

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Фурсов Владимир Александрович

Должность: И.о. директора Пятигорского института (филиал) Северо-Кавказского федерального университета

Дата подписания: 2025.02.11 10:44

Уникальный программный ключ:

1c378726a41fd0143ae5bccc8ba81860b00daa62

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**
**Федеральное государственное автономное образовательное учреждение
высшего образования**
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»
Пятигорский институт (филиал) СКФУ
Колледж Пятигорского института (филиал) СКФУ

**МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ САМОСТОЯТЕЛЬНОЙ
РАБОТЫ**

МДК.02.02 Осуществление интеграции программных модулей

Специальность СПО

09.02.11 Разработка и управление программным обеспечением

Пятигорск 2026

Методические указания для самостоятельной работы по дисциплине «Осуществление интеграции программных модулей» составлены в соответствии с требованиями ФГОС СПО к подготовке выпуска для получения квалификации специалист по информационным системам. Предназначены для студентов, обучающихся по специальности 09.02.11 Разработка и управление программным обеспечением.

СОДЕРЖАНИЕ:

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА	4
ПЛАН-ГРАФИК ВЫПОЛНЕНИЯ СРС	6
МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ К СРС	6

Пояснительная записка

Методические указания предназначены для студентов групп СПО специальности 09.02.11 Разработка и управление программным обеспечением.

Самостоятельная работа студентов проводится с целью:

систематизации и закрепления полученных теоретических знаний и практических умений обучающихся;

углубления и расширения теоретических знаний;

формирования умений использовать нормативную, правовую, справочную документацию и специальную литературу;

развития познавательных способностей и активности обучающихся, творческой инициативы, ответственности и организованности;

формирования самостоятельности мышления, способностей к саморазвитию, самосовершенствованию и самореализации;

развития исследовательских умений.

В результате освоения учебной дисциплины обучающийся должен

уметь:

- Распознавать задачу и/или проблему в профессиональном и/или социальном контексте; анализировать и выделять её составные части;
- Определять задачи для поиска информации, необходимые источники, планировать процесс поиска, структурировать и оформлять результаты;
- Анализировать архитектуру приложения и определять точки интеграции;
- Проектировать интеграционное взаимодействие с использованием диаграмм последовательности и компонентов;
- Реализовывать синхронный и асинхронный обмен данными между модулями;
- Разрабатывать REST API и SOAP-сервисы, документировать API;
- Настраивать брокеры сообщений (RabbitMQ, Apache Kafka) и реализовывать producer-consumer;
- Применять методы аутентификации и авторизации (JWT, OAuth2, API-ключи) в интеграционных решениях;
- Контейнеризовать приложения с помощью Docker и запускать многоконтейнерные приложения;
- Развертывать функции в Serverless-средах;
- Настраивать пайплайны CI/CD и управлять версиями API;
- Настраивать сбор логов, метрик и трассировку запросов в распределённых системах.

знать:

- Актуальный профессиональный и социальный контекст, основные источники информации и ресурсы для решения задач;
- Номенклатуру информационных источников, приемы структурирования информации, современные средства информатизации;
- Понятие, цели, задачи и виды интеграции программных модулей;

- Архитектурные стили интеграции (клиент-серверная, многоуровневая, SOA, микросервисная);
- Шаблоны интеграции корпоративных приложений (EIP);
- Протоколы передачи данных (HTTP/HTTPS, JMS, AMQP, MQTT) и форматы обмена (JSON, XML, Protobuf);
- Принципы синхронного и асинхронного взаимодействия;
- Структуру SOAP-сообщения, WSDL, UDDI, сравнение REST и SOAP;
- Принципы работы брокеров сообщений (RabbitMQ, Apache Kafka);
- Методы обеспечения безопасности при интеграции (OAuth2, JWT, API-ключи, TLS/SSL, Rate limiting, CORS);
- Принципы контейнеризации (Docker) и оркестрации (Kubernetes);
- Основы облачных интеграций и Serverless (FaaS, триггеры);
- Стратегии управления версиями API и семантическое версионирование;
- Инструменты мониторинга и отладки распределённых систем (ELK, Jaeger, Prometheus, Grafana);
- Отраслевую нормативную документацию, современный отечественный и зарубежный опыт в области интеграции программных модулей.

План-график выполнения СРС

№	Наименование разделов и тем дисциплины, их краткое содержание; вид самостоятельной работы	Форма контроля	Зачетные единицы (часы)
V семестр			
1.	Тема 2.5. Интеграция на основе веб-сервисов SOAP Изучение структуры SOAP-сообщения, WSDL, UDDI. Сравнительный анализ REST и SOAP: области применения, преимущества и недостатки	Собеседование	8
2.	Тема 2.6. Интеграция с использованием брокеров сообщений Анализ принципов работы брокеров сообщений (очереди, топики). Сравнение RabbitMQ и Apache Kafka: сходства, различия, сценарии использования	Собеседование	8
3.	Тема 2.7. Безопасность при интеграции Изучение методов аутентификации и авторизации (OAuth2, JWT, API-ключи). Анализ способов защиты API (Rate limiting, CORS, валидация входных данных). Принципы шифрования TLS/SSL	Собеседование	8
Итого за V семестр			24
VI семестр			
4.	Тема 2.8. Микросервисная архитектура и контейнеризация Изучение принципов микросервисной архитектуры (декомпозиция, независимость, отказоустойчивость). Анализ контейнеризации с Docker: Dockerfile, образы, контейнеры. Обзор оркестрации (Kubernetes, Docker Swarm)	Собеседование	6
5.	Тема 2.9. Интеграция в облачных средах и Serverless Изучение облачных платформ (AWS, Azure, Yandex Cloud) и их интеграционных сервисов (API Gateway, очереди, функции). Анализ бессерверных вычислений (FaaS, триггеры)	Собеседование	6
6.	Тема 2.10. Управление зависимостями и версионирование при интеграции Изучение стратегий версионирования API (URI, заголовки, медиатипы). Анализ инструментов управления зависимостями (Maven, Gradle, npm, pip). Принципы семантического версионирования и CI/CD	Собеседование	6
7.	Тема 2.11. Мониторинг и отладка интеграционных решений Изучение систем логирования и трассировки (ELK, Jaeger). Анализ метрик и мониторинга (Prometheus, Grafana)	Собеседование	6
Итого за VI семестр			24
Всего за V-VI семестры			48

Методические рекомендации к СРС

1. Методические рекомендации по проведению собеседования.

Собеседование - наиболее распространенный метод контроля знаний учащихся, вариант текущей проверки, процессе которого преподаватель получает

широкие возможности для изучения индивидуальных возможностей усвоения учащимися учебного материала.

При подготовке к собеседованию студент должен:

1. Предварительно повторить теоретический материал темы (тем) по которой проводится устный опрос.
2. Ознакомиться с заданием, уяснить его фабулу и поставленные вопросы.
3. Продумать логику и последовательность изложения материала. Ответы на поставленные вопросы должны быть аргументированными.

1. Критерии оценивания компетенций

1. Оценка «ОТЛИЧНО» выставляется студенту, глубоко и прочно усвоившему программный, в том числе лекционный материал, последовательно, четко и самостоятельно (без наводящих вопросов) отвечающему на вопрос.

2. Оценка «ХОРОШО» выставляется студенту, твердо знающему программный, в том числе лекционный материал, грамотно и по существу отвечающему на вопрос и не допускающему при этом существенных неточностей (неточностей, которые не могут быть исправлены наводящими вопросами или не имеют важного практического значения). То же относится к освещению практически важных вопросов

3. Оценка «УДОВЛЕТВОРИТЕЛЬНО» выставляется студенту, который обнаруживает знание основного материала, но не знает его деталей, допускает неточности, недостаточно правильные формулировки, излагает материал с нарушением последовательности, отвечает на практически важные вопросы с помощью или поправками преподавателя.

4. Оценка «НЕУДОВЛЕТВОРИТЕЛЬНО» выставляется студенту, который не знает значительной части программного, в том числе лекционного материала.

Список рекомендуемой литературы

3.2.1. Основные печатные издания

1. Ньюмен, С. Создание микросервисов / С. Ньюмен. — Санкт-Петербург : Питер, 2022. — 400 с. — ISBN 978-5-4461-1840-5.
2. Ричардсон, К. Микросервисы. Паттерны разработки и рефакторинга / К. Ричардсон. — Санкт-Петербург : Питер, 2021. — 544 с. — ISBN 978-5-4461-1687-6.
3. Холп, Э. Шаблоны интеграции корпоративных приложений / Э. Холп, Б. Вулф, Б. Д'Соуза. — Москва : Вильямс, 2020. — 736 с. — ISBN 978-5-8459-2008-3.
4. Санчес, Р. Д. RabbitMQ в действии / Р. Д. Санчес. — Москва : ДМК Пресс, 2021. — 280 с. — ISBN 978-5-97060-914-5.
5. Шапиро, М. Kafka в действии / М. Шапиро. — Москва : ДМК Пресс, 2022. — 344 с. — ISBN 978-5-97060-970-1.
6. Любич, М. Проектирование API: эффективные практики REST, GraphQL и gRPC / М. Любич. — Санкт-Петербург : Питер, 2023. — 304 с. — ISBN 978-5-4461-2232-7.
7. Мартин, Р. Чистая архитектура. Искусство разработки программного обеспечения / Р. Мартин. — Санкт-Петербург : Питер, 2021. — 352 с. — ISBN 978-5-4461-1260-1.
8. Фаулер, М. Шаблоны корпоративных приложений / М. Фаулер. — Москва : Вильямс, 2021. — 544 с. — ISBN 978-5-8459-1906-3.
9. Бернштейн, П. А. Основы промежуточного программного обеспечения / П. А. Бернштейн. — Москва : Национальный Открытый Университет «ИНТУИТ», 2022. — 450 с.
10. Тузовский, А. Ф. Проектирование и разработка распределенных информационных систем : учебное пособие / А. Ф. Тузовский. — Томск : Изд-во ТПУ, 2020. — 145 с.