

**Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего
образования
«ИРКУТСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ»**

Структурное подразделение «Сибирская школа геонаук (119)»

УТВЕРЖДЕНА:
на заседании ДОТ
Протокол №29 от 10 апреля 2025 г.

Рабочая программа дисциплины

**«АРХИТЕКТУРА ИНФОРМАЦИОННЫХ СИСТЕМ / INFORMATION SYSTEM
ARCHITECTURE»**

Направление: 09.03.02 Информационные системы и технологии

Информационные технологии в науках о Земле и окружающей среде / Information
Technologies in Earth and Environmental Sciences

Квалификация: Бакалавр

Форма обучения: очная

Документ подписан простой
электронной подписью
Составитель программы:
Ланько Анна Викторовна
Дата подписания: 18.12.2025

Документ подписан простой
электронной подписью
Утвердил: Ланько Анна
Викторовна
Дата подписания: 18.12.2025

Документ подписан простой
электронной подписью
Согласовал: Паршин
Александр Вадимович
Дата подписания: 13.01.2026

Год набора – 2025

Иркутск, 2025 г.

1 Перечень планируемых результатов обучения по дисциплине, соотнесённых с планируемыми результатами освоения образовательной программы

1.1 Дисциплина «Архитектура информационных систем / Information System Architecture» обеспечивает формирование следующих компетенций с учётом индикаторов их достижения

Код, наименование компетенции	Код индикатора компетенции
ОПК ОС-7 Способен осуществлять выбор платформ и инструментальных программно-аппаратных средств для реализации информационных систем	ОПК ОС-7.2

1.2 В результате освоения дисциплины у обучающихся должны быть сформированы

Код индикатора	Содержание индикатора	Результат обучения
ОПК ОС-7.2	Способен оценивать выбор платформ и аппаратно-программных средств для реализации информационной системы	<p>Знать архитектурные домены ИС (бизнес, данные, приложения), модели (клиент-сервер, SOA, MVC), платформы (web, облачные, NoSQL/реляционные СУБД) и их применение для геоданных.</p> <p>Уметь анализировать требования ИС недропользования, сравнивать платформы (on-premise vs cloud) и аппаратно-программные стеки по критериям производительности, масштабируемости и стоимости для геообработки.</p> <p>Владеть методиками выбора платформ (TCO, benchmarking), инструментами моделирования (UML, IDEF0) и паттернами (Zachman) для обоснования аппаратно-программных средств в проектах ГИС недропользования.</p>

2 Место дисциплины в структуре ООП

Изучение дисциплины «Архитектура информационных систем / Information System Architecture» базируется на результатах освоения следующих дисциплин/практик: «Архитектура ЭВМ и систем / Computer and System Architecture»

Дисциплина является предшествующей для дисциплин/практик: «Инфокоммуникационные системы и сети / Infocommunication Systems and Networks»

3 Объем дисциплины

Объем дисциплины составляет – 3 ЗЕТ

Вид учебной работы	Трудоемкость в академических часах (Один академический час соответствует 45 минутам астрономического часа)
--------------------	---

	Всего	Семестр № 4
Общая трудоемкость дисциплины	108	108
Аудиторные занятия, в том числе:	54	54
лекции	18	18
лабораторные работы	36	36
практические/семинарские занятия	0	0
Самостоятельная работа (в т.ч. курсовое проектирование)	54	54
Трудоемкость промежуточной аттестации	0	0
Вид промежуточной аттестации (итогового контроля по дисциплине)	Зачет	Зачет

4 Структура и содержание дисциплины

4.1 Сводные данные по содержанию дисциплины

Семестр № 4

№ п/п	Наименование раздела и темы дисциплины	Виды контактной работы						СРС		Форма текущего контроля
		Лекции		ЛР		ПЗ(СЕМ)				
№	Кол. Час.	№	Кол. Час.	№	Кол. Час.	№	Кол. Час.	№	Кол. Час.	
1	2	3	4	5	6	7	8	9	10	11
1	1. Понятие архитектуры ИС	1	4	1	4					Устный опрос
2	2. Функциональное моделирование ИС	2	2	2	5			3	5	Устный опрос
3	3. Клиент-серверная архитектура	3	2	3	5			1	5	Устный опрос
4	4. Распределенные ИС и SOA	4	2	4	5			5	10	Устный опрос
5	5. Архитектуры web-приложений	5	2	5	5			4	5	Устный опрос
6	6. СУБД и хранилища данных	6	2	6	6			1	5	Устный опрос
7	7. Архитектурные стили ИС	7	2	7	6			2, 5	14	Устный опрос
8	8. Фреймворки и паттерны проектирования ИС	8	2					6	10	Устный опрос
	Промежуточная аттестация									Зачет
	Всего		18		36				54	

4.2 Краткое содержание разделов и тем занятий

Семестр № 4

№	Тема	Краткое содержание
1	1. Понятие архитектуры ИС	Определение архитектуры информационных систем как набора принципов организации бизнес-, данных-, прикладных и технологических доменов; уровни представления (концептуальный, логический, физический); рамки Zachman и TOGAF; специфика ИС недропользования (ГИС, обработка геоданных, сейсмология); требования к платформам (масштабируемость, надежность для больших объемов данных).
2	2. Функциональное моделирование ИС	Методы функционального моделирования (IDEF0, DFD); бизнес-процессы в недропользовании (разведка, моделирование пластов, отчетность); декомпозиция процессов; интеграция с аппаратными платформами; выбор платформ для поддержки workflow (BPMN-инструменты).
3	3. Клиент-серверная архитектура	Двух- и трехуровневые модели (thin/fat client); протоколы (HTTP, REST, gRPC); балансировка нагрузки; аппаратные платформы (серверы, виртуализация); применение в веб-ГИС для просмотра геоданных (Leaflet/OpenLayers).
4	4. Распределенные ИС и SOA	Сервис-ориентированная архитектура (SOA), микросервисы; контейнеризация (Docker); оркестрация (Kubernetes); облачные платформы (AWS, Azure для геоданных); отказоустойчивость и масштабируемость для распределенной обработки сейсмических данных.
5	5. Архитектуры web-приложений	Паттерны MVC, MVP, MVVM; frontend-фреймворки (React, Vue.js); backend (Node.js, Django); full-stack платформы для ГИС (Next.js + PostgreSQL/PostGIS); аппаратно-программные требования к web-кластерам недропользования.
6	6. СУБД и хранилища данных	Реляционные СУБД (PostgreSQL/PostGIS), NoSQL (MongoDB, Cassandra); пространственные базы (GeoServer); big data платформы (Hadoop, Spark для геоданных); выбор аппаратных платформ (SSD RAID для высоконагруженных ГИС).
7	7. Архитектурные стили ИС	Стили (event-driven, pipe-filter, layered); RESTful API; message queues (RabbitMQ, Kafka для потоковых геоданных); интеграция стилей с аппаратными платформами; оценка для реального времени в мониторинге недр.
8	8. Фреймворки и паттерны проектирования ИС	Enterprise Architecture Frameworks (Zachman, FEAF); паттерны (Repository, Factory, Observer); DevOps-практики (CI/CD для ГИС); комплексная оценка платформ (TCO, ROI); кейсы выбора стеков для ИС недропользования.

4.3 Перечень лабораторных работ

Семестр № 4

№	Наименование лабораторной работы	Кол-во академических часов
1	Лабораторная работа №1: Анализ рамки Zachman на примере ГИС для недропользования в Draw.io (диаграммы по ячейкам Who/What/How для геологоразведки).	4
2	Лабораторная работа №2: Построение IDEF0-диаграммы процесса сейсмообработки в Draw.io, выбор платформы (on-premise vs SaaS) для каждого блока.	5
3	Лабораторная работа №3: Развёртывание простого клиент-серверного приложения (Flask + HTML/JS) для отображения геоданных, тестирование на localhost с замерами latency.	5
4	Лабораторная работа №4: Создание Docker-контейнера с микросервисом геообработки (Python Flask API), тестирование масштабирования с docker-compose.	5
5	Лабораторная работа №5: Разработка MVC web-приложения для визуализации скважинных данных (React frontend + Express backend), оценка производительности на разных платформах.	5
6	Лабораторная работа №6: Создание базы геоданных в PostgreSQL/PostGIS, импорт shapefile скважин, SQL-запросы с пространственными индексами, сравнение с MongoDB.	6
7	Лабораторная работа №7: Реализация pipe-filter архитектуры для обработки потока сейсмических данных в Python (pandas pipeline), замер throughput на разных конфигурациях.	6

4.4 Перечень практических занятий

Практических занятий не предусмотрено

4.5 Самостоятельная работа

Семестр № 4

№	Вид СРС	Кол-во академических часов
1	Оформление отчетов по лабораторным и практическим работам	10
2	Подготовка к зачёту	9
3	Подготовка к практическим занятиям	5
4	Подготовка к практическим занятиям (лабораторным работам)	5
5	Подготовка к сдаче и защите отчетов	15
6	Проработка разделов теоретического материала	10

В ходе проведения занятий по дисциплине используются следующие интерактивные методы обучения: работа в малых группах

5 Перечень учебно-методического обеспечения дисциплины

5.1 Методические указания для обучающихся по освоению дисциплины

5.1.1 Методические указания для обучающихся по лабораторным работам:

Лабораторная работа №1. Анализ рамки Zachman для ГИС

Цель работы: Освоить рамку Zachman для анализа архитектуры ИС недропользования и оценки платформ по доменам.

Теоретический справочник:

Zachman Framework — матрица 6x6 для описания ИС: строки (контексты: Scope, Business, System, Technology, Detailed, Functioning); столбцы (перспективы: What=данные, How=функции, Where=сети, Who=люди, When=время, Why=мотивация). ГИС — геоинформационная система для пространственных данных. TOGAF — The Open Group Architecture Framework для enterprise-архитектуры.

Ход работы:

1. Установить Draw.io (или использовать онлайн).
2. Заполнить ячейки Zachman для ГИС недропользования (What=скважины/сейсма, How=обработка).
3. Для 3 ячеек оценить платформы (on-premise PostgreSQL vs AWS RDS).
4. Экспортировать диаграмму PDF.

Ожидаемый результат: Диаграмма Zachman с оценкой платформ, таблица сравнения (стоимость, масштабируемость).

Контрольные вопросы:

1. Что описывает столбец "What" в Zachman?
2. Разница контекстов Scope и Detailed?
3. Когда применять TOGAF vs Zachman?

Лабораторная работа №2. Моделирование бизнес-процессов IDEF0

Цель работы: Построить функциональную модель геологоразведки и обосновать выбор платформ для процессов.

Теоретический справочник:

IDEF0 — методология функционального моделирования: блоки (A0-A3), стрелки (входы/выходы/управление/механизмы). BPMN — Business Process Model and Notation (диаграммы процессов). SaaS — Software as a Service (облачные сервисы по подписке).

Ход работы:

1. Создать контекст-диаграмму A0 (сейсмообработка).
2. Декомпозировать A0 → A1-A3 (прием данных, обработка, анализ).
3. Для каждого блока выбрать платформу (локальный сервер/облако).
4. Обосновать выбор по критериям (TCO, latency).

Ожидаемый результат: IDEF0-диаграммы, таблица выбора платформ с обоснованием.

Контрольные вопросы:

1. Роль стрелок в IDEF0?
2. Разница IDEF0 и DFD?
3. Когда предпочтеть SaaS локальному серверу?

Лабораторная работа №3. Клиент-серверное приложение для геоданных

Цель работы: Развернуть 3-tier клиент-серверную архитектуру и оценить производительность платформ.

Теоретический справочник:

Клиент-сервер — модель: client (браузер), application server (логика), database server. REST

— Representational State Transfer (HTTP API). Latency — задержка ответа.

Ход работы:

1. Установить Flask, создать API /geodata.
2. Frontend HTML+JS (fetch API) для карты.
3. Тестировать на localhost, измерить response time.
4. Сравнить с Node.js версией.

Ожидаемый результат: Рабочий стек, таблица latency (Flask vs Node.js).

Контрольные вопросы:

1. Преимущества 3-tier над 2-tier?
2. Что такое RESTful API?
3. Как измерить latency?

Лабораторная работа №4. Docker-контейнер с микросервисом геообработки

Цель работы: Освоить контейнеризацию SOA-компонентов и оценить масштабируемость платформ для распределенных ИС.

Теоретический справочник:

Docker — платформа контейнеризации, изолирует приложения с зависимостями.

Микросервис — независимый сервис в SOA (Service-Oriented Architecture). Kubernetes — оркестратор контейнеров для масштабирования. Docker Compose — инструмент для multi-container приложений.

Ход работы:

1. Создать Flask API микросервис (/process-seismic).
2. Написать Dockerfile (FROM python:3.9, COPY, pip install).
3. Создать docker-compose.yml (сервис + PostgreSQL).
4. Запустить docker-compose up, протестировать API.
5. Масштабировать: docker-compose scale service=3, измерить throughput.

Ожидаемый результат: Docker-образ, compose-файл, графики throughput при масштабировании (1 → 3 контейнера).

Контрольные вопросы:

1. В чем разница контейнера и виртуальной машины?
2. Зачем нужна оркестрация в Kubernetes?
3. Как Docker обеспечивает portability платформ?

Лабораторная работа №5. MVC web-приложение для скважинных данных

Цель работы: Разработать full-stack MVC-приложение и сравнить производительность веб-платформ для ГИС.

Теоретический справочник:

MVC (Model-View-Controller) — паттерн: Model (данные), View (UI), Controller (логика).

React — JavaScript библиотека для UI-компонентов. Express.js — Node.js фреймворк для backend API. PostGIS — пространственное расширение PostgreSQL.

Ход работы:

1. Backend: Express.js API (/wells, /map-data).
2. Frontend: React app с Leaflet картой скважин.
3. База: PostgreSQL + shapefile импорт.
4. Замерить rendering time (performance.now()).
5. Сравнить с Django+Vue.js аналогом.

Ожидаемый результат: Деплоинуемое MVC-приложение, таблица производительности (React vs Vue).

Контрольные вопросы:

1. Роли компонентов в MVC?
2. Зачем нужен state management в React?
3. Сравнение монолита и микросервисов для ГИС?

Лабораторная работа №6. СУБД для геоданных (PostGIS vs MongoDB)

Цель работы: Сравнить реляционные и NoSQL платформы для хранения пространственных геоданных.

Теоретический справочник:

PostGIS — PostgreSQL extension для GIS (точки, линии, полигоны, ST_Distance). NoSQL — нереляционные БД (MongoDB document-oriented). Shapefile — формат векторных данных ESRI. Spatial Index — R-tree индекс для пространственных запросов.

Ход работы:

1. Установить PostgreSQL+PostGIS, MongoDB.
2. Импортировать shapefile скважин (ogr2ogr / mongoimport).
3. SQL-запросы: расстояния, bounding box (ST_DWithin).
4. MongoDB: GeoJSON queries (\$near, \$geoWithin).
5. Замерить query time на 10K+ записей.

Ожидаемый результат: Базы данных, таблица query performance (PostGIS быстрее spatial joins).

Контрольные вопросы:

1. Когда выбрать PostGIS, а когда MongoDB для геоданных?
2. Что такое spatial index?
3. Преимущества ACID в PostgreSQL?

Лабораторная работа №7. Pipe-filter архитектура для потоковых геоданных

Цель работы: Реализовать архитектурный стиль pipe-filter для обработки реального времени сейсмических данных.

Теоретический справочник:

Pipe-filter — архитектурный стиль: фильтры (обработчики) соединены pipes (очередями). Kafka — распределенная message queue для streaming. Pandas pipeline — chain операций обработки данных. Throughput — пропускная способность (объем/время).

Ход работы:

1. Создать pipeline: raw_data → filter_noise → fft → detect_events.
2. Использовать multiprocessing.Queue как pipes.
3. Генерировать поток сейсмических данных (numpy noise + signal).
4. Замерить end-to-end latency и throughput.
5. Тестировать на разных hardware (CPU cores).

Ожидаемый результат: Рабочий pipeline, графики throughput vs latency, анализ bottlenecks.

Контрольные вопросы:

1. Преимущества pipe-filter перед монолитом?
2. Как Kafka обеспечивает fault tolerance?
3. Что ограничивает throughput в streaming ИС?

5.1.2 Методические указания для обучающихся по самостоятельной работе:

Рекомендации по самостоятельной работе:

1. Рекомендации по самостоятельной подготовке к лабораторным работам

- Изучите теоретический материал по теме лабораторной работы.

Ознакомьтесь с учебниками, лекциями и дополнительными источниками, чтобы понимать цели и задачи работы, основные понятия и методы, используемые в лабораторном задании1.

- Внимательно ознакомьтесь с методическими указаниями и требованиями к лабораторной работе. Обратите внимание на последовательность выполнения этапов, используемое программное обеспечение, форматы исходных и выходных данных, требования к визуализации и анализу результатов.

- Подготовьте исходные данные. Проверьте наличие всех необходимых файлов, убедитесь в их корректности (форматы, структура, отсутствие ошибок и пропусков

данных).

- Освойте необходимые функции и инструменты программного обеспечения. Повторите работу с теми модулями и инструментами, которые будут использоваться в лабораторной работе.

• Планируйте время. Разделите выполнение работы на этапы: подготовка данных, выполнение анализа, оформление визуализации, написание отчета.

2. Рекомендации по оформлению отчетов по лабораторным работам

- Структурируйте отчет по стандартной схеме:
- Титульный лист (название работы, ФИО, группа, дата)
- Цель работы
- Краткое описание исходных данных
- Описание используемых методов и программного обеспечения
- Последовательное изложение этапов работы с иллюстрациями (скриншотами, графиками, картами)
- Анализ полученных результатов (выявленные особенности, сравнение с теорией, интерпретация)
- Выводы и рекомендации
- Список использованных источников
- Используйте качественные иллюстрации. Все графические материалы должны быть четкими, снабжены подписями, масштабами, легендами и пояснениями.
- Формулируйте выводы по существу. Кратко и ясно отражайте основные результаты работы, выявленные закономерности, достоинства и ограничения применяемых методов.
- Оформляйте отчет в соответствии с требованиями ДОТ. Соблюдайте стандарты оформления текста, таблиц, рисунков и ссылок на источники.

3. Рекомендации по самостоятельной проработке отдельных разделов тем

- Изучайте рекомендованную литературу и дополнительные источники. Используйте учебники, статьи, электронные ресурсы, профессиональные базы данных и справочные материалы, указанные в рабочей программе дисциплины1.
- Выполняйте конспектирование ключевых понятий и алгоритмов. Составляйте краткие записи по основным определениям, алгоритмам, этапам работы с ПО, особенностям визуализации и анализа данных.
- Практикуйтесь в самостоятельном выполнении типовых заданий. Решайте задачи, связанные с обработкой и визуализацией геолого-геофизических данных, используя различные программные средства.
- Формулируйте вопросы и уточнения для обсуждения на занятиях. Записывайте непонятные моменты, чтобы получить разъяснения у преподавателя или в ходе дискуссии.
- Анализируйте примеры из практики. Изучайте реальные кейсы решения задач геофизики, сравнивайте разные подходы и делайте выводы о целесообразности их применения.

4. Общие рекомендации

- Развивайте навыки поиска и критического анализа информации. Пользуйтесь современными информационными ресурсами, анализируйте достоверность и актуальность найденных данных.
- Акцентируйте внимание на интеграции знаний и умений. Страйтесь связывать теоретические знания с практическими задачами, анализируйте, как выбранные методы и технологии влияют на качество и достоверность графического представления информации.
- Соблюдайте академическую честность. Все результаты, представленные в отчетах, должны быть получены самостоятельно, с обязательным указанием источников

затемствованных данных и иллюстраций.

6 Фонд оценочных средств для контроля текущей успеваемости и проведения промежуточной аттестации по дисциплине

6.1 Оценочные средства для проведения текущего контроля

6.1.1 семестр 4 | Устный опрос

Описание процедуры.

Опрос может проводиться:

Фронтально — в форме беседы с группой, когда вопросы задаются всей группе, а ответы даются по очереди или по желанию.

Индивидуально — каждый студент отвечает на один или несколько вопросов, давая развернутый, связный ответ, часто с примерами и пояснениями.

Комбинированно — сочетаются оба подхода, а также используются дополнительные методы (например, письменные карточки, рецензирование ответов товарищей)

Критерии оценивания.

полнота и правильность ответа;
понимание и осознанность материала;
логичность и последовательность изложения;
корректность терминологии;
способность отвечать на уточняющие вопросы

6.2 Оценочные средства для проведения промежуточной аттестации

6.2.1 Критерии и средства (методы) оценивания индикаторов достижения компетенции в рамках промежуточной аттестации

Индикатор достижения компетенции	Критерии оценивания	Средства (методы) оценивания промежуточной аттестации
ОПК ОС-7.2	полнота и правильность ответа; понимание и осознанность материала; логичность и последовательность изложения; корректность терминологии; способность отвечать на уточняющие вопросы	устный опрос

6.2.2 Типовые оценочные средства промежуточной аттестации

6.2.2.1 Семестр 4, Типовые оценочные средства для проведения зачета по дисциплине

6.2.2.1.1 Описание процедуры

К зачету допускаются студенты сдавшие все отчеты по лабораторным (практическим) работам. Зачёт проводится в форме устного опроса или тестирования, включающего 5 вопросов — по одному из каждой основной темы курса. В некоторых случаях допускается комбинированная форма: тест + устный опрос.

Время на ответ ограничено, ответы должны быть чёткими, логичными и аргументированными.

В случае неудовлетворительного результата студенту предоставляется возможность пересдачи в установленные сроки. При повторном не сдаче возможна дополнительная консультация и индивидуальное собеседование. Оценка выставляется по шкале с учётом полноты и правильности ответов.

Примерные вопросы для подготовки к зачету:

Что такое архитектура ИС? Опишите домены (бизнес, данные, приложения, технологии) с примерами для ГИС недропользования.

В чём разница уровней представления архитектуры ИС: концептуальный, логический, физический?

Опишите рамку Zachman: строки и столбцы, пример заполнения для геологоразведки.

Что моделирует IDEF0? Перечислите типы стрелок (входы, выходы, управление, механизмы).

Разница IDEF0 и DFD в функциональном моделировании бизнес-процессов недропользования?

Опишите двух- и трехуровневые клиент-серверные архитектуры для веб-ГИС.

Что такое RESTful API? Протоколы HTTP, gRPC для обмена геоданными.

Принципы SOA и микросервисов? Преимущества перед монолитами для сейсмообработки.

Что такое Docker? Разница контейнера и виртуальной машины в платформах ИС.

Опишите Kubernetes: поды, сервисы, deployment для масштабирования ГИС.

Паттерны MVC, MVP, MVVM? Роли компонентов в web-приложениях для скважинных данных.

Сравните frontend-фреймворки React и Vue.js по производительности для ГИС.

Что такое PostGIS? Пространственные функции ST_Distance, ST_Intersects для геоданных.

Разница реляционных СУБД (PostgreSQL) и NoSQL (MongoDB) для пространственных данных?

Что такое spatial index (R-tree)? Преимущества для запросов по расстоянию в ГИС.

Архитектурные стили: pipe-filter, event-driven, layered. Примеры для потоковых геоданных.

Что такое message queues (Kafka, RabbitMQ)? Применение в реальном времени мониторинга недр.

Опишите фреймворки enterprise-архитектуры: Zachman, TOGAF, FEAF.

Что такое паттерны Repository, Factory, Observer в проектировании ИС?

DevOps-практики CI/CD для ГИС: инструменты Jenkins, GitHub Actions.

Методики выбора платформ: TCO (Total Cost of Ownership), ROI, benchmarking.

Сравните on-premise и облачные платформы (AWS RDS vs локальный PostgreSQL) по стоимости для геоданных.

Как заполнить ячейки Zachman для ГИС в лабораторной №1? Критерии выбора платформ. Постройте IDEF0-диаграмму A0-A3 для сейсмообработки (лабораторная №2).

Замерьте latency клиент-серверного Flask-приложения для геоданных (лабораторная №3).

Создайте Dockerfile для микросервиса геообработки, протестируйте docker-compose scale (лабораторная №4).

Реализуйте MVC в React+Express для визуализации скважин, сравните производительность (лабораторная №5).

Сравните query time PostGIS vs MongoDB для 10K shapefile скважин (лабораторная №6). Реализуйте pipe-filter pipeline для сейсмических данных, замерьте throughput (лабораторная №7). Обоснуйте выбор платформы (workstation vs cloud кластер) для ИС недропользования по ТСО.

6.2.2.1.2 Критерии оценивания

Зачтено	Не зачтено
выставляется студенту, твердо знающему программный материал, грамотно и по существу его излагающему, который не допускает существенных неточностей в ответе на вопросы, правильно применяет теоретические положения при решении практических задач;	выставляется студенту, который не знает значительной части программного материала, допускает существенные ошибки, не может ответить на дополнительные вопросы

7 Основная учебная литература

1. Архитектура информационных систем : учебник для вузов по направлению подготовки 230400 "Информационные системы и технологии" / Б. Я. Советов, А. И. Водяхо, В. А. Дубенецкий, В. В. Цехановский, 2012. - 283.
2. Демина, Л. С. Архитектура информационных систем : учебное пособие / Л. С. Демина, Д. А. Зубок, А. В. Маятин. — Санкт-Петербург : НИУ ИТМО, 2024. — 74 с. — Текст : электронный // Лань : электронно-библиотечная система.
3. Иванова, С. М. Архитектура информационных систем. Способы моделирования : учебное пособие / С. М. Иванова, З. В. Ильинская. — Москва : РТУ МИРЭА, 2024. — 83 с. — ISBN 978-5-7339-2153-2. — Текст : электронный // Лань : электронно-библиотечная система.

8 Дополнительная учебная литература и справочная

1. Аникеев, Д. В. Архитектура информационных систем : учебное пособие / Д. В. Аникеев. — Рязань : РГРТУ, 2022. — 72 с. — Текст : электронный // Лань : электронно-библиотечная система.

9 Ресурсы сети Интернет

1. <http://library.istu.edu/>
2. <https://e.lanbook.com/>

10 Профессиональные базы данных

1. <http://new.fips.ru/>
2. <http://www1.fips.ru/>

11 Перечень информационных технологий, лицензионных и свободно распространяемых специализированных программных средств, информационных справочных систем

1. Лицензионное программное обеспечение Системное программное обеспечение
2. Лицензионное программное обеспечение Пакет прикладных офисных программ
3. Лицензионное программное обеспечение Интернет-браузер

12 Материально-техническое обеспечение дисциплины

1. Учебная аудитория для проведения лекционных занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации. Оснащение: комплект учебной мебели, рабочее место преподавателя, доска. Мультимедийное оборудование (в том числе переносное): мультимедийный проектор, экран, акустическая система, компьютер с выходом в интернет.
2. Учебная аудитория для проведения лабораторных/практических (семинарских) занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации. Оснащение: комплект учебной мебели, рабочее место преподавателя, доска. Мультимедийное оборудование (в том числе переносное): мультимедийный проектор, экран, акустическая система, компьютер с выходом в интернет.