

**УТВЕРЖДЕНА**

Решением Ученого совета  
АНО ВО «Центральный университет»  
«07» марта 2024 г.  
Протокол №1

**Рабочая программа дисциплины (модуля)  
«Разработка backend-сервисов»**

**Направление подготовки:** 02.03.01 Математика и компьютерные науки

**Направленность (профиль) подготовки:** Разработка

**Квалификация (степень) выпускника:** бакалавр

**Форма обучения:** очная

**Срок освоения программы:** 4 года

**Год набора:** 2024

**Москва  
2024**

## Содержание

<b>1. Краткая характеристика дисциплины (модуля)</b> .....	<b>3</b>
<b>2. Перечень планируемых результатов обучения</b> .....	<b>4</b>
<b>3. Тематический план</b> .....	<b>7</b>
<b>4. Содержание дисциплины (модуля)</b> .....	<b>7</b>
<b>5. Учебно-методическое обеспечение</b> .....	<b>8</b>
<b>6. Материально-техническое обеспечение</b> .....	<b>8</b>
<b>7. Методические и оценочные материалы</b> .....	<b>10</b>

## **1. Краткая характеристика дисциплины (модуля)**

Рабочая программа дисциплины (модуля) «Разработка backend-сервисов» составлена в соответствии с федеральным государственным образовательным стандартом высшего образования – бакалавриат по специальности 02.03.01 Математика и компьютерные науки, профиль Разработка, утвержденный приказом Министерства науки и высшего образования Российской Федерации № 807 от 23.08.2017 года.

Изучение дисциплины (модуля) «Разработка backend-сервисов» позволяет создавать надежные и масштабируемые серверные приложения, обеспечивающие эффективное взаимодействие между клиентом и сервером. Это фундаментальный навык для разработки современных веб-приложений и сервисов, поддерживающих бизнес-логику и обработку данных.

### **Место дисциплины (модуля) в структуре образовательной программы**

Настоящая дисциплина (модуль) включена в учебный план по программе подготовки бакалавриата по направлению 02.03.01 Математика и компьютерные науки, профиль Разработка и входит в вариативную часть Блока 1, формируемую участниками образовательных отношений.

Дисциплина (модуль) является выборной и доступна для изучения на 3 или 4 курсе в 5, 6, 7, 8 семестрах на выбор.

**Цель изучения дисциплины (модуля):** формирование навыков проектирования, создания и поддержки серверной части приложений для обеспечения надежного и эффективного обмена данными и бизнес-логики.

**Задачи изучения дисциплины (модуля)** направлены на формирование у студентов следующий знаний, умений и навыков:

- знание фреймворков Spring и технологий вспомогательных проектов;
- умение создавать Web-приложения на микросервисной архитектуре;
- умение встраивать сервисы в существующую инфраструктуру;
- навык решения высокоуровневых задач по разработке.

## 2. Перечень планируемых результатов обучения

Компетенции, формируемые в результате освоения дисциплины (модуля) при проведении учебных занятий в форме контактной работы обучающихся с педагогическими работниками Университета и в форме самостоятельной работы обучающихся:

Компетенция	Содержание компетенции	Индикатор компетенции	Перечень планируемых результатов обучения по дисциплине (модулю)
УК-1.	Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1.1.	Знает методы поиска и анализа информации в области разработки, основные принципы критической оценки источников информации и их релевантности
		УК-1.2.	Умеет критически оценивать источники информации и синтезировать данные из различных источников для решения задач, применять системный подход к анализу и решению комплексных проблем
		УК-1.3.	Имеет практический опыт работы с современными инструментами и технологиями для обработки информации, формулировании и структурировании задач на основе полученной информации
УК-2.	Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений	УК-2.1.	Знает действующие правовые нормы, регулирующие деятельность в области решения задач, основные методы и подходы к определению круга задач
		УК-2.2.	Умеет определять круг задач в рамках поставленной цели, выбирать оптимальные способы решения задач, учитывая имеющиеся ресурсы и ограничения
		УК-2.3.	Имеет практический опыт применения знаний о правовых нормах и ресурсах в реальных ситуациях, разработки и реализации решений в соответствии с установленными ограничениями

ОПК-1.	Способен консультировать и использовать фундаментальные знания в области математического анализа, комплексного и функционального анализа алгебры, аналитической геометрии, дифференциальной геометрии и топологии, дифференциальных уравнений, дискретной математики и математической логики, теории вероятностей, математической статистики и случайных процессов, численных методов, теоретической механики в профессиональной деятельности	ОПК-1.1.	Знает основные концепции и теории в области математического анализа и смежных дисциплин; методы и подходы, используемые в различных областях математики
		ОПК-1.2.	Умеет применять математические методы для решения профессиональных задач
		ОПК-1.3.	Имеет практический опыт разработки и реализация математических моделей в профессиональной деятельности
ПК-1.	Способен формулировать задачи с математической точностью, обосновывать утверждения строго и анализировать полученные результаты в области математики и компьютерных наук	ПК-1.1.	Обладает базовыми знаниями, полученными в области математических наук, программирования и информационных технологий
		ПК-1.2.	Умеет находить, формулировать и решать стандартные задачи в собственной научно-исследовательской деятельности в математике и информатике
		ПК-1.3.	Имеет опыт работы с задачами в области математики и компьютерных наук, включая применение математических методов для решения практических задач
ПК-2.	Способен решать типовые задачи профессиональной деятельности в области разработки, опираясь на информационную и библиографическую культуру, используя информационно-коммуникационные технологии и учитывая основные требования информационной безопасности	ПК-2.1.	Знает основные принципы информационной и библиографической культуры, а также правила и стандарты информационной безопасности
		ПК-2.2.	Умеет эффективно использовать информационно-коммуникационные технологии для решения стандартных задач профессиональной деятельности с учетом требований информационной безопасности

		ПК-2.3.	Имеет практический опыт работы с информационными ресурсами и инструментами в рамках своей профессиональной деятельности в области разработки, соблюдая требования информационной безопасности
--	--	---------	---

### 3. Тематический план

№п/п	Наименование раздела дисциплины (модуля)	Трудоемкость, академические часы				ТКУ (текущий контроль успеваемости)
		<i>Очная форма</i>				
		Контактная работа		Контроль	Самостоятельная работа	
Лекции	Семинары (практические занятия)					
1	Введение в Spring Framework	4	4		20	Подготовка к семинару, Домашние задания
2	Работа с БД	4	4		20	Подготовка к семинару, Домашние задания
3	Разработка Web-приложений	4	4		20	Подготовка к семинару, Домашние задания, Контрольная работа
4	Продвинутые возможности Spring	16	16		70	Подготовка к семинару, Домашние задания, Контрольная работа
	<i>Зачет с оценкой</i>			4		Проект
	<b>Итого:</b>	<b>28</b>	<b>28</b>	<b>4</b>	<b>130</b>	
	<i>Объем дисциплины (модуля) (в ак. ч.)</i>	<b>190</b>				
	<i>Объем дисциплины (модуля) (в зач. ед.)</i>	<b>5</b>				

### 4. Содержание дисциплины (модуля)

№п/п	Наименование раздела дисциплины (модуля)	Содержание дисциплины (модуля) по темам
1	Введение в Spring Framework	Контейнер инверсии управления. Основы Spring Beans.
2	Работа с БД	Подключение к базе данных. ORM и JPA.
3	Разработка Web-приложений	REST API создание. Обработка HTTP-запросов.
4	Продвинутые возможности Spring	Безопасность Spring Security. Кэширование данных. Аспектно-ориентированное программирование. Микросервисы и Spring Boot. Управление транзакциями. Работа с WebSocket. Интеграция с внешними сервисами. Тестирование и отладка приложений.

## 5. Учебно-методическое обеспечение

Университет располагает полным набором лицензионного и свободно распространяемого программного обеспечения, включая продукты отечественного производства.

Каждый студент в течение всего периода обучения получает индивидуальный неограниченный доступ к электронно-библиотечной системе и электронной информационно-образовательной среде университета. Эти системы предоставляют возможность доступа к ресурсам из любой точки, где есть подключение к сети Интернет, как на территории университета, так и за его пределами.

Студентам обеспечен удаленный доступ к современным профессиональным базам данных и информационным справочным системам.

### *Основная литература:*

1. Зараменских, Е. П. Разработка информационных систем : учебник и практикум для вузов / Е. П. Зараменских. — 2-е изд. — Москва : Издательство Юрайт, 2025. — 78 с. — (Высшее образование). — ISBN 978-5-534-21420-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/571333>.

### *Дополнительная литература:*

1. Полуэктова, Н. Р. Разработка веб-приложений : учебник для вузов / Н. Р. Полуэктова. — 2-е изд. — Москва : Издательство Юрайт, 2025. — 204 с. — (Высшее образование). — ISBN 978-5-534-18645-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/567610>.

## 6. Материально-техническое обеспечение

Университет располагает материально-технической базой, соответствующей действующим противопожарным правилам и нормам и обеспечивающей проведение всех видов дисциплинарной и междисциплинарной подготовки, практической и научно-исследовательской работ обучающихся, предусмотренных учебным планом.

Помещения, которые представляют собой учебные аудитории для проведения занятий лекционного типа, занятий семинарского (практического) типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, а также помещения для самостоятельной работы и помещения для хранения и профилактического обслуживания учебного оборудования. Помещения укомплектованы специализированной мебелью и техническими средствами обучения, служащими для представления учебной информации большой аудитории.

Изучение дисциплины (модуля) обеспечивается в учебных аудиториях, оснащенных:

- столами и стульями;
- компьютерной техникой;
- специализированным оборудованием, включая демонстрационное оборудование.

Помещения для самостоятельной работы обучающихся, в том числе приспособленные для использования инвалидами и лицами с ограниченными возможностями здоровья, оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа к в электронную информационно-образовательную среду Университета.

Обучающимся предоставляется доступ (в том числе удаленный) к ресурсам информационно-телекоммуникационной сети «Интернет», электронным ресурсам (в том числе электронным библиотечным системам, современным профессиональным базам данных и информационным справочным системам):

№	Наименование портала (издания, курса, документа)	Ссылка
1.	Научная электронная библиотека elibrary.ru библиотека	<a href="https://elibrary.ru/defaultx.asp">https://elibrary.ru/defaultx.asp</a>
2.	База данных для IT-специалистов	<a href="https://habr.com">https://habr.com</a>
3.	База данных ScienceDirect	<a href="https://www.sciencedirect.com">https://www.sciencedirect.com</a>
4.	Официальный сайт Министерства науки и высшего образования Российской Федерации	<a href="https://minobrnauki.gov.ru/">https://minobrnauki.gov.ru/</a>
5.	Федеральный портал «Российское образование»	<a href="https://www.edu.ru/">https://www.edu.ru/</a>
6.	Информационная система "Единое окно доступа к образовательным ресурсам"	<a href="http://window.edu.ru/">http://window.edu.ru/</a>
7.	Единая коллекция цифровых образовательных ресурсов	<a href="http://school-collection.edu.ru/">http://school-collection.edu.ru/</a>
8.	Федеральный центр информационно - образовательных ресурсов	<a href="http://fcior.edu.ru/">http://fcior.edu.ru/</a>

Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), в том числе комплект лицензионного программного обеспечения, современные профессиональные базы данных и информационные справочные системы:

Наименование ПО	Производство	Лицензионное / свободно распространяемое
<b>Операционные системы:</b>		
Microsoft Imagine (Windows Client, Server)	зарубежное	лицензионное
<b>Браузеры:</b>		
Яндекс.Браузер	отечественное	свободно распространяемое
Google Chrome	зарубежное	свободно распространяемое
<b>Офисные приложения:</b>		
Microsoft Imagine (Visio, OneNote)	зарубежное	лицензионное
TeXstudio	зарубежное	свободно распространяемое
Adobe Acrobat Reader	зарубежное	свободно распространяемое
<b>Программное обеспечение для планирования и учета времени:</b>		
Toggle app	зарубежное	свободно распространяемое
<b>Системы управления проектами:</b>		
Microsoft Imagine (Project)	зарубежное	лицензионное
<b>Системы управления базами данных:</b>		
Microsoft Imagine (SQL Server)	зарубежное	лицензионное
<b>Системы резервного копирования (backup):</b>		
Acronis Backup Advanced for HyperV	зарубежное	лицензионное
<b>Справочно-правовые системы:</b>		
КонсультантПлюс: справочно-правовая система	отечественное	лицензионное
<b>Средства антивирусной защиты:</b>		
Kaspersky Endpoint Security для бизнеса Стандартный Russian Edition	отечественное	лицензионное
<b>Среды разработки:</b>		
Visual Studio Code	зарубежное	свободно распространяемое
Bash (Unix shell)	зарубежное	свободно распространяемое
Anaconda	зарубежное	свободно распространяемое
Robotic Operating System	зарубежное	свободно распространяемое
CopelliaSim	зарубежное	свободно распространяемое
Google Colaboratory	зарубежное	свободно распространяемое
<b>Пакеты программных средств и библиотек:</b>		

AutoPsy	зарубежное	свободно распространяемое
Interactive Disassembler (IDA)	зарубежное	свободно распространяемое
<b>Системы управления библиографической информацией:</b>		
Zotero	зарубежное	свободно распространяемое
<b>Сервисы и службы:</b>		
Bind	зарубежное	свободно распространяемое
Docker	зарубежное	свободно распространяемое

## 7. Методические и оценочные материалы

### Методические указания для обучающихся по освоению дисциплины (модуля)

В процессе изучения дисциплины (модуля) «Разработка backend-сервисов» в рамках текущего контроля успеваемости используются такие виды учебной работы, как лекции, семинары, контрольные работы, домашние задания, проект, а также различные виды самостоятельной работы обучающихся по заданию преподавателя, направленные на развитие навыков профессиональной лексики, закрепление практических профессиональных компетенций, поощрение инициатив.

*Лекция* – систематическое, последовательное, монологическое изложение преподавателем учебного материала, как правило, теоретического характера.

В процессе лекций рекомендуется вести конспект лекций: кратко и схематично фиксировать основные идеи, выводы и обобщения лекции; выделять важные мысли, ключевые слова и термины. Необходимо отметить вопросы или материалы, которые вызывают затруднения, и попытаться найти ответы в рекомендованной литературе. Если разобраться в материале не удастся, следует сформулировать вопрос и задать его преподавателю на консультации или во время семинарского (практического) занятия.

*Участие в семинаре (аудиторная работа)* – активная работа студента на семинаре, его ответы на вопросы преподавателя и участие в дискуссии.

Для успешного участия в семинаре студентам рекомендуется заранее ознакомиться с темой обсуждения, прочитать необходимые материалы и подготовить вопросы. Важно активно слушать и вовлекаться в дискуссию, высказывая свои мнения и аргументируя их. При ответах на вопросы преподавателя стоит быть уверенным, четким и логичным, опираясь на изученный материал. Также полезно поддерживать диалог с однокурсниками, чтобы обогатить обсуждение и расширить свои знания.

*Домашнее задание* – набор задач по темам недели.

При работе над домашними заданиями важно внимательно ознакомиться с требованиями и сроками выполнения. Рекомендуется разбивать задания на этапы, чтобы избежать перегрузки и лучше усвоить материал. Использовать различные источники информации, включая учебники и онлайн-ресурсы, для более глубокого понимания темы.

*Контрольная работа* – письменная работа с набором задач, которые нужно решить за ограниченное время.

Цель контрольной работы - получить специальные знания по одной или нескольким темам дисциплины (модуля) и продемонстрировать навыки их практического применения.

*Проект* – исследовательская работа по курсу и презентация результатов.

Для успешной подготовки к проекту: четко определите цели и задачи проекта, распределите роли и обязанности между участниками, а также установите сроки выполнения каждой части работы. Регулярно проводите встречи для обсуждения прогресса и решения возникающих вопросов.

*Самостоятельная работа* – работа студентов, направленная на углубленное изучение отдельных тем и вопросов учебной дисциплины (модуля).

В процессе самостоятельной работы студенты взаимодействуют с рекомендованными материалами при минимальном участии преподавателя. Задачи студента включают работу с конспектами лекций (обработка текста), повторное изучение учебных материалов, планов и тезисов ответов, изучение дополнительных тем, выполнение учебно-исследовательских заданий и другое.

### **Система оценивания результатов обучения по дисциплине (модулю)**

#### **Критерии получения уровня и оценивания сформированности компетенций по дисциплине (модулю) «Разработка backend-сервисов»**

Оценивание уровня учебных достижений, обучающихся по дисциплине (модулю), осуществляется в виде текущего контроля успеваемости и промежуточной аттестации.

**Промежуточная аттестация** по дисциплине (модулю) осуществляется в форме *зачета с оценкой*, при этом проводится оценка компетенций, сформированных по дисциплине.

Для оценивания текущего контроля успеваемости и промежуточной аттестации используется десятибалльная шкала оценивания, которая соотносится с традиционной пятибалльной шкалой следующим образом:

Десятибалльная оценка	Пятибалльная оценка	Оценка за зачет	Общая характеристика результата обучения по дисциплине (модулю)
10	Отлично	Зачтено	Студент полностью владеет знаниями, изложенными в рабочей программе, и глубоко осмысляет дисциплину. Он самостоятельно и логически последовательно отвечает на все вопросы, акцентируя внимание на наиболее важном. Умеет анализировать, сравнивать, классифицировать, обобщать, конкретизировать и систематизировать изученный материал, выделяя ключевые моменты и устанавливая причинно-следственные связи. Четко формулирует ответы, уверенно интерпретирует результаты анализов и других исследований, а также решает сложные задачи. Студент хорошо знаком с методами исследования, необходимыми для практической деятельности, и умеет связывать теоретические аспекты дисциплины (модуля) с практическими задачами.
9	Отлично	Зачтено	
8	Отлично	Зачтено	
7	Хорошо	Зачтено	Студент обладает знаниями предмета почти в полном объеме рабочей программы и самостоятельно, логически последовательно и всесторонне отвечает на все вопросы, акцентируя внимание на
6	Хорошо	Зачтено	

Десятибалльная оценка	Пятибалльная оценка	Оценка за зачет	Общая характеристика результата обучения по дисциплине (модулю)
			наиболее значимых моментах. Он умеет анализировать, сравнивать, классифицировать, обобщать, конкретизировать и систематизировать изученный материал, выделяя его ключевые аспекты и устанавливая причинно-следственные связи. Формулирует свои ответы, уверенно интерпретирует результаты анализов и других исследований, а также решает сложные ситуационные задачи. Студент хорошо знаком с методами исследования, необходимыми для практической деятельности, и умеет связывать теоретические аспекты предмета с практическими задачами.
5	Удовлетворительно	Зачтено	Студент обладает базовыми знаниями по дисциплине (модулю), но испытывает трудности при самостоятельных ответах и использует неточные формулировки. В ходе ответов он допускает ошибки, касающиеся сути вопросов. Студент способен решать только самые простые задачи и владеет лишь минимальным набором методов исследования.
4	Удовлетворительно	Зачтено	
3	Не сдан	Не зачтено	Студент не овладел обязательным минимумом знаний по предмету и не может ответить на вопросы, даже если преподаватель задает дополнительные наводящие вопросы.
2	Не сдан	Не зачтено	
1	Не сдан	Не зачтено	

Дисциплина (модуль) «Разработка backend-сервисов» оценивается следующим образом:

Активность	Вес	Количество	Описание
Домашние задания	20%	13	Набор задач по темам недели
Аудиторная работа	15%	14	Активная работа студента на семинаре
Контрольные работы	30%	2	Письменная работа с набором задач, которые нужно решить за ограниченное время
Зачет с оценкой	35%	1	Защита итогового проекта

**Формула расчёта итоговой оценки по дисциплине (модулю) «Разработка backend-сервисов»:**  $\langle 0,2 \times \text{среднее за домашние задания} + 0,15 \times \text{аудиторная работа} + 0,3 \times \text{среднее за контрольные работы} + 0,35 \times \text{зачет с оценкой} \rangle$ .

**Текущий контроль успеваемости обучающихся по дисциплине (модулю)**

## Примерные домашние задания

### Домашнее задание по теме «Контейнер инверсии управления»

1. Создайте простой Spring-проект и определите несколько Spring Beans с помощью аннотаций.
2. Настройте конфигурацию Spring с использованием Java-конфигурации (без XML).
3. Внедрите зависимости в класс через конструктор и сеттеры.
4. Продемонстрируйте работу с областями видимости бинов (singleton и prototype).
5. Зарегистрируйте собственный компонент в контейнере Spring.

### Домашнее задание по теме «Подключение к базе данных»

1. Настройте подключение Spring-приложения к реляционной базе данных (например, PostgreSQL или MySQL).
2. Выполните вставку и выборку данных через JDBC с помощью простых SQL-запросов.
3. Используйте Spring Data JPA для создания репозитория и выполнения CRUD-операций.
4. Определите JPA-сущности с аннотациями для маппинга таблиц базы данных.
5. Реализуйте управление транзакциями при выполнении нескольких связанных операций с БД.

### Домашнее задание по теме «Обработка HTTP-запросов»

1. Создайте REST-контроллер с методами для обработки GET и POST запросов.
2. Настройте маршрутизацию запросов с использованием аннотаций @RequestMapping и @GetMapping.
3. Обработайте параметры запроса и тело запроса в контроллере.
4. Реализуйте обработку ошибок с возвратом корректных HTTP-статусов.
5. Создайте веб-приложение, принимающее и возвращающее JSON-данные.

## Типовые вопросы для подготовки к семинарам

### Вопросы к семинару по теме «Основы Spring Beans»

1. Что такое Spring Bean и как происходит его создание в контейнере Spring?
2. Какие существуют способы регистрации бинов в Spring-приложении?
3. Чем отличаются области видимости бинов singleton и prototype?
4. Как реализуется внедрение зависимостей (Dependency Injection) в Spring?
5. Какие преимущества предоставляет контейнер инверсии управления (IoC) в Spring?

### Вопросы к семинару по теме «ORM и JPA»

1. Что такое ORM и какую роль играет JPA в работе с базами данных?
2. Каковы основные аннотации JPA для определения сущностей и их полей?
3. Чем отличаются методы EntityManager persist, merge и remove?
4. Как реализовать связь между сущностями (OneToOne, OneToMany, ManyToMany) в JPA?
5. Какие существуют стратегии генерации идентификаторов в JPA?

### Вопросы к семинару по теме «Безопасность Spring Security»

1. Какие основные компоненты входят в архитектуру Spring Security?
2. Как настроить аутентификацию пользователей в Spring Security?
3. Что такое роли и как реализовать авторизацию на основе ролей?
4. Как обрабатывать ошибки доступа и перенаправлять пользователей при отсутствии прав?
5. Какие способы защиты от CSRF-атак поддерживает Spring Security?

## Примерные задания по контрольным работам

### Контрольная работа № 1

1. Создайте Spring-проект с Java-конфигурацией без XML.
2. Определите и зарегистрируйте в контейнере Spring три разных бина с использованием аннотаций.
3. Реализуйте внедрение зависимостей через конструктор и сеттеры.
4. Продемонстрируйте работу с областями видимости Spring Beans: singleton и prototype.
5. Настройте подключение к реляционной базе данных (например, PostgreSQL).
6. Создайте JPA-сущность с аннотациями для маппинга таблицы.
7. Реализуйте Spring Data JPA репозиторий для CRUD-операций с сущностью.
8. Выполните вставку, обновление и удаление записи в базе данных через репозиторий.
9. Создайте REST-контроллер с методами GET и POST для работы с сущностью.
10. Обработайте параметры запроса в GET-методе контроллера.
11. Реализуйте получение JSON-запроса в POST-методе и возврат JSON-ответа.
12. Настройте обработку ошибок с возвратом корректных HTTP-статусов (например, 404, 400).
13. Используйте аннотации `@RequestMapping`, `@GetMapping` и `@PostMapping` для маршрутизации.
14. Продемонстрируйте работу с транзакциями при сохранении данных (например, с `@Transactional`).
15. Напишите unit-тест на один из сервисных методов с использованием Spring Test.

### Контрольная работа № 2

1. Настройте Spring Security для аутентификации пользователей с использованием in-memory или JDBC.
2. Реализуйте авторизацию на основе ролей и ограничьте доступ к REST-эндпоинтам.
3. Настройте обработку ошибок при аутентификации и авторизации.
4. Внедрите кэширование данных с использованием Spring Cache (например, `@Cacheable`).
5. Создайте аспект с помощью Spring AOP, который логирует вызовы методов сервиса.
6. Разработайте Spring Boot микросервис с REST API.
7. Настройте управление транзакциями с использованием аннотации `@Transactional` в микросервисе.
8. Реализуйте двунаправленную коммуникацию по WebSocket (сервер и клиент).
9. Интегрируйте внешний REST-сервис, выполнив HTTP-запрос с помощью RestTemplate или WebClient.
10. Настройте обработку ошибок при вызове внешних сервисов.
11. Напишите интеграционные тесты для REST-контроллеров с использованием MockMvc.
12. Используйте профили Spring для конфигурации приложения (например, dev и prod).
13. Реализуйте механизм обновления конфигурации без перезапуска приложения (Spring Cloud Config или аналог).
14. Настройте логирование и мониторинг приложения (например, с помощью Actuator).
15. Проведите отладку приложения и исправьте найденные ошибки.

## Примерное описание и критерии оценивания к проекту

### Описание:

Разработайте backend-приложение на Spring Boot с REST API для управления предметной областью (например, система управления библиотекой, заказами или блогом). Приложение должно включать:

- Конфигурацию и использование Spring Beans с внедрением зависимостей (IoC).
- Подключение и работу с реляционной базой данных через JPA и Spring Data.
- Реализацию основных CRUD-операций через REST-контроллеры с корректной обработкой HTTP-запросов и ошибок.
- Безопасность приложения: аутентификацию и авторизацию пользователей с разграничением доступа по ролям.
- Кэширование наиболее часто запрашиваемых данных для повышения производительности.
- Логирование вызовов сервисных методов с помощью аспектно-ориентированного программирования (AOP).
- Управление транзакциями при работе с базой данных для обеспечения целостности данных.
- Интеграцию с внешним REST-сервисом (например, получение актуальной информации или данных для расширения функционала).
- Реализацию WebSocket для отправки уведомлений пользователям в реальном времени.
- Наличие набора юнит- и интеграционных тестов, покрывающих ключевые компоненты приложения.
- Документацию проекта с описанием функционала и инструкциями по запуску.

### Критерии оценивания:

#### 1. Функциональность и полнота реализации

Проект должен полностью соответствовать техническому заданию: реализованы все ключевые функции (CRUD, безопасность, интеграция, WebSocket и т.д.). Недостающие или частично реализованные функции снижают оценку.

#### 2. Качество архитектуры и использование Spring

Корректное применение принципов IoC и DI, правильное разделение слоёв (контроллеры, сервисы, репозитории), использование Spring Boot и сопутствующих технологий согласно best practices.

#### 3. Работа с базой данных и управление транзакциями

Корректное моделирование данных с помощью JPA, эффективное использование Spring Data, правильная настройка и применение транзакций для предотвращения ошибок и потери данных.

#### 4. Безопасность приложения

Надёжная аутентификация и авторизация с разграничением прав доступа, обработка ошибок безопасности, предотвращение типичных уязвимостей.

#### 5. Производительность и расширяемость

Использование кэширования для оптимизации, логирование через AOP для удобства поддержки и диагностики, а также гибкая архитектура, позволяющая легко расширять функционал.

#### 6. Интеграция и коммуникация

Корректная и надёжная интеграция с внешними REST-сервисами, обработка ошибок при взаимодействии, а также работа WebSocket для обмена сообщениями в реальном времени.

#### 7. Тестирование и документация

Наличие покрывающих тестов (юнит и интеграционных) с хорошим покрытием ключевых компонентов, понятная и полная документация, позволяющая запустить и использовать приложение без дополнительных пояснений.

### Задания для промежуточной аттестации по дисциплине (модулю)

№ п/п	Задание	Ответ	Компетенция
1.	Что такое контейнер инверсии управления (IoC) в Spring Framework? а) Компонент, который управляет жизненным циклом объектов и внедрением зависимостей б) Система безопасности для приложений в) Средство для создания REST API г) Механизм кэширования данных	а	УК-1
2.	Какой HTTP-метод обычно используется для обновления ресурса в REST API?	PUT	УК-1
3.	Какой стандарт описывает объектно-реляционное отображение (ORM) в Java?	JPA	УК-2
4.	Какой механизм в Spring используется для управления транзакциями?	@Transactional	УК-2
5.	Как называется объект, управляемый Spring контейнером, который создаётся, настраивается и управляется в течение жизненного цикла приложения? а) Service б) Repository в) Entity г) Bean	г	ОПК-1
6.	Какой метод EntityManager используется для сохранения нового объекта в базу данных? а) merge() б) remove() в) persist() г) find()	в	ПК-1
7.	Какой механизм Spring Security отвечает за аутентификацию пользователя? а) AuthorizationManager б) AuthenticationManager в) CacheManager г) TransactionManager	б	ПК-2
8.	Как называется процесс внедрения зависимостей в объекты Spring?	Инверсия управления	УК-1
9.	Как называется язык запросов, используемый в JPA?	JPQL	УК-2
10.	Как называется аннотация для обработки HTTP GET-запроса в Spring?	@GetMapping	ОПК-1
11.	Какой протокол используется для двунаправленной связи в Spring WebSocket?	STOMP	ПК-2