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)

Технологии хранения и обработки данных

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 формирует системное представление о технологиях хранения и обработки данных, применяемых при разработке программных продуктов. Первый семестр посвящен реляционным базам данных на примере PostgreSQL и охватывает концептуальное моделирование, реляционную модель, нормализацию, язык SQL, ограничения целостности, индексы, представления, транзакции, оптимизацию запросов, развертывание базы данных и программный доступ из Java через JDBC, JPA и Hibernate. Второй семестр посвящен современным нереляционным и аналитическим технологиям хранения данных, включая MongoDB, Redis, ClickHouse, Prometheus, стек Elasticsearch, Logstash, Kibana, а также основы построения потоков подготовки данных с использованием dbt и Airflow. На практических и лабораторных занятиях обучающиеся последовательно проектируют модель данных, реализуют схемы и запросы, подключают Java-приложение к базе данных, осваивают разные модели хранения, выполняют загрузку, обработку и наблюдение данных, а также готовят техническое описание принятых решений.

Целью освоения дисциплины является формирование способности использовать современные технологии хранения и обработки данных и разрабатывать программные решения на Java, работающие с реляционными, нереляционными, аналитическими и временными хранилищами в условиях требований целостности, производительности, безопасности и сопровождаемости.

Для достижения поставленной цели в рамках дисциплины решается комплекс задач, направленных на формирование у обучающихся способности – проектировать концептуальные и логические модели данных, нормализовать реляционные схемы, создавать объекты PostgreSQL, писать запросы SQL, управлять транзакциями и индексами, подключать Java-приложения к базе данных через JDBC, JPA и Hibernate, выбирать модель хранения данных под характер задачи, использовать MongoDB, Redis, ClickHouse, Prometheus и стек Elasticsearch, Logstash, Kibana, строить воспроизводимые потоки подготовки данных и оформлять техническую документацию.

2. Планируемые результаты обучения по дисциплине (модулю).

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

ДПК-БП-4 - Способен проектировать и реализовывать архитектуру данных нагруженного бизнес-приложения;

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

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

Знать:

- роль данных, базы данных, системы управления базами данных и информационной системы в программном продукте
- уровни абстракции в системе управления базами данных и распределение ответственности между приложением и хранилищем
- концептуальное моделирование предметной области, сущности, атрибуты, связи и ограничения
- нотации моделирования данных, включая нотацию Чена, IDEF1X и связи с внешними ключами
- реляционную модель данных, домены, атрибуты, кортежи, отношения, ключи и ограничения целостности
- операции реляционной алгебры и их связь с конструкциями SQL
- устройство PostgreSQL, базы данных, схемы, таблицы, типы данных, представления, индексы и табличные пространства
- язык SQL, выборку, соединения, подзапросы, агрегирование, группировку, внешние соединения и изменение данных
- функциональные зависимости, избыточность, аномалии, ключи, замыкание и покрытие зависимостей
- нормальные формы отношений, включая первую, вторую, третью нормальные формы, нормальную форму Бойса-Кодда и четвертую нормальную форму
- транзакции, свойства ACID, уровни изоляции и проблемы параллельного доступа к данным
- принципы оптимизации запросов PostgreSQL, планы выполнения, статистику, индексы и выбор стратегии соединения
- программный доступ к данным из Java через JDBC, пул соединений и обработку ошибок работы с базой данных
- принципы объектно-реляционного отображения, JPA, Hibernate, сущности, связи, контекст сохранения и транзакции приложения
- модели нереляционных хранилищ, включая ключ-значение, документную, колоночную, графовую и временную модели

- устройство MongoDB, коллекции, документы, индексы, агрегирование и проектирование документной модели

- устройство Redis, структуры данных, срок жизни ключей, кэширование и сценарии быстрых операций

- устройство ClickHouse, колоночное хранение, партиционирование, сжатие и аналитические запросы

- устройство Prometheus как временного хранилища метрик и принципы запроса временных рядов

- назначение Elasticsearch, Logstash и Kibana при поиске, загрузке, хранении и визуальном анализе событий

- назначение dbt и Airflow при построении воспроизводимых потоков подготовки и обработки данных

Уметь:

- уметь моделировать предметную область при помощи ER-диаграмм и IDEF1X в условиях разработки информационной системы

- уметь проектировать реляционную схему при помощи PostgreSQL и нормальных форм в условиях требований целостности и минимизации избыточности

- уметь создавать объекты базы данных при помощи SQL в PostgreSQL в условиях ограничений домена, ключей и связей между таблицами

- уметь писать запросы выборки и изменения данных при помощи SQL в PostgreSQL в условиях нескольких связанных таблиц

- уметь оптимизировать запросы при помощи EXPLAIN, индексов и анализа плана выполнения в условиях ограничений времени ответа

- уметь управлять транзакциями при помощи PostgreSQL в условиях параллельного доступа и разных уровней изоляции

- уметь подключать Java-приложение к базе данных при помощи JDBC в условиях безопасной обработки параметров и ошибок

- уметь реализовывать слой доступа к данным при помощи JPA и Hibernate в условиях работы с сущностями, связями и транзакциями приложения

- уметь проектировать документную модель при помощи MongoDB в условиях вложенных данных и запросов по документам

- уметь использовать Redis при помощи структур данных и сроков жизни ключей в условиях кэширования и быстрых операций

- уметь проектировать аналитическую таблицу при помощи ClickHouse в условиях больших объемов событий и агрегирующих запросов

- уметь хранить и запрашивать временные ряды при помощи Prometheus в условиях сбора метрик приложения

- уметь загружать и исследовать журнальные события при помощи Elasticsearch, Logstash и Kibana в условиях поиска и фильтрации сообщений

- уметь описывать преобразования данных при помощи dbt и планировать выполнение потоков при помощи Airflow в условиях воспроизводимой обработки

- уметь готовить техническую документацию по модели данных, схеме хранения, запросам, ограничениям, транзакциям и программному доступу

Владеть:

- навыком построения концептуальной и логической модели данных для прикладной информационной системы

- навыком создания схемы PostgreSQL с ограничениями целостности, индексами и представлениями

- навыком разработки SQL-запросов разной сложности и оценки их плана выполнения

- навыком управления транзакциями и проверкой параллельного доступа к данным

- навыком разработки Java-кода для доступа к данным через JDBC, JPA и Hibernate

- навыком выбора модели хранения данных под функциональные и эксплуатационные требования

- навыком работы с MongoDB, Redis, ClickHouse, Prometheus и стеком Elasticsearch, Logstash, Kibana

- навыком подготовки воспроизводимого потока обработки данных с использованием dbt и Airflow

- навыком подготовки технической документации по хранению, обработке и программному доступу к данным

3. Объем дисциплины (модуля).

3.1. Общая трудоемкость дисциплины (модуля).

Общая трудоемкость дисциплины (модуля) составляет 8 з.е. (288 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

		№3	№4
Контактная работа при проведении учебных занятий (всего):	160	80	80
В том числе:			
Занятия лекционного типа	64	32	32
Занятия семинарского типа	96	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 128 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	Базы данных и системы управления данными Рассматриваемые вопросы: - база данных, предметная область, информационная система и система управления базами данных; - достоинства централизованного хранения данных и уровни абстракции; - роли разработчика, администратора данных и приложения в жизненном цикле данных.
2	Концептуальное моделирование предметной области Рассматриваемые вопросы: - сущности, атрибуты, идентификаторы и ограничения предметной области; - связи один к одному, один ко многим и многие ко многим; - нотация Чена и переход от описания предметной области к модели данных.
3	Логическое моделирование и IDEF1X Рассматриваемые вопросы: - сущности, первичные ключи, внешние ключи и зависимые сущности; - обязательные и необязательные связи между сущностями; - преобразование концептуальной модели в логическую схему.
4	Реляционная модель данных Рассматриваемые вопросы: - домен, атрибут, кортеж, отношение и схема отношения; - первичный, потенциальный и внешний ключ; - ограничения целостности и связь реляционной модели с таблицами SQL.

№ п/п	Тематика лекционных занятий / краткое содержание
5	Реляционная алгебра и основы SQL Рассматриваемые вопросы: - выборка, проекция, соединение, объединение, разность и декартово произведение; - выражение операций реляционной алгебры средствами SQL; - общая структура запроса и порядок логической обработки данных.
6	Архитектура PostgreSQL Рассматриваемые вопросы: - база данных, схема, табличное пространство и объектная организация PostgreSQL; - типы данных, последовательности, ограничения и представления; - роль системного каталога и прав доступа в PostgreSQL.
7	Создание реляционной схемы в PostgreSQL Рассматриваемые вопросы: - создание и изменение таблиц, первичных и внешних ключей; - ограничения NOT NULL, UNIQUE, CHECK и ссылочная целостность; - каскадные действия и правила изменения связанных данных.
8	Запросы выборки в SQL Рассматриваемые вопросы: - фильтрация, сортировка, выражения и работа с неопределенными значениями; - соединения нескольких таблиц и подзапросы; - квантор существования и проверка наличия связанных данных.
9	Агрегирование и аналитические возможности SQL Рассматриваемые вопросы: - агрегатные функции, группировка и фильтрация групп; - внешние соединения и сохранение строк без соответствий; - оконные функции, разбиение, порядок и рамка окна.
10	Изменение данных и представления Рассматриваемые вопросы: - вставка, обновление и удаление строк; - представления, материализованные представления и ограничения обновляемости; - подготовка тестовых данных и контроль изменений.
11	Функциональные зависимости и нормализация Рассматриваемые вопросы: - избыточность, аномалии вставки, удаления и обновления; - функциональная зависимость, ключ, замыкание и покрытие зависимостей; - первая, вторая, третья нормальные формы и нормальная форма Бойса-Кодда.
12	Многозначные зависимости и качество схемы Рассматриваемые вопросы: - многозначная зависимость и четвертая нормальная форма; - декомпозиция без потерь и сохранение зависимостей; - компромисс между нормализацией, простотой запросов и производительностью.
13	Индексы и оптимизация запросов PostgreSQL Рассматриваемые вопросы: - назначение индексов и основные типы индексов PostgreSQL; - план выполнения запроса, статистика и EXPLAIN; - выбор стратегии соединения и причины медленных запросов.
14	Транзакции и параллельный доступ Рассматриваемые вопросы: - свойства ACID и границы транзакции; - проблемы параллельного доступа и уровни изоляции; - блокировки, версии строк и разрешение конфликтов.

№ п/п	Тематика лекционных занятий / краткое содержание
15	Доступ к данным из Java через JDBC Рассматриваемые вопросы: - соединение с PostgreSQL, подготовленные запросы и обработка результатов; - пул соединений, параметры подключения и обработка ошибок; - безопасность параметров запроса и управление транзакциями приложения.
16	Объектно-реляционное отображение в Java Рассматриваемые вопросы: - сущности JPA, связи, контекст сохранения и жизненный цикл объекта; - средство Hibernate, ленивые загрузки, каскады и проблема лишних запросов; - границы слоя доступа к данным в Java-приложении.
17	Нереляционные модели хранения данных Рассматриваемые вопросы: - причины появления нереляционных хранилищ и ограничения универсальной реляционной модели; - модели ключ-значение, документная, колоночная, графовая и временная; - выбор модели хранения по характеру запросов, объему и изменчивости данных.
18	Распределенные данные и согласованность Рассматриваемые вопросы: - масштабирование, репликация, секционирование и отказоустойчивость; - теорема CAP и компромиссы согласованности, доступности и устойчивости к разделению сети; - требования к распределенной базе данных в прикладной системе.
19	Документная модель MongoDB Рассматриваемые вопросы: - база данных, коллекция, документ и вложенная структура; - проектирование документа с учетом сценариев чтения и записи; - ограничения денормализации и дублирования данных.
20	Запросы и агрегирование в MongoDB Рассматриваемые вопросы: - фильтрация документов, проекция, сортировка и обновление; - индексы MongoDB и влияние индекса на поиск; - конвейер агрегирования и преобразование документов.
21	Хранилище ключ-значение Redis Рассматриваемые вопросы: - ключи, строки, списки, множества, хэши и упорядоченные множества; - срок жизни ключей, вытеснение и кэширование; - сценарии использования Redis в Java-приложении.
22	Кэширование и согласованность данных Рассматриваемые вопросы: - шаблоны чтения через кэш и записи через кэш; - устаревание данных, инвалидация и срок жизни; - риски несогласованности между PostgreSQL и Redis.
23	Колоночная аналитическая СУБД ClickHouse Рассматриваемые вопросы: - колоночное хранение, сжатие и векторная обработка; - таблицы MergeTree, партиционирование и сортировочный ключ; - агрегирующие запросы по большим объемам событий.
24	Загрузка и подготовка данных для аналитики Рассматриваемые вопросы: - пакетная загрузка, потоковая загрузка и контроль качества данных;

№ п/п	Тематика лекционных занятий / краткое содержание
	- преобразование типов, очистка и дедубликация; - витрины данных и проверяемость расчетных показателей.
25	Временные ряды и Prometheus Рассматриваемые вопросы: - метрика, метка, временной ряд и модель хранения Prometheus; - запросы PromQL и агрегирование временных рядов; - ограничения Prometheus как хранилища прикладных данных.
26	Журнальные события и стек Elasticsearch, Logstash, Kibana Рассматриваемые вопросы: - событие, индекс, документ и полнотекстовый поиск; - загрузка и преобразование событий через Logstash; - визуальное исследование данных в Kibana.
27	Сопоставление хранилищ под требования приложения Рассматриваемые вопросы: - транзакционные, документные, кэшированные, аналитические и временные данные; - критерии выбора PostgreSQL, MongoDB, Redis, ClickHouse и Prometheus; - стоимость хранения, скорость ответа и сложность сопровождения.
28	Программный доступ к нереляционным хранилищам из Java Рассматриваемые вопросы: - драйверы Java для MongoDB, Redis и ClickHouse; - преобразование объектов приложения в документы, ключи и строки аналитических таблиц; - обработка ошибок, повторные попытки и ограничения соединений.
29	Потоки преобразования данных с dbt Рассматриваемые вопросы: - модель преобразования данных, источники, модели и зависимости; - проверка качества данных и документирование преобразований; - применение dbt для построения витрин поверх аналитического хранилища.
30	Планирование обработки данных с Airflow Рассматриваемые вопросы: - направленный ациклический граф задач, оператор и расписание; - зависимости задач, повтор выполнения и контроль ошибок; - запуск регулярных потоков загрузки и преобразования данных.
31	Безопасность, резервирование и сопровождение хранилищ Рассматриваемые вопросы: - учетные записи, роли, права доступа и защита соединений; - резервное копирование, восстановление и проверка восстановления; - журналы, аудит и эксплуатационная документация.
32	Архитектура прикладной системы хранения и обработки данных Рассматриваемые вопросы: - сочетание транзакционного, кэшированного, документного, аналитического и временного хранилища; - границы ответственности Java-приложения, базы данных и потока обработки; - техническая документация, ограничения решения и передача на сопровождение.

4.2. Занятия семинарского типа.

Лабораторные работы

№ п/п	Наименование лабораторных работ / краткое содержание
1	Развертывание PostgreSQL и создание базы данных Студент устанавливает PostgreSQL или Postgres Pro в учебной среде. Создается база данных, пользователь, схема и тестовое подключение. Параметры подключения фиксируются в техническом журнале.
2	Создание таблиц и ограничений целостности Студент реализует таблицы по логической модели. Добавляются первичные и внешние ключи, ограничения уникальности, проверки значений и правила ссылочной целостности. Схема проверяется на корректных и ошибочных данных.
3	Заполнение и изменение данных средствами SQL Студент подготавливает набор тестовых данных для нескольких связанных таблиц. Выполняются операции вставки, обновления и удаления. Проверяется поведение ограничений и каскадных действий.
4	Запросы выборки к связанным таблицам Студент реализует запросы с фильтрацией, сортировкой, соединениями и подзапросами. Для каждого запроса проверяется соответствие результата поставленному условию. Ошибочные или избыточные запросы исправляются.
5	Агрегирование, внешние соединения и оконные функции Студент реализует запросы с группировкой, агрегатами и внешними соединениями. Дополнительно выполняются оконные расчеты по группам данных. Результаты сверяются на небольшом контрольном наборе строк.
6	Представления и материализованные представления Студент создает представления для часто используемых запросов. Для расчетного отчета создается материализованное представление и выполняется его обновление. Проверяется различие между обычным и материализованным представлением.
7	Индексы и планы выполнения запросов Студент выполняет запросы до и после создания индексов. С помощью EXPLAIN сравниваются планы выполнения и способы соединения таблиц. В отчете фиксируется влияние индекса на выбранные запросы.
8	Транзакции и уровни изоляции Студент выполняет параллельные сценарии изменения данных в двух сеансах PostgreSQL. Проверяются эффекты разных уровней изоляции и блокировок. Результаты оформляются как таблица наблюдений.
9	Резервное копирование и восстановление PostgreSQL Студент выполняет выгрузку схемы и данных учебной базы. Затем база восстанавливается в отдельное окружение. Проверяется сохранность таблиц, ограничений и данных.
10	Доступ к PostgreSQL из Java через JDBC Студент создает приложение на Java для выполнения параметризованных запросов. Реализуются чтение, вставка и обновление данных с обработкой ошибок. Отдельно проверяется закрытие соединений и ресурсов.
11	Пул соединений и транзакции Java-приложения Студент подключает пул соединений к приложению на Java. Выполняется транзакционный сценарий с фиксацией и откатом изменений. Проверяется поведение приложения при ошибке внутри транзакции.
12	Сущности JPA и отображение таблиц Студент описывает сущности JPA для таблиц учебной базы. Настраиваются идентификаторы, простые поля и связи между сущностями. Выполняется чтение и сохранение объектов приложения.
13	Работа с Hibernate и запросами к сущностям Студент реализует репозитории и запросы к сущностям. Проверяется ленивое получение связанных

№ п/п	Наименование лабораторных работ / краткое содержание
	данных и влияние каскадных операций. Результаты сопоставляются с запросами SQL, выполняемыми Hibernate.
14	Миграции схемы базы данных Студент подготавливает последовательность изменений схемы для учебной базы данных. Миграции применяются к пустой и заполненной базе. Проверяется воспроизводимость структуры после повторного развертывания.
15	Развертывание Java-приложения с PostgreSQL Студент запускает приложение на Java и PostgreSQL в локальной среде или контейнерной конфигурации. Проверяется подключение, выполнение основных сценариев и сохранение данных. Параметры среды фиксируются в файле настроек.
16	Итоговая проверка реляционного решения Студент объединяет схему PostgreSQL, данные, запросы SQL, транзакционные сценарии и доступ из Java. Выполняется контрольный запуск основных операций приложения. Итоговый материал оформляется как связное портфолио реляционного решения.
17	Развертывание MongoDB и создание документной базы Студент устанавливает MongoDB или совместимую среду. Создаются база, коллекции и набор документов по ранее подготовленной модели. Проверяется подключение из командной оболочки и приложения на Java.
18	Запросы и индексы MongoDB Студент выполняет фильтрацию, сортировку, обновление и удаление документов. Создаются индексы для основных запросов. Результаты запросов сравниваются до и после индексирования.
19	Агрегирование документов MongoDB Студент реализует конвейер агрегирования для расчета показателей по документам. Используются этапы отбора, группировки и преобразования структуры результата. Итог сверяется с контрольной выборкой.
20	Доступ к MongoDB из Java Студент подключает приложение на Java к MongoDB. Реализуются операции создания, чтения, изменения и удаления документов. Проверяется обработка ошибок подключения и некорректных данных
21	Развертывание Redis и работа со структурами данных Студент запускает Redis и выполняет операции со строками, списками, множествами, хэшами и упорядоченными множествами. Для ключей задается срок жизни. Результаты проверяются через командную оболочку.
22	Кэширование данных Java-приложения с Redis Студент добавляет Redis в сценарий чтения данных приложения. Реализуется чтение через кэш и обновление значения при изменении источника. Проверяется поведение при устаревании и удалении ключа.
23	Развертывание ClickHouse и создание аналитической таблицы Студент устанавливает ClickHouse или подключается к учебному серверу. Создается таблица для событий с выбранным движком, партиционированием и сортировочным ключом. Загружается тестовый набор событий.
24	Аналитические запросы ClickHouse Студент выполняет агрегирующие запросы по большим наборам событий. Проверяются группировки, фильтрация по времени и расчет показателей. Результаты фиксируются для нескольких временных интервалов.
25	Сбор и запрос временных рядов Prometheus Студент настраивает сбор учебных метрик приложения или демонстрационного источника. Выполняются запросы PromQL для отбора и агрегирования временных рядов. Проверяется влияние меток на результат запроса.

№ п/п	Наименование лабораторных работ / краткое содержание
26	Загрузка журнальных событий в Elasticsearch Студент подготавливает поток журнальных событий через Logstash. Данные загружаются в Elasticsearch с заданной структурой полей. В Kibana выполняется поиск и фильтрация событий.
27	Построение визуального представления данных в Kibana Студент создает несколько представлений для анализа событий. Настраиваются фильтры, временной интервал и группировки. Результат используется для проверки гипотез о поведении приложения.
28	Преобразования данных с dbt Студент описывает источники, промежуточные модели и итоговую витрину данных. Запускаются преобразования и проверки качества данных. Документация моделей создается средствами dbt.
29	Планирование потока обработки данных в Airflow Студент создает направленный граф задач для загрузки и преобразования данных. Настраивается расписание, зависимости и повтор выполнения при ошибке. Выполняется пробный запуск потока.
30	Интеграция нескольких хранилищ в Java-приложении Студент подключает приложение на Java к двум или нескольким хранилищам данных. Реализуется сценарий чтения транзакционных данных, кэширования и записи события для аналитики. Проверяется согласованность результата на контрольном сценарии.
31	Резервирование и восстановление данных Студент выполняет резервное копирование выбранных хранилищ и восстановление в отдельное окружение. Проверяется доступность данных после восстановления. Итоги фиксируются в техническом журнале.
32	Итоговая проверка системы хранения и обработки данных Студент объединяет реляционное, документное, кэшированное, аналитическое и временное хранение в единую учебную архитектуру. Выполняется контрольный сценарий загрузки, обработки, поиска и чтения данных. Итоговый материал оформляется как связное портфолио системы данных.

Практические занятия

№ п/п	Тематика практических занятий/краткое содержание
1	Анализ предметной области и границ данных Студент описывает выбранную предметную область, участников, основные сущности, события и ограничения. Для каждой сущности фиксируются атрибуты, идентификаторы и правила существования. Результат оформляется как текстовая спецификация данных.
2	Построение концептуальной модели данных Студент строит модель сущностей и связей в нотации Чена. Для каждой связи определяется кратность, обязательность и смысловое ограничение. Модель проверяется на полноту относительно выбранных пользовательских сценариев.
3	Построение логической модели IDEF1X Студент преобразует концептуальную модель в логическую модель IDEF1X. Определяются первичные ключи, внешние ключи, зависимые сущности и справочники. Результат становится основой для дальнейшей физической схемы.
4	Нормализация реляционной схемы Студент выявляет функциональные зависимости и возможные аномалии изменения данных. Отношения приводятся к третьей нормальной форме или нормальной форме Бойса и Кодда. Для каждой декомпозиции фиксируется причина и ожидаемый эффект.
5	Проектирование ограничений целостности Студент определяет доменные, сущностные и ссылочные ограничения. Для таблиц задаются правила обязательности, уникальности, допустимых значений и каскадных действий. Итоговая спецификация используется при создании схемы PostgreSQL.

№ п/п	Тематика практических занятий/краткое содержание
6	Проектирование запросов к реляционной схеме Студент формулирует набор запросов к нескольким связанным таблицам. Для каждого запроса определяется требуемый результат, условия отбора, группировка и сортировка. Отдельно выделяются запросы для отчетов и проверки целостности данных.
7	Проектирование транзакционных сценариев Студент описывает операции, которые должны выполняться атомарно. Для каждого сценария фиксируются изменяемые таблицы, возможные конфликты и требуемый уровень изоляции. Результат используется для лабораторной проверки параллельного доступа.
8	Проектирование слоя доступа к данным Java-приложения Студент определяет сущности приложения, репозитории, запросы и границы транзакций. Для JDBC и JPA выделяются разные обязанности и ограничения. Итогом становится схема программного доступа к PostgreSQL.
9	Выбор модели нереляционного хранения Студент рассматривает набор сценариев чтения, записи и аналитики для выбранной предметной области. Для каждого сценария выбирается подходящая модель хранения данных. Решение оформляется как матрица соответствия требований и хранилищ.
10	Проектирование документной модели Студент выделяет данные, которые целесообразно хранить в виде документов. Определяется структура вложенных объектов, идентификаторы и индексы. Модель проверяется на соответствие основным запросам приложения.
11	Проектирование кэширования и быстрых операций Студент выбирает данные и операции, подходящие для Redis. Для ключей определяются структура, срок жизни и правило обновления. Отдельно фиксируются риски устаревания и способы инвалидации.
12	Проектирование аналитического хранения событий Студент определяет поток событий и показатели, которые должны рассчитываться в ClickHouse. Задаются измерения, факты, партиционирование и сортировочный ключ. Результат используется для создания аналитической таблицы.
13	Проектирование временных рядов и журнальных событий Студент разделяет метрики и журнальные события по назначению и структуре. Для метрик задаются имена и метки, для событий поля поиска и фильтрации. Итогом становится схема наблюдаемых данных.
14	Проектирование потока подготовки данных Студент описывает источники, преобразования, проверки качества и итоговые витрины данных. Для каждого шага фиксируется вход, выход и зависимость от предыдущих шагов. Модель используется для реализации в dbt и Airflow.
15	Сопоставление хранилищ в архитектуре приложения Студент определяет, какие данные остаются в PostgreSQL, какие переносятся в MongoDB, Redis, ClickHouse, Prometheus и Elasticsearch. Для каждого хранилища фиксируется причина выбора и ограничение применения. Итогом становится архитектурная схема данных.
16	Подготовка технической документации по системе данных Студент собирает описание моделей, схем, индексов, запросов, потоков обработки и правил доступа. Документация проверяется на полноту для передачи решения команде разработки. Итоговый материал оформляется как единое описание системы хранения и обработки данных.

4.3. Самостоятельная работа обучающихся.

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

4.4. Примерный перечень тем курсовых работ

Разработка программного обеспечения для работы с реляционной базой данных в предметной области «Ресторан».

Разработка программного обеспечения для работы с реляционной базой данных в предметной области «Реестр ценных бумаг».

Разработка программного обеспечения для работы с реляционной базой данных в предметной области «Склад магазина».

Разработка программного обеспечения для работы с реляционной базой данных в предметной области «Музыкальный магазин».

Разработка программного обеспечения для работы с реляционной базой данных в предметной области «Аэропорт».

Разработка программного обеспечения для работы с реляционной базой данных в предметной области «Университет».

Разработка программного обеспечения для работы с реляционной базой данных в предметной области «Клиника».

Разработка программного обеспечения для работы с реляционной базой данных в предметной области «Авиакомпания».

Разработка программного обеспечения для работы с реляционной базой данных в предметной области «Логистическая компания».

Разработка программного обеспечения для работы с реляционной базой данных в предметной области «Библиотека».

Разработка программного обеспечения для работы с реляционной базой данных в предметной области «Гостиничный комплекс».

Разработка программного обеспечения для работы с реляционной базой данных в предметной области «Грузоперевозки».

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

№ п/п	Библиографическое описание	Место доступа

1	Токмаков, Г. П. Базы данных: Модели и структуры данных, язык SQL, программирование баз данных : учебное пособие / Г. П. Токмаков. — Ульяновск : УлГТУ, 2021. — 362 с. — ISBN 978-5-9795-2184-8. — Текст : электронный	Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/259706 (дата обращения: 22.06.2026)
2	Фешина, Е. В. Базы данных : учебник / Е. В. Фешина, В. В. Ткаченко. — Краснодар : КубГАУ, 2020. — 172 с. — ISBN 978-5-907402-36-2. — Текст : электронный	Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/254261 (дата обращения: 22.06.2026)
3	Домбровская, Г. Оптимизация запросов PostgreSQL / Г. Домбровская, Б. Новиков, А. Бейликова ; перевод с английского Д. А. Беликова. — Москва : ДМК Пресс, 2021. — 278 с. — ISBN 978-5-97060-963-7. — Текст : электронный	Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/241103 (дата обращения: 22.06.2026)
4	Григорьев, Ю. А. Реляционные базы данных и системы NoSQL : учебное пособие / Ю. А. Григорьев, А. Д. Плутенко, О. Ю. Плужникова. — Благовещенск : АмГУ, 2018. — 424 с. — ISBN 978-5-93493-308-2. — Текст : электронный	Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/156492 (дата обращения: 22.06.2026)

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/.

Документация PostgreSQL – https://www.postgresql.org/docs/.

Документация Java Platform, Standard Edition – https://docs.oracle.com/en/java/javase/.

Документация Hibernate ORM – https://hibernate.org/orm/documentation/.

Документация MongoDB – https://www.mongodb.com/docs/.

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

Документация ClickHouse – https://clickhouse.com/docs/.

Документация Prometheus – https://prometheus.io/docs/.

Документация Elasticsearch – https://www.elastic.co/guide/.

Документация dbt – https://docs.getdbt.com/.

Документация Apache Airflow – https://airflow.apache.org/docs/.

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.

Реляционные СУБД – PostgreSQL, Postgres Pro.

Средства разработки – OpenJDK, IntelliJ IDEA Community Edition или Eclipse IDE, Maven или Gradle.

Доступ к данным Java – JDBC, JPA, Hibernate.

Нереляционные и аналитические хранилища – MongoDB, Redis, ClickHouse, Prometheus, Elasticsearch, Logstash, Kibana.

Потоки обработки данных – dbt, Apache Airflow.

Сопровождение проекта – Git, DBeaver Community, pgAdmin.

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:

Курсовая работа в 3 семестре.

Зачет в 3, 4 семестрах.

10. Оценочные материалы.

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

Авторы:

доцент кафедры «Цифровые технологии управления транспортными процессами» Института железнодорожного транспорта — И.И. Бутов

Согласовано:

Заведующий кафедрой ЦТУТП — В.Е. Нутович

Директор ЦС IT - программ ВИШ — О.Б. Проневич

Председатель учебно-методической комиссии — Н.А. Андриянова