

Аннотация к рабочей программы дисциплины Б1.О.33 Микросервисная архитектура

Объем трудоемкости: 2 зачетные единицы

Цель дисциплины: освоение ключевых архитектурных стилей и паттернов микросервисов, получение практических навыков проектирования и развертывания распределенных систем, а также применение полученных знаний для внедрения микросервисных решений в реальных проектах. Это позволит будущим специалистам оптимизировать программную архитектуру, проводить организационные трансформации, принимать стратегические решения и эффективно управлять командами разработки

Задачи дисциплины:

- Изучение принципов микросервисной архитектуры и ее преимуществ перед монолитной.
- Приобретение практического опыта проектирования, разработки и тестирования микросервисов.
- Освоение инструментов и технологий для создания, развертывания и оркестрации микросервисов (Docker, Kubernetes, API Gateway, Service Mesh).
- Формирование навыков командной работы в условиях микросервисной архитектуры.
- Анализ проблем масштабирования и управления микросервисными системами, изучение способов их решения.
- Изучение методов мониторинга, логирования и отладки распределенных систем (ELK-стек, Prometheus, Grafana).
- Применение микросервисного подхода в реальных проектах для повышения их масштабируемости, гибкости и надежности.

Место дисциплины в структуре образовательной программы

Дисциплина «Микросервисная архитектура» относится к обязательной части Блока 1 «Дисциплины (модули)» учебного плана.

Для успешного освоения дисциплины необходимы знания, умения и навыки, полученные при изучении предшествующих дисциплин, таких как «Основы программирования», «Методы программирования», «Объектно-ориентированное программирование», «Компьютерные сети», «Базы данных», «Разработка пользовательского WEB интерфейса», «Системное программное обеспечение GNU/Linux», «Аппаратно-программные средства WEB».

Знания и навыки, полученные в ходе изучения дисциплины, являются основой для последующего изучения таких дисциплин, как «Высоконагруженные приложения», «Основы облачных технологий», «Технологии тестирования программного обеспечения», «Коллективная разработка приложений», а также для выполнения выпускной квалификационной работы.

Требования к уровню освоения дисциплины

Изучение данной учебной дисциплины направлено на формирование у обучающихся следующих компетенций:

Код и наименование индикатора*	Результаты обучения по дисциплине (<i>знает, умеет, владеет (навыки и/или опыт деятельности)</i>)
ОПК-3. Способен применять современные информационные технологии, в том числе отечественные, при создании программных продуктов и программных комплексов различного назначения.	
ИД-1.ОПК-3. Аргументировано применяет современные информационные технологии, в том числе отечественные, при создании программных продуктов и программных комплексов различного назначения.	Знать: современные и перспективные инструменты разработки ПО, включая отечественные платформы и фреймворки для микросервисов; методологии и технологии разработки ПО, включая принципы декомпозиции монолитных систем; принципы и инструменты проектирования ПО, включая шаблоны проектирования микросервисов (Service Discovery, Circuit Breaker, Saga); методы проектирования программных интерфейсов (REST/gRPC API, event-driven взаимодействие).
	Уметь: анализировать и обосновывать выбор технологических решений, включая оценку целесообразности перехода на микросервисную архитектуру.
	Владеть: навыками проектирования программных интерфейсов для взаимодействия микросервисов; методами сбора, обработки, анализа и обобщения данных исследований в распределенных системах; подходами к решению аналитических задач, включая проблемы масштабируемости, отказоустойчивости и мониторинга микросервисных систем.
ИД-2.ОПК-3. Ориентируется в современных положениях и концепциях прикладного и системного программного обеспечения, архитектуры компьютеров и сетей (в том числе и глобальных), технологии создания и сопровождения программных продуктов и программных комплексов.	Знать: современные и перспективные средства разработки ПО, включая инструменты оркестрации микросервисов (Kubernetes, Nomad, OpenShift); методологии и технологии программирования, включая подходы к разработке и тестированию микросервисов (Contract Testing, Canary Releases); методы проектирования ПО, включая стратегии развертывания и управления жизненным циклом микросервисов; принципы проектирования программных интерфейсов, включая API-шлюзы и межсервисную асинхронную коммуникацию.
	Уметь: формулировать варианты реализации технических требований, включая выбор между монолитной и микросервисной архитектурой.
	Владеть: методами оценки трудоемкости и сроков реализации ПО с учетом особенностей распределенных систем; навыками согласования сроков выполнения задач в

Код и наименование индикатора*	Результаты обучения по дисциплине (<i>знает, умеет, владеет (навыки и/или опыт деятельности)</i>)
	условиях параллельной разработки множества микросервисов; техниками проектирования программных интерфейсов для обеспечения слабой связанности микросервисов; методами сбора, анализа и обобщения результатов исследований в контексте мониторинга и логирования микросервисных систем; подходами к решению аналитических задач, включая оптимизацию производительности и управление распределенными транзакциями.

Содержание дисциплины:

Распределение видов учебной работы и их трудоемкости по разделам дисциплины.

№	Наименование разделов (тем)	Количество часов				
		Всего	Аудиторная работа			Внеаудиторная работа
			Л	ПЗ	ЛР	СРС
1	2	3	4	5	6	7
1.	Введение в микросервисную архитектуру	8	2		4	2
2.	Принципы построения микросервисов	8	2		4	2
3.	Коммуникация между микросервисами	8	2		4	2
4.	Управление конфигурацией и развертыванием микросервисов	8	2		4	2
5.	Мониторинг и логирование в микросервисной архитектуре	8	2		4	2
6.	Тестирование микросервисов	8	2		4	4
7.	Проблемы и решения в микросервисной архитектуре	10	2		4	4
8.	Реализация микросервисной архитектуры в конкретных проектах	9,8	2		4	3,8
ИТОГО по разделам дисциплины		69,8	16		32	21,8
Контроль самостоятельной работы (КСР)		2				
Промежуточная аттестация (ИКР)		0,2				
Подготовка к текущему контролю						
Общая трудоемкость по дисциплине		72				

Курсовые работы: не предусмотрены

Форма проведения аттестации по дисциплине: зачет

Автор: Харченко А.В., доцент, канд. пед наук