

УТВЕРЖДЕНА

Решением Ученого совета
АНО ВО «Центральный университет»
«24» июня 2025 г.
Протокол №2

**Рабочая программа дисциплины (модуля)
«SRE (Обеспечение надёжности приложений)»**

Направление подготовки: 02.03.01 Математика и компьютерные науки

Направленность (профиль) подготовки: Искусственный интеллект

Квалификация (степень) выпускника: бакалавр

Форма обучения: очная

Срок освоения программы: 4 года

Год набора: 2025

**Москва
2025**

Содержание

1. Краткая характеристика дисциплины (модуля)	3
2. Перечень планируемых результатов обучения	4
3. Тематический план	6
4. Содержание дисциплины (модуля)	6
5. Учебно-методическое обеспечение	7
6. Материально-техническое обеспечение	8
7. Методические и оценочные материалы	9

1. Краткая характеристика дисциплины (модуля)

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

Изучение дисциплины (модуля) «SRE (Обеспечение надёжности приложений)» позволяет обеспечить стабильную и бесперебойную работу приложений, минимизируя время простоя и быстро реагируя на инциденты. Это критически важно для поддержания высокого уровня пользовательского опыта и эффективности бизнес-процессов.

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 Искусственный интеллект и входит в вариативную часть Блока 1, формируемую участниками образовательных отношений.

Дисциплина (модуль) является выборной и доступна для изучения на 2, 3 или 4 курсе в 4, 5, 6, 7 или 8 семестре на выбор.

Цель изучения дисциплины (модуля): освоение методов и практик для повышения надежности, масштабируемости и эффективности эксплуатации программных систем.

Задачи изучения дисциплины (модуля):

— изучить фундаментальные принципы SRE, включая концепции надёжности, доступности, производительности и масштабируемости, а также их применение в реальных проектах;

— освоить практические навыки мониторинга, анализа инцидентов, автоматизации процессов и управления рисками для обеспечения бесперебойной работы приложений;

— разработать умения по интеграции SRE-практик в жизненный цикл разработки ПО, включая работу с инструментами для тестирования, деплоя и оптимизации систем

В результате освоения дисциплины (модуля) обучающийся должен:

знать:

- основные концепции и практики SRE, включая ключевые метрики;
- стандартные методы мониторинга приложений и инфраструктуры;
- различные стратегии и методы развёртывания приложений;
- принципы работы систем уведомлений;

уметь:

- устанавливать и настраивать инструменты мониторинга;
- разворачивать тестовые приложения и управлять их контейнеризацией;
- проектировать и настраивать метрики для приложений;
- настраивать систему визуализации метрик;

владеть:

— навыком проектирования стратегии балансировки нагрузки для повышения устойчивости сервисов;

— навыком настройки системы мониторинга для своевременного реагирования на инциденты.

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
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 реализации математических моделей в профессиональной деятельности
ОПК-6	Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения	ОПК-6.1.	Знает алгоритмы разработки, компьютерные программы, а также алгоритмы вычислительной математики в области искусственного интеллекта
		ОПК-6.2.	Умеет разрабатывать математические программные продукты и комплексы с использованием современных

			технологий программирования в области искусственного интеллекта
		ОПК-6.3.	Имеет практический опыт разработки интеллектуальных информационных систем для визуализации результатов исследований в области искусственного интеллекта
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

3. Тематический план

№п/п	Наименование раздела дисциплины (модуля)	Трудоемкость, академические часы				ТКУ (текущий контроль успеваемости)
		Очная форма				
		Контактная работа		Контроль	Самостоятельная работа	
Лекции	Семинары (практические занятия)					
1	Введение в SRE	7	7		32	Кейс
2	Современные инструменты SRE	7	7		32	Кейс
3	Мониторинг	7	7		33	Кейс, Коллоквиум
4	Проект	7	7		33	Проект
	<i>Зачет с оценкой</i>			4		
	<i>Итого:</i>	<i>28</i>	<i>28</i>	<i>4</i>	<i>130</i>	
	<i>Объем дисциплины (модуля) (в ак. ч.)</i>	<i>190</i>				
	<i>Объем дисциплины (модуля) (в зач. ед.)</i>	<i>5</i>				

4. Содержание дисциплины (модуля)

№п/п	Наименование раздела дисциплины (модуля)	Содержание дисциплины (модуля) по темам
1	Введение в SRE	Введение в SRE: роли, процессы, SLO/SLI/Error Budget. Kubernetes: базовые расширения и сетевой доступ (CRD, Ingress). Деплой тестового приложения в Kubernetes и организация on-call.
2	Современные инструменты SRE	Метрики: сбор и хранение с Prometheus. Логи: сбор и анализ с Kibana (ELK/OpenSearch). Обсервабилити Kubernetes: метрики, логи, события, базовые практики. Проектирование метрик приложения и SLI по best practices. Визуализация: Grafana, дашборды и панели SLO.
3	Мониторинг	Алертинг: правила, маршрутизация, Alertmanager, runbooks. Стратегии деплоя: rolling, blue/green, canary, прогрессивная выдача. Балансировка нагрузки и устойчивость (L4/L7, общие подходы). Troubleshooting: Kubernetes, приложения и Kafka; разбор инцидентов.
4	Проект	Проект: постановка, дизайн и план SRE. Проект: реализация и интеграция. Проект: испытания и защита.

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. Site Reliability Engineering. Надежность и безотказность как в Google : практическое руководство / Б. Бейер, К. Джоунс, Дж. Петофф, Н. Р. Мёрфи. - Санкт-Петербург : Питер, 2019. - 592 с. - (Серия «Бестселлеры O'Reilly»). - ISBN 978-5-4461-0976-0. - Текст : электронный. - URL: <https://znanium.ru/catalog/product/2139467>.

2. Site Reliability Workbook: практическое применение : научно-популярное издание / сост. Б. Бейер, Н. Р. Мёрфи, Д. Рензин [и др.]. - Санкт-Петербург : Питер, 2021. - 544 с. - (Серия «Бестселлеры O'Reilly»). - ISBN 978-5-4461-1087-2. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1733732>.

3. Розенталь, К. Хаос-инжиниринг. Революция в разработке устойчивых систем : практическое руководство / К. Розенталь, Н. Джонс ; пер. с англ. В. С. Яценкова. - Москва : ДМК Пресс, 2021. - 284 с. - ISBN 978-5-97060-796-1. - Текст : электронный. - URL: <https://znanium.com/catalog/product/2109480>.

4. Пивотто, Ж. Запускаем Prometheus. Мониторинг инфраструктуры и приложений : практическое руководство / Ж. Пивотто, Б. Бразил ; пер. с англ. А. Н. Киселева. – Москва : Books.kz, 2023. - 394 с. – ISBN 978-6-01810-341-4. - Текст : электронный. - URL: <https://znanium.ru/catalog/product/2205125>.

Дополнительная литература:

1. Сысолетин Е. Г. Разработка интернет-приложений : учебник для вузов / Е. Г. Сысолетин, С. Д. Ростунцев ; под научной редакцией Л. Г. Доросинского. — Москва : Издательство Юрайт, 2025. — 80 с. — (Высшее образование). — ISBN 978-5-534-17124-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/562916>.

2. Web-разработки в asp. Net web forms : учебник для вузов / С. Т. Гуляева, В. В. Миронов, Н. О. Котелина, И. И. Лавреш. — Москва : Издательство Юрайт, 2025. — 134 с. — (Высшее образование). — ISBN 978-5-534-19885-0. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/569218>.

3. Вехен, Д. Безопасный DevOps. Эффективная эксплуатация систем : практическое руководство / Д. Вехен. - Санкт-Петербург : Питер, 2020. - 432 с. - (Серия «Для профессионалов»). - ISBN 978-5-4461-1336-1. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1756107> 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>.

4. Тузовский А. Ф. Проектирование и разработка web-приложений : учебник для вузов / А. Ф. Тузовский. — Москва : Издательство Юрайт, 2025. — 219 с. — (Высшее образование). — ISBN 978-5-534-16300-1. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/561176>.

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;
- механическими калькуляторами;
- 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 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	https://elibrary.ru/defaultx.asp
2.	База данных для IT-специалистов	https://habr.com
3.	База данных ScienceDirect	https://www.sciencedirect.com
4.	Официальный сайт Министерства науки и высшего образования Российской Федерации	https://minobrnauki.gov.ru/
5.	Федеральный портал «Российское образование»	https://www.edu.ru/
6.	Информационная система "Единое окно доступа к образовательным ресурсам"	http://window.edu.ru/
7.	Единая коллекция цифровых образовательных ресурсов	http://school-collection.edu.ru/
8.	Федеральный центр информационно - образовательных ресурсов	http://fcior.edu.ru/

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
Операционные системы:		
Microsoft Imagine (Windows Client, Server)	зарубежное	лицензионное

Браузеры:		
Яндекс.Браузер	отечественное	свободно распространяемое
Google Chrome	зарубежное	свободно распространяемое
Офисные приложения:		
Microsoft Imagine (Visio, OneNote)	зарубежное	лицензионное
TeXstudio	зарубежное	свободно распространяемое
Adobe Acrobat Reader	зарубежное	свободно распространяемое
Программное обеспечение для планирования и учета времени:		
Toggle app	зарубежное	свободно распространяемое
Системы управления проектами:		
Microsoft Imagine (Project)	зарубежное	лицензионное
Системы управления базами данных:		
Microsoft Imagine (SQL Server)	зарубежное	лицензионное
Системы резервного копирования (backup):		
Acronis Backup Advanced for HyperV	зарубежное	лицензионное
Справочно-правовые системы:		
КонсультантПлюс: справочно-правовая система	отечественное	лицензионное
Средства антивирусной защиты:		
Kaspersky Endpoint Security для бизнеса Стандартный Russian Edition	отечественное	лицензионное
Среды разработки:		
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
Пакеты программных средств и библиотек:		
AutoPsy	зарубежное	свободно распространяемое
Interactive Disassembler (IDA)	зарубежное	свободно распространяемое
Системы управления библиографической информацией:		
Zotero	зарубежное	свободно распространяемое
Сервисы и службы:		
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) «SRE (Обеспечение надёжности приложений)» в рамках текущего контроля успеваемости используются такие виды учебной работы, как лекции, семинары, кеймы, коллоквиум, 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.

Кейс – практическая работа студентов над реальными или смоделированными задачами, что позволяет студенту применять теоретические знания на практике.

Студент самостоятельно разрабатывает стратегию решения поставленной задачи, что способствует развитию навыков критического мышления и самостоятельного принятия решений. Такой подход помогает подготовить будущих специалистов к реальным вызовам в их профессиональной деятельности.

Коллоквиум – устные ответы на вопросы, список которых известен студенту заранее.

В процессе подготовки к коллоквиуму необходимо проанализировать учебные материалы, ознакомившись с лекциями, учебниками и дополнительными источниками, акцентируя внимание на ключевых темах. Рекомендуется создать структурированные конспекты, выделяя основные идеи, термины и формулы.

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) «SRE (Обеспечение надёжности приложений)»

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	<p>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.</p>
9	Отлично	Зачтено	
8	Отлично	Зачтено	
7	Хорошо	Зачтено	<p>Студент обладает знаниями предмета почти в полном объеме рабочей программы и самостоятельно, логически последовательно и всесторонне отвечает на все вопросы, акцентируя внимание на 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.</p>
6	Хорошо	Зачтено	
5	Удовлетворительно	Зачтено	<p>Студент обладает базовыми знаниями по дисциплине (модулю), но испытывает трудности при самостоятельных ответах и использует неточные формулировки. В ходе ответов он допускает ошибки,</p>
4	Удовлетворительно	Зачтено	

Десятибалльная оценка	Пятибалльная оценка	Оценка за зачет	Общая характеристика результата обучения по дисциплине (модулю)
			касающиеся сути вопросов. Студент способен решать только самые простые задачи и владеет лишь минимальным набором методов исследования.
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) «SRE (Обеспечение надёжности приложений)» оценивается следующим образом:

Активность	Вес	Количество	Описание
Кейс	40%	4	Практическая работа студентов над реальными или смоделированными задачами, что позволяет студенту применять теоретические знания на практике
Коллоквиум	30%	1	Устные ответы на вопросы, список которых известен студенту заранее.
Проект	30%	1	Защита итогового проекта

Формула расчёта итоговой оценки по дисциплине (модулю) «SRE (Обеспечение надёжности приложений)»: « $0,4 \times \text{среднее за кейсы} + 0,3 \times \text{коллоквиум} + 0,3 \times \text{проект}$ ».

Текущий контроль успеваемости обучающихся по дисциплине (модулю)

Примерные задания кейсов

Введение в SRE

1. **Определение ролей в SRE:** В команде разработки приложения для онлайн-магазина введите роли SRE-инженера, DevOps-инженера и разработчика. Опишите, как эти роли взаимодействуют в процессе обеспечения надёжности, и предложите план распределения обязанностей для обработки инцидентов.

2. **Установка SLO и Error Budget:** Для сервиса доставки еды определите SLO по доступности (99,9%), SLI (время отклика API) и Error Budget (0,1% downtime). Рассчитайте допустимое время простоя за месяц и объясните, как Error Budget влияет на принятие решений о релизах.

3. **Работа с CRD в Kubernetes:** Создайте собственное Custom Resource Definition (CRD) для управления конфигурациями микросервисов в Kubernetes. Разверните его в кластере и продемонстрируйте, как оно интегрируется с контроллерами для автоматизации масштабирования.

4. **Настройка Ingress для сетевого доступа:** В Kubernetes-кластере настройте Ingress для веб-приложения с поддержкой HTTPS, маршрутизацией трафика на основе путей и базовой аутентификацией. Протестируйте доступность приложения извне и обработку ошибок 404/500.

5. **Деплой тестового приложения и организация on-call:** Разверните простое веб-приложение (например, на базе Nginx) в Kubernetes с использованием Helm-чартов. Организуйте ротацию on-call дежурств для команды из 3 человек, включая расписание и процедуры эскалации инцидентов в нерабочее время.

Современные инструменты SRE

1. **Сбор метрик с Prometheus:** Установите Prometheus в Kubernetes-кластере и настройте сбор метрик с приложений (например, CPU, память, HTTP-запросы). Создайте правила для агрегации данных и экспортируйте их в хранилище для последующего анализа.

2. **Анализ логов с Kibana:** Интегрируйте ELK-стек (Elasticsearch, Logstash, Kibana) для сбора логов из Kubernetes-подов. Настройте дашборд в Kibana для фильтрации и поиска ошибок в логах приложения, включая визуализацию трендов по времени.

3. **Обсервабилити Kubernetes:** В кластере Kubernetes настройте мониторинг метрик (с Prometheus), логов (с Fluentd) и событий (с Kubernetes API). Создайте базовую практику для автоматического оповещения при превышении порогов по использованию ресурсов.

4. **Проектирование метрик и SLI:** Для API-сервиса спроектируйте метрики (latency, error rate, throughput) и соответствующие SLI по best practices (RED/USE методологии). Объясните, как эти метрики помогут в оценке надёжности и предложите способы их измерения.

5. **Визуализация с Grafana:** Создайте дашборд в Grafana, подключённый к Prometheus, с панелями для SLO (доступность, latency) и пользовательскими метриками. Добавьте алерты на основе порогов и продемонстрируйте, как дашборд помогает в мониторинге состояния системы.

Мониторинг

1. **Настройка алертинга с Alertmanager:** В Prometheus настройте правила алертинга для критических метрик (например, downtime сервиса) и маршрутизацию через Alertmanager на Slack/email. Создайте runbook с шагами по устранению типичного инцидента, такого как перегрузка CPU.

2. **Реализация стратегии rolling deployment:** В Kubernetes реализуйте rolling deployment для обновления версии приложения без downtime. Опишите процесс, включая проверку health checks, и смоделируйте сценарий отката при обнаружении ошибок.

3. **Blue/Green и Canary deployment:** Настройте blue/green deployment для веб-сервиса в Kubernetes, включая балансировку трафика. Затем перейдите к canary-стратегии, постепенно направляя 10% трафика на новую версию, и опишите критерии для полного переключения.

4. **Балансировка нагрузки и устойчивость:** Настройте L7-балансировщик (например, с NGINX Ingress) для распределения трафика между микросервисами. Реализуйте механизмы устойчивости, такие как circuit breaker и retry, для обработки сбоев в зависимостях.

5. **Troubleshooting инцидентов:** В кластере Kubernetes с приложением и Kafka разберитесь инцидент: потеря сообщений в топике. Используйте инструменты (kubectl, logs, metrics) для диагностики, определите root cause (например, конфигурация retention) и предложите меры по предотвращению.

Примерные вопросы для коллоквиума

1. Что такое SRE (Site Reliability Engineering) и каковы основные роли в этой дисциплине?

2. Объясните ключевые процессы в SRE, такие как управление инцидентами и постмортемы.

3. Что такое SLO, SLI и Error Budget? Приведите примеры их применения в практике.

4. Что такое CRD (Custom Resource Definitions) в Kubernetes и как они расширяют функциональность кластера?

5. Как работает Ingress в Kubernetes для управления сетевым доступом? Приведите пример конфигурации.

6. Опишите процесс деплоя тестового приложения в Kubernetes и организацию on-call дежурств.
7. Как Prometheus собирает и хранит метрики? Объясните основные компоненты его архитектуры.
8. Что такое ELK-стек и как Kibana используется для анализа логов? Как это относится к OpenSearch?
9. Какие виды обсервабилити доступны в Kubernetes (метрики, логи, события) и какие базовые практики их мониторинга?
10. Как проектировать метрики приложения и SLI по best practices? Приведите примеры хороших и плохих метрик.
11. Что такое SLO-панели в Grafana и как они помогают в визуализации данных?
12. Опишите процесс создания дашбордов в Grafana для мониторинга Kubernetes-кластера.
13. Как интегрировать Prometheus с Grafana для построения комплексных дашбордов по SLO?
14. Как работают правила алертинга в Prometheus и маршрутизация через Alertmanager?
15. Что такое runbook и как он используется в процессе обработки алертов?
16. Сравните стратегии деплоя: rolling, blue/green, canary и прогрессивная выдача. Когда какую выбрать?
17. Объясните разницу между L4 и L7 балансировкой нагрузки и общие подходы к обеспечению устойчивости системы.
18. Как проводить troubleshooting в Kubernetes? Приведите примеры распространенных проблем и их решений.

Примерное описание и критерии оценивания к проекту

Тема проекта: Разработка и внедрение комплексного решения для развертывания, мониторинга и поддержки надежного приложения в Kubernetes с использованием инструментов SRE.

Задачи проекта:

- Установить и настроить Kubernetes-кластер с необходимыми CRD и Ingress-контроллером.
- Развернуть тестовое приложение и обеспечить его доступность через Ingress.
- Настроить сбор метрик с помощью Prometheus и визуализацию в Grafana.
- Организовать централизованный сбор и анализ логов с помощью Kibana.
- Разработать и внедрить правила алертов с Alertmanager.
- Настроить балансировку нагрузки для обеспечения отказоустойчивости.
- Провести тестирование и диагностику системы, описать процесс OnCall.

Критерии оценивания:

1. Корректность и полнота развертывания Kubernetes-ресурсов:

- Правильная установка CRD и Ingress-контроллера.
- Успешный деплой и доступность приложения.

2. Настройка мониторинга и визуализации:

- Корректный сбор метрик Prometheus.
- Создание информативных дашбордов в Grafana.
- Централизованный сбор и анализ логов в Kibana.

3. Управление алертами:

- Разработка релевантных правил алертов.
- Настройка Alertmanager и проверка получения уведомлений.

4. Обеспечение устойчивости и балансировка нагрузки:

- Настройка и проверка работы балансировщика нагрузки.
- Оценка влияния балансировки на отказоустойчивость.

5. Диагностика и OnCall-процессы:

- Проведение тестирования и устранение выявленных проблем.
- Описание процесса OnCall и действий при инцидентах.

6. Документированность и качество отчёта:

- Четкое описание выполненных шагов и результатов.
- Анализ проблем и предложенные улучшения.
- Качество оформления и полнота отчёта.

Задания для промежуточной аттестации по дисциплине (модулю)

№ п/п	Задание	Ответ	Компетенция
1	Назовите основной инструмент для сбора метрик в SRE.	Prometheus	УК-1
2	Укажите тип источника информации, который считается наиболее релевантным для анализа best practices SLO в Kubernetes.	Документация	УК-1
3	Назовите метод синтеза данных из метрик Prometheus и логов ELK для troubleshooting инцидента.	Агрегация	УК-1
4	Укажите количество основных этапов в системном подходе к анализу инцидента с потерей данных в Kafka.	4	УК-1
5	Назовите инструмент для визуализации дашбордов SLO в SRE.	Grafana	УК-1
6	Укажите математическую концепцию, применяемую для расчета вероятности превышения SLO.	Теория вероятностей	ОПК-1
7	Назовите статистическую меру для оценки дисперсии latency в метриках Prometheus.	Дисперсия	ОПК-1
8	Укажите численный метод, используемый для аппроксимации трендов метрик в Grafana.	Регрессия	ОПК-1
9	Назовите раздел дискретной математики, применяемый для проектирования логических правил алертинга.	Математическая логика	ОПК-1
10	Укажите тип случайного процесса для моделирования нагрузки в Kubernetes.	Пуассоновский процесс	ОПК-1
11	Назовите алгоритм сбора метрик в Prometheus, связанный с анализом временных рядов.	Sliding window	ОПК-6
12	Укажите алгоритм обработки логов в ELK для anomaly detection.	Clustering	ОПК-6
13	Назовите язык программирования для автоматизации алертинга с Alertmanager.	Python	ОПК-6
14	Укажите математический метод для расчетов SLI на основе метрик Kubernetes.	Статистическое среднее	ОПК-6
15	Назовите тип интеллектуальной системы для визуализации предиктивного анализа инцидентов.	Дашборд в Grafana	ОПК-6
16	Укажите уровень uptime в процентах для SLO 99.9%.	99.9	ПК-1
17	Назовите базовый принцип программирования, связанный с анализом логов в SRE.	Парсинг	ПК-1
18	Укажите математический метод для оптимизации SLI по latency.	Линейная регрессия	ПК-1
19	Назовите ключевой элемент анализа результатов troubleshooting в Kafka.	Метрики	ПК-1
20	Укажите математический метод для моделирования балансировки нагрузки в SRE.	Оптимизация	ПК-1