цифровых систем. В ходе изучения рассматриваются основы HTTP-взаимодействия, cookie, серверные сессии, Redis, Spring IoC, контроллеры Spring MVC, шаблоны Thymeleaf, формы, валидация, обработка ошибок, доступ к данным через Spring Data JPA, кэширование и разграничение доступа. Итоговый образовательный результат связан с созданием серверного веб-приложения с каталогом предметных объектов, пользовательскими сценариями, хранением данных, ролевой моделью и технической документацией.

Целью освоения дисциплины является формирование способности проектировать, реализовывать, тестировать и сопровождать серверную часть веб-приложений на платформе Java Spring MVC с учетом требований к корректной обработке HTTP-взаимодействия, сохранению пользовательского состояния, целостности данных, безопасности и сопровождаемости программного решения.

Для достижения поставленной цели в рамках дисциплины решается комплекс задач, направленных на формирование у обучающихся способности – анализировать пользовательские сценарии веб-приложения и переводить их в структуру страниц, переходов и серверных обработчиков, реализовывать HTTP-взаимодействие от простого сокетного сервера до контроллеров Spring MVC, управлять состоянием пользователя при помощи cookie, серверных сессий и Redis, проектировать многослойную серверную архитектуру на основе Spring IoC, контроллеров, служебного слоя, шаблонов Thymeleaf и слоя доступа к данным, реализовывать валидацию, обработку ошибок, поиск, пагинацию, кэширование и разграничение доступа, проводить многоуровневую проверку веб-сценариев и готовить техническую документацию по серверному веб-приложению.

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;

ППК-Р3 - Способен выполнять интеграцию программных модулей и компонентов и проверять работоспособность выпусков программного продукта;

ППК-Р4 - Способен использовать базы данных при создании программных модулей и компонентов.

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

Знать:

- принципы клиент-серверного взаимодействия в веб-приложениях, включая HTTP-запрос, HTTP-ответ, заголовки, cookie и перенаправления
- назначение cookie, серверных сессий, идентификатора сессии, времени жизни cookie и серверного удаления состояния пользователя
- способы хранения сессионных данных в памяти приложения и во внешнем хранилище Redis
- принципы прототипирования веб-приложения на уровне пользовательских историй, структуры страниц и переходов между страницами
- модель Spring Framework, включая контейнер инверсии управления, внедрение зависимостей, жизненный цикл компонентов и области видимости singleton, request и session
- архитектуру Spring MVC, включая DispatcherServlet, контроллеры, модель, представление, маршрутизацию запросов и связывание параметров
- механизмы Spring MVC для PathVariable, RequestParam, ModelAttribute, RedirectAttributes и шаблона перенаправления после отправки формы
- назначение Thymeleaf как серверного шаблонизатора и основные средства вывода данных, условного отображения, циклов, ссылок и фрагментов
- слоистую структуру веб-приложения, включая контроллеры, служебный слой, модели представления, предметные сущности и слой доступа к данным
- принципы реализации операций создания, чтения, изменения, удаления и поиска объектов предметной области
- основы валидации входных данных, BindingResult, пользовательских валидаторов и отображения ошибок формы
- принципы пагинации, сортировки, фильтрации и работы с зависимыми справочниками в веб-интерфейсе
- механизмы централизованной обработки ошибок через ControllerAdvice и правила подготовки понятных страниц ошибок
- основы работы с реляционной БД через Spring Data JPA, JpaRepository, JPQL и транзакционные операции служебного слоя

- назначение ModelMapper или аналогичных преобразователей моделей при отделении данных формы от предметных сущностей

- принципы кэширования прикладных данных в Redis с использованием Cacheable и CacheEvict

- основы Spring Security, включая аутентификацию, авторизацию, BCrypt и разграничение доступа по ролям USER, MODERATOR и ADMIN

- возможности Thymeleaf Security для отображения элементов интерфейса с учетом прав пользователя

- методы проверки серверного веб-приложения, включая проверку HTTP-сценариев, контроллеров, форм, сессий, БД, кэша и правил безопасности

- требования к технической документации серверного веб-приложения, включая описание сценариев, архитектуры, схемы данных, ролей доступа и результатов проверки

Уметь:

- уметь реализовывать простой HTTP-сервер при помощи Java sockets в условиях ручной обработки запроса, ответа, заголовков и cookie

- уметь управлять состоянием посещений при помощи cookie и Redis в условиях хранения в cookie только идентификатора сессии

- уметь проектировать пользовательские истории и структуру страниц при помощи диаграмм переходов в условиях подготовки серверного веб-приложения

- уметь настраивать компоненты Spring при помощи IoC-контейнера, областей видимости singleton, request и session в условиях различения общего и пользовательского состояния

- уметь реализовывать контроллеры Spring MVC при помощи PathVariable, RequestParam, RedirectAttributes и шаблона перенаправления после отправки формы в условиях каталога объектов предметной области

- уметь создавать серверные представления при помощи Thymeleaf в условиях вывода списков, условий, ссылок, фрагментов и визуального выделения значимых данных

- уметь реализовывать формы редактирования при помощи ModelAttribute, Jakarta Bean Validation и BindingResult в условиях отображения ошибок пользователю

- уметь реализовывать поиск, пагинацию, удаление и выбор связанных сущностей при помощи Spring MVC и Spring Data JPA в условиях работы с каталогом предметной области

- уметь проектировать служебный слой при помощи Service и Transactional в условиях отделения прикладных правил от контроллеров и репозиториев

- уметь реализовывать слой доступа к данным при помощи JpaRepository и JPQL в условиях выборки, фильтрации и изменения объектов БД

- уметь применять преобразование моделей при помощи ModelMapper или собственного преобразователя в условиях отделения форм и представлений от сущностей хранения

- уметь реализовывать кэширование справочных данных при помощи Redis Cache в условиях обновления и сброса устаревших значений

- уметь настраивать Spring Security, BCrypt и разграничение доступа по ролям в условиях защиты административных и модераторских сценариев

- уметь проверять веб-приложение при помощи JUnit, Spring MVC Test и ручных HTTP-сценариев в условиях контроля корректности форм, сессий, ошибок и ролей доступа

- уметь готовить техническую документацию серверного веб-приложения в условиях представления пользовательских сценариев, структуры страниц, архитектуры, схемы данных и результатов проверки

Владеть:

- навыком разработки серверного веб-приложения на Spring MVC с контроллерами, моделями, представлениями и формами

- навыком реализации cookie, серверных сессий и внешнего сессионного хранения в Redis

- навыком применения областей видимости компонентов Spring для общего и пользовательского состояния

- навыком создания страниц Thymeleaf с фрагментами, условиями, циклами и отображением ошибок формы

- навыком реализации каталога объектов с операциями создания, чтения, изменения, удаления, поиска, пагинации и фильтрации

- навыком подключения Spring Data JPA, транзакционного служебного слоя, JPQL-запросов и преобразования моделей

- навыком настройки кэширования Redis и разграничения доступа через Spring Security

- навыком подготовки технической документации и демонстрации серверного веб-приложения

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):	80	80
В том числе:		
Занятия лекционного типа	32	32
Занятия семинарского типа	48	48

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 100 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	Серверная часть веб-приложения и HTTP-взаимодействие Рассматриваемые вопросы: - роль серверной части в обработке пользовательских действий веб-приложения; - структура HTTP-запроса, HTTP-ответа, заголовков и кодов состояния; - связь пользовательского сценария, маршрута, обработчика и ответа сервера.
2	Cookie и управление состоянием пользователя Рассматриваемые вопросы: - назначение cookie и правила передачи cookie между клиентом и сервером;

№ п/п	Тематика лекционных занятий / краткое содержание
	- время жизни cookie, удаление cookie и ограничения клиентского хранения; - счетчики посещений и типовые ошибки хранения пользовательского состояния.
3	Серверные сессии и Redis как внешнее хранилище состояния Рассматриваемые вопросы: - идентификатор сессии и отделение ключа сессии от данных пользователя; - хранение сессионных данных в Redis и правила именования ключей; - удаление сессии и завершение пользовательского сценария.
4	Пользовательские сценарии и структура веб-интерфейса Рассматриваемые вопросы: - пользовательские истории как основа проектирования веб-сценариев; - структура страниц, переходы, формы и состояния интерфейса; - согласование сценариев, маршрутов и серверных обработчиков.
5	Spring Framework, компоненты и области видимости Рассматриваемые вопросы: - контейнер инверсии управления и внедрение зависимостей; - области видимости singleton, request и session; - жизненный цикл компонента, PostConstruct и PreDestroy.
6	Архитектура Spring MVC Рассматриваемые вопросы: - dispatcherServlet, контроллер, модель и представление; - маршрутизация запроса к обработчику и формирование ответа; - границы ответственности контроллера в серверном веб-приложении.
7	Параметры запросов и перенаправления в Spring MVC Рассматриваемые вопросы: - pathVariable и requestParam в обработке адресов и параметров запроса; - redirectAttributes и шаблон перенаправления после отправки формы; - реализация операций каталога через контроллеры.
8	Thymeleaf как серверный шаблонизатор Рассматриваемые вопросы: - вывод значений, циклы, условия и построение ссылок в шаблонах; - фрагменты страниц и повторное использование элементов интерфейса; - отображение предметных данных и визуальное выделение значимых состояний.
9	Формы, связывание моделей и валидация Рассматриваемые вопросы: - modelAttribute и связывание данных формы с объектом модели; - jakarta bean validation и BindingResult; - отображение ошибок ввода на странице формы.
10	Поиск, пагинация и зависимые справочники Рассматриваемые вопросы: - поиск по предметным признакам и фильтрация списка объектов; - пагинация, сортировка и удаление записей; - заполнение выпадающих списков зависимыми данными.
11	Централизованная обработка ошибок Рассматриваемые вопросы: - исключения контроллеров и служебного слоя; - controllerAdvice и единая логика подготовки страницы ошибки; - различение пользовательских, доменных и инфраструктурных ошибок.
12	Служебный слой и транзакционные операции Рассматриваемые вопросы: - назначение служебного слоя и отделение прикладных правил от контроллеров;

№ п/п	Тематика лекционных занятий / краткое содержание
	- transactional и границы изменения данных; - согласование сценариев каталога с операциями БД.
13	Доступ к данным через Spring Data JPA Рассматриваемые вопросы: - jpa-сущности, репозитории и методы JpaRepository; - jpql-запросы для поиска и фильтрации; - преобразование сущностей в модели представления.
14	Кэширование данных в Redis Рассматриваемые вопросы: - назначение кэширования справочных и часто запрашиваемых данных; - cacheable и cacheEvict в Spring-приложении; - согласованность кэша при изменении данных каталога.
15	Аутентификация и разграничение доступа Рассматриваемые вопросы: - spring security, роли пользователя и защита маршрутов; - хранение паролей с использованием BCrypt; - отображение элементов интерфейса с учетом прав пользователя.
16	Проверка качества и документирование серверного веб-приложения Рассматриваемые вопросы: - проверка контроллеров, форм, сессий, БД, кэша и правил безопасности; - критерии корректности пользовательских сценариев и серверных обработчиков; - структура технической документации и порядок представления результата.

4.2. Занятия семинарского типа.

Лабораторные работы

№ п/п	Наименование лабораторных работ / краткое содержание
1	Реализация HTTP-сервера на сокетах и счетчика посещений через cookie Студент реализует простой HTTP-сервер на Java sockets и формирует HTML-ответ с русскоязычным заголовком. Счетчик посещений хранится в cookie, изменяется с заданным шагом и ограничивается верхним значением. Добавляется маршрут уменьшения счетчика и время жизни cookie.
2	Реализация сессий на Redis с хранением идентификатора в cookie Студент переносит данные счетчика из cookie во внешнее хранилище Redis. Cookie содержит только идентификатор сессии, а значения сохраняются с заданным префиксом ключей. Добавляется маршрут завершения работы, который удаляет сессию и cookie.
3	Прототипирование серверного веб-приложения Студент готовит набор пользовательских историй и строит прототип страниц. В прототип включаются главная страница, каталог, карточка объекта, страница выбранных элементов или заявки и авторизация. Между страницами фиксируются переходы и основные действия пользователя.
4	Реализация компонентов Spring с различными областями видимости Студент создает компоненты с областями видимости singleton, request и session. Реализуются общий счетчик с синхронизацией и отдельный счетчик пользовательской сессии. В компоненты добавляются методы жизненного цикла PostConstruct и PreDestroy.
5	Реализация контроллеров Spring MVC для каталога Студент реализует контроллеры каталога с использованием PathVariable, RequestParam, RedirectAttributes и перенаправления после отправки формы. В каталоге создаются операции

№ п/п	Наименование лабораторных работ / краткое содержание
	добавления, просмотра, изменения, удаления и поиска по предметному признаку. Результаты операций отображаются пользователю через сообщения после перенаправления.
6	Создание шаблонов Thymeleaf для каталога Студент создает страницы каталога на Thymeleaf с выводом текста, циклов, условий и ссылок. Для объектов с высоким значением рейтинга или приоритета добавляется визуальное выделение и текстовый признак. Общая шапка страницы выносится во фрагмент.
7	Реализация форм Spring MVC, связывания данных и валидации Студент реализует формы создания и редактирования объекта каталога через ModelAttribute. Для полей задаются ограничения Jakarta Bean Validation и обработка BindingResult. Ошибки ввода отображаются рядом с соответствующими полями формы.
8	Реализация пагинации, поиска, удаления и связанных данных Студент добавляет постраничный вывод каталога, поиск и удаление записей. Для связанной сущности создается выпадающий список, заполняемый данными из приложения. Результаты поиска и переходы по страницам сохраняют выбранные параметры.
9	Реализация централизованной обработки ошибок в Spring MVC Студент создает обработчик ошибок через ControllerAdvice. Для отсутствующего объекта, ошибки валидации и внутренней ошибки приложения готовятся разные варианты ответа. Пользователь получает страницу ошибки с понятным описанием ситуации.
10	Реализация аутентификации пользователя в веб-приложении Студент добавляет страницу входа и базовую модель пользователя. После успешного входа сведения о пользователе применяются в пользовательском сценарии. Неавторизованный пользователь перенаправляется на страницу входа при обращении к защищенной области.
11	Реализация служебного слоя и транзакционных операций Студент переносит прикладные правила каталога в служебные компоненты. Для операций изменения данных задаются транзакционные границы. Контроллеры обращаются к служебному слою без прямого выполнения прикладных правил.
12	Реализация доступа к данным через JpaRepository и JPQL Студент подключает Spring Data JPA и реализует репозитории для основных сущностей приложения. Для поиска создается JPQL-запрос или метод репозитория с предметным условием. Формы и представления отделяются от сущностей хранения через преобразователь моделей.
13	Реализация кэширования справочных данных в Redis Студент подключает Redis Cache для часто запрашиваемых данных каталога или справочника. Для чтения применяется кэширование, а при изменении данных выполняется сброс устаревших значений. Поведение кэша проверяется повторными запросами и журналом приложения.
14	Настройка разграничения доступа через Spring Security Студент настраивает роли USER, MODERATOR и ADMIN для страниц и действий веб-приложения. Пароли сохраняются с использованием BCrypt. Для каждой роли проверяется доступ к просмотру, изменению, удалению и административным страницам.
15	Отображение интерфейса с учетом прав пользователя Студент подключает Thymeleaf Security и изменяет отображение меню, кнопок и сообщений в зависимости от роли пользователя. Для формы добавляется пользовательский валидатор, учитывающий предметные ограничения. Интерфейс проверяется под учетными записями разных ролей.
16	Комплексная проверка серверного веб-приложения Студент проверяет полный пользовательский маршрут, работу сессий, каталог, формы, БД, кэш и правила безопасности. Исправляются несогласованности между прототипом, реализацией и технической документацией. Итоговое представление результата включает запуск приложения, выполнение ключевого сценария и предъявление результатов проверки.

Практические занятия

№ п/п	Тематика практических занятий/краткое содержание
1	Анализ HTTP-сценариев и состояния пользователя Студент описывает пользовательский сценарий посещения веб-приложения и определяет, какие данные передаются в запросах, ответах и cookie. Для счетчика посещений задаются правила изменения, ограничения значения и завершения сценария. Подготовленное описание используется как задание для реализации на сокетах и Redis.
2	Проектирование структуры страниц и переходов Студент формирует не менее пяти пользовательских историй и определяет состав страниц веб-приложения. Для главной страницы, каталога, карточки объекта, корзины или заявки и авторизации строится схема переходов. Итоговый прототип задает структуру будущих контроллеров и шаблонов.
3	Проектирование компонентов Spring и контроллеров MVC Студент распределяет общее, запросное и сессионное состояние между компонентами Spring. Для каталога предметных объектов определяются маршруты, параметры, перенаправления и сообщения после выполнения операций. Решение становится техническим заданием для реализации счетчиков и контроллеров каталога.
4	Проектирование представлений Thymeleaf и форм Студент определяет состав шаблонов, фрагментов, условий отображения и списков данных. Для форм создания и редактирования задаются поля, правила валидации и места вывода ошибок. Проект представлений используется при реализации страниц каталога.
5	Проектирование поиска, пагинации и обработки ошибок Студент задает правила поиска, пагинации, сортировки, удаления и выбора связанных сущностей. Для ошибочных сценариев определяется, какие сообщения получает пользователь и какие страницы отображаются. Подготовленное решение используется при реализации расширенных возможностей Spring MVC.
6	Проектирование служебного слоя и доступа к данным Студент выделяет прикладные операции, транзакционные границы, сущности БД и запросы к репозиториям. Для моделей формы и сущностей хранения задаются правила преобразования. Итоговая схема используется при реализации служебного слоя, JPA и JPQL.
7	Проектирование кэширования и разграничения доступа Студент определяет данные, пригодные для кэширования, условия обновления кэша и правила сброса устаревших значений. Затем задаются роли USER, MODERATOR и ADMIN, защищаемые страницы и ограничения действий. Результат используется при настройке Redis Cache и Spring Security.
8	Проектирование проверки серверного веб-приложения Студент составляет перечень проверок для основных страниц, форм, сессий, ролей, БД и кэша. Для демонстрации выбирается полный пользовательский маршрут от входа до выполнения защищенной операции. Итогом становится план технического отчета и представления веб-приложения.

4.3. Самостоятельная работа обучающихся.

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

4.4. Примерный перечень тем курсовых проектов

Система управления грузовыми перевозками и отслеживания отправок

Корпоративный портал учета отпусков и согласования заявлений сотрудников

Веб-приложение для учета автопарка и планирования технического обслуживания

Система инвентаризации и выдачи корпоративного IT-оборудования

Сервис бронирования билетов и управления расписанием междугородних рейсов

Платформа электронного документооборота и маршрутизации внутренних заявок

Система управления маршрутами доставки и закрепления курьеров

Корпоративный сервис учета заявок в службу технической поддержки (Helpdesk)

Веб-приложение складского учета товаров и оформления отгрузочных накладных

Портал корпоративного обучения и учета прохождения сертификаций

Система бронирования корпоративных парковочных мест и контроля доступа

Веб-сервис подачи, проверки и согласования финансовых авансовых отчетов

Система учета топливных карт и логов заправок корпоративного транспорта

Сервис бронирования переговорных комнат и резервирования оборудования

Платформа регистрации и отслеживания инцидентов на транспортных узлах

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

№ п/п	Библиографическое описание	Место доступа
1	Спилкэ, Л. Spring Security в действии : руководство / Л. Спилкэ ; пер. с англ. В. С. Яценкова. — Москва : ДМК Пресс, 2025. — 480 с. — ISBN 978-5-93700-256-3. — Текст : электронный	Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/514863 (дата обращения: 19.06.2026)

2	Васюткина, И. А. Разработка серверной части web-приложений на Java : учебное пособие / И. А. Васюткина. — Новосибирск : НГТУ, 2021. — 83 с. — ISBN 978-5-7782-4394-1. — Текст : электронный	Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/216155 (дата обращения: 19.06.2026)
3	Бабаев, С. И. Технологии, стандарты и протоколы вычислительных сетей. Технологии вычислительных сетей : учебное пособие / С. И. Бабаев, Б. В. Костров, М. Б. Никифоров. — Рязань : РГРТУ, 2024 — Часть 1 — 2024. — 164 с. — ISBN 978-5-907352-53-7. — Текст : электронный	Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/439760 (дата обращения: 19.06.2026)
4	Рогов, Е. PostgreSQL 16 изнутри : руководство / Е. Рогов ; главный редактор Д. А. Мовчан. — Москва : ДМК Пресс, 2024. — 664 с. — ISBN 978-5-93700-305-8. — Текст : электронный	Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/456800 (дата обращения: 19.06.2026)

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

ЭБС Лань – https://e.lanbook.com/.

Образовательная платформа Юрайт – https://urait.ru/.

Единый реестр российских программ для ЭВМ и баз данных – https://reestr.digital.gov.ru/reestr/.

Профессиональные стандарты и квалификации, справочная информация КонсультантПлюс – https://www.consultant.ru/document/cons_doc_LAW_157436/.

Транспортная стратегия Российской Федерации до 2030 года с прогнозом на период до 2035 года, КонсультантПлюс – https://www.consultant.ru/document/cons_doc_LAW_402052/.

Документация Java SE 21 – https://docs.oracle.com/en/java/javase/21/docs/api/index.html.

Документация Spring Framework – https://docs.spring.io/spring-framework/reference/.

Документация Spring MVC – https://docs.spring.io/spring-framework/reference/web/webmvc.html.

Документация Spring Boot – https://docs.spring.io/spring-boot/index.html.

Документация Spring Security – https://docs.spring.io/spring-security/reference/index.html.

Документация Thymeleaf – https://www.thymeleaf.org/documentation.html.

Документация Redis – https://redis.io/docs/latest/.

Документация PostgreSQL – https://www.postgresql.org/docs/current/.
Документация JUnit – https://docs.junit.org/current/user-guide/.

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

Операционные системы – Astra Linux, ALT Linux, РЕД ОС, Debian GNU/Linux.

Платформа разработки – OpenJDK 21+, Maven или Gradle, IntelliJ IDEA Community Edition.

Веб-разработка – Spring Boot, Spring MVC, Spring Data JPA, Spring Security, Thymeleaf.

Хранение данных – PostgreSQL или Postgres Pro, Redis.

Проверка и сопровождение – JUnit, Spring MVC Test, Git, diagrams.net, curl.

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 и лабораторных занятий – наличие персональных компьютеров вычислительного класса.

9. Форма промежуточной аттестации:

Курсовой проект в 5 семестре.
Экзамен в 5 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»
Института железнодорожного
транспорта Е.А. Заманов

Согласовано:

Заведующий кафедрой ЦТУТП В.Е. Нутович

Директор ЦС IT - программ ВИШ О.Б. Проневич

Председатель учебно-методической
комиссии Н.А. Андриянова