МИНИСТЕРСТВО ТРАНСПОРТА РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«РОССИЙСКИЙ УНИВЕРСИТЕТ ТРАНСПОРТА»
(РУТ (МИИТ)

Рабочая программа дисциплины (модуля),
как компонент образовательной программы
специализированного высшего образования
по направлению подготовки
09.04.01 Информатика и вычислительная техника,
утвержденной первым проректором ФГАОУ ВО
РУТ (МИИТ) Тимониным В.С.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Инженерия хаоса и тестирование отказоустойчивости

Направление подготовки: 09.04.01 Информатика и вычислительная техника

Направленность (профиль): Технологии проектирования программного обеспечения

Форма обучения: Очная

1. Общие сведения о дисциплине (модуле).

Дисциплина посвящена инженерным методам проверки отказоустойчивости распределенных программных систем. В ходе изучения рассматриваются надежность, доступность, устойчивость к отказам, модель критичных пользовательских сценариев, контрольные показатели сервиса, наблюдаемость, нагрузочные проверки, безопасное внесение отказов, сетевые задержки, отказ зависимостей, потеря узла, деградация базы данных, восстановление после сбоя и документирование результатов. На практических занятиях обучающиеся последовательно проектируют программу проверки отказоустойчивости серверного приложения и готовят техническую документацию с гипотезами, сценариями, рисками, метриками и планом улучшений.

Целью освоения дисциплины является формирование способности проектировать, проводить, анализировать и документировать проверки отказоустойчивости распределенных программных систем, позволяющие заранее выявлять слабые места архитектуры, оценивать влияние отказов на пользовательские сценарии и обосновывать меры повышения эксплуатационной устойчивости.

Для достижения поставленной цели в рамках дисциплины решается комплекс задач, направленных на формирование у обучающихся способности – анализировать архитектуру распределенной системы, выделять критичные пользовательские сценарии, формулировать гипотезы устойчивости, определять показатели нормальной работы, проектировать безопасные сценарии внесения отказов, оценивать влияние сетевых, вычислительных и прикладных сбоев, сопоставлять метрики, журналы событий и трассировки, планировать восстановление и готовить техническую документацию.

2. Планируемые результаты обучения по дисциплине (модулю).

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

ПК-2 - Способен проектировать и разрабатывать распределенные высокопроизводительные программные продукты с применением методов оптимизации программного обеспечения для корпоративного рынка;

ПК-4 - Способен осуществлять руководство процессом обеспечения качества разрабатываемого программного продукта для корпоративного рынка.

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

Знать:

- понятие отказоустойчивости программной системы и ее связь с доступностью, надежностью, восстанавливаемостью и качеством пользовательского сценария
- принципы инженерии хаоса как управляемого эксперимента над распределенной системой для повышения уверенности в ее устойчивости
- модель распределенной программной системы, включая сервисы, базы данных, очереди сообщений, кэш, внешние зависимости, сеть и инфраструктурные ресурсы
- способы выделения критичных пользовательских сценариев, включая целевое действие, допустимое время ответа, долю ошибок и влияние на бизнес-процесс
- показатели нормальной работы сервиса, включая доступность, задержку, пропускную способность, долю ошибок, насыщение ресурсов и время восстановления
- требования к безопасному проведению экспериментов с отказами, включая область воздействия, остановку эксперимента, окно проведения и уведомление ответственных лиц
- классификация отказов распределенной системы, включая отказ узла, отказ контейнера, сетевую задержку, потерю пакетов, недоступность зависимости, исчерпание ресурсов и ошибку конфигурации
- методы внесения отказов в контейнерной и облачной среде, включая остановку модуля, ограничение ресурсов, сетевое нарушение и отказ внешней службы
- назначение Kubernetes при анализе отказов контейнерных приложений, включая модуль, развертывание, службу, проверку готовности и самовосстановление
- назначение LitmusChaos, Chaos Mesh и Toxiproxy при проектировании управляемых отказов в распределенных системах
- роль наблюдаемости при проведении экспериментов, включая метрики, журналы событий, трассировки, проверки доступности и временную линию события
- методы нагрузочной проверки устойчивости, включая профиль нагрузки, контрольную нагрузку, предельную нагрузку и сравнение поведения до и после отказа

- способы анализа деградации сервиса, включая рост задержки, каскадный отказ, накопление очереди, повторные запросы и исчерпание соединений

- архитектурные меры повышения устойчивости, включая тайм-ауты, повторные попытки, ограничение скорости, предохранитель, резервирование, идемпотентность и безопасную деградацию

- принципы восстановления после отказа, включая резервное копирование, восстановление данных, переключение на резервный ресурс и проверку целостности состояния

- требования к документированию эксперимента, включая гипотезу, область воздействия, метрики, сценарий отказа, результат, выводы, риски и план улучшений

- этические и организационные ограничения инженерии хаоса, включая недопущение вреда пользователям, сохранность данных, прозрачность действий и ответственность команды

Уметь:

- уметь анализировать архитектуру распределенной системы при помощи схемы сервисов, зависимостей и потоков данных в условиях транспортно-логистического программного продукта

- уметь выделять критичные пользовательские сценарии при помощи карты бизнес-процесса и показателей качества в условиях ограниченного времени простоя

- уметь формулировать гипотезу устойчивости при помощи описания нормального состояния, ожидаемого отказа и допустимой реакции системы в условиях контролируемого эксперимента

- уметь проектировать безопасный эксперимент при помощи области воздействия, правил остановки и плана уведомления в условиях защиты данных и пользовательских сценариев

- уметь выбирать метрики наблюдения при помощи Prometheus, Grafana, Loki и Tempo в условиях анализа времени ответа, ошибок, ресурсов и трассировок

- уметь проектировать нагрузочную проверку при помощи k6 или Apache JMeter в условиях сравнения нормального режима, деградации и восстановления

- уметь проектировать отказ контейнерного компонента при помощи Kubernetes и LitmusChaos или Chaos Mesh в условиях проверки самовосстановления приложения

- уметь проектировать сетевое нарушение при помощи Toxiproxy или Chaos Mesh в условиях проверки тайм-аутов, повторных попыток и устойчивости зависимостей

- уметь оценивать отказ базы данных или очереди сообщений при помощи сценария недоступности, задержки и восстановления в условиях сохранения целостности данных

- уметь анализировать результаты эксперимента при помощи сопоставления метрик, журналов событий и трассировок в условиях поиска причины деградации

- уметь обосновывать меры повышения устойчивости при помощи тайм-аутов, повторных попыток, предохранителя, идемпотентности и безопасной деградации в условиях конкретного отказа

- уметь готовить техническую документацию по проверке отказоустойчивости при помощи описания гипотез, сценариев, рисков, результатов и плана улучшений

Владеть:

- навыком анализа архитектуры распределенной программной системы с точки зрения отказов и зависимостей

- навыком формулирования гипотез устойчивости и показателей нормальной работы сервиса

- навыком проектирования безопасных сценариев внесения отказов в контейнерной и облачной среде

- навыком выбора метрик, журналов событий и трассировок для наблюдения за экспериментом

- навыком проектирования нагрузочной проверки и анализа деградации сервиса

- навыком анализа отказов Kubernetes-компонентов, сетевых нарушений и отказов внешних зависимостей

- навыком подготовки мер повышения отказоустойчивости и плана восстановления

- навыком подготовки технической документации по результатам проверки отказоустойчивости

3. Объем дисциплины (модуля).

3.1. Общая трудоемкость дисциплины (модуля).

Общая трудоемкость дисциплины (модуля) составляет 2 з.е. (72 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
Контактная работа при проведении учебных занятий (всего):	48	48
В том числе:		
Занятия лекционного типа	16	16
Занятия семинарского типа	32	32

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 24 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	Отказоустойчивость распределенных программных систем Рассматриваемые вопросы: - доступность, надежность, восстанавливаемость и качество пользовательского сценария; - типовые причины отказов распределенных приложений; - связь отказоустойчивости с архитектурой, эксплуатацией и сопровождением сервиса.
2	Инженерия хаоса как управляемый эксперимент Рассматриваемые вопросы: - гипотеза устойчивости, область воздействия и ожидаемое поведение системы; - отличие эксперимента с отказом от обычной проверки качества; - ограничения безопасного проведения экспериментов над программной системой.
3	Критичные пользовательские сценарии и показатели сервиса Рассматриваемые вопросы: - целевое действие пользователя, допустимое время ответа и допустимая доля ошибок;

№ п/п	Тематика лекционных занятий / краткое содержание
	- доступность, задержка, пропускная способность, насыщение ресурсов и время восстановления; - выбор показателей для транспортно-логистического программного продукта.
4	Модель отказов распределенной системы Рассматриваемые вопросы: - отказ узла, контейнера, базы данных, очереди сообщений и внешней зависимости; - сетевые задержки, потеря пакетов, разрыв соединения и ошибка конфигурации; - каскадный отказ, накопление очереди и исчерпание ресурсов.
5	Наблюдаемость эксперимента с отказом Рассматриваемые вопросы: - метрики, журналы событий, трассировки и проверки доступности; - временная линия эксперимента и сопоставление событий разных компонентов; - признаки деградации, восстановления и нарушения критерия остановки.
6	Нагрузочная проверка устойчивости Рассматриваемые вопросы: - профиль нагрузки, контрольная нагрузка, предельная нагрузка и длительность проверки; - сравнение поведения системы до отказа, во время отказа и после восстановления; - применение k6 и Apache JMeter при проектировании проверочных сценариев.
7	Отказы контейнерной инфраструктуры Kubernetes Рассматриваемые вопросы: - модуль, развертывание, служба, проверка готовности и самовосстановление; - удаление модуля, ограничение ресурсов и недоступность узла; - оценка влияния отказа контейнерного компонента на пользовательский сценарий.
8	Инструменты управляемого внесения отказов Рассматриваемые вопросы: - назначение LitmusChaos и Chaos Mesh при проверке Kubernetes-приложений; - назначение Toxiproxy при проверке сетевых зависимостей; - правила выбора инструмента по типу отказа и области воздействия.
9	Сетевые нарушения и отказ внешних зависимостей Рассматриваемые вопросы: - задержка, потеря пакетов, недоступность службы и нестабильное соединение; - тайм-ауты, повторные попытки и ограничение числа обращений; - предотвращение каскадного отказа при нарушении сетевого взаимодействия.
10	Отказы хранилищ данных и очередей сообщений Рассматриваемые вопросы: - задержка базы данных, потеря соединения, переполнение очереди и повторная доставка сообщений; - идемпотентность, целостность данных и восстановление после повторной обработки; - контроль резервного копирования и восстановления состояния.
11	Архитектурные приемы повышения устойчивости Рассматриваемые вопросы: - тайм-аут, повторная попытка, предохранитель и ограничение скорости; - безопасная деградация, резервирование и разделение критичных зависимостей; - связь архитектурного решения с проверяемой гипотезой устойчивости.
12	План восстановления после отказа Рассматриваемые вопросы: - целевое время восстановления, целевая точка восстановления и порядок действий ответственных лиц; - переключение на резервный ресурс и проверка целостности данных; - обучение команды действиям при аварийном событии.

№ п/п	Тематика лекционных занятий / краткое содержание
13	Безопасность и организационные ограничения экспериментов Рассматриваемые вопросы: - защита пользовательских данных и недопущение вреда реальным пользователям; - согласование окна эксперимента, уведомление ответственных лиц и критерии остановки; - журналирование действий и ответственность за внесенные воздействия.
14	Анализ результатов эксперимента Рассматриваемые вопросы: - сопоставление гипотезы, фактической реакции системы и телеметрических данных; - выявление первопричины деградации и проверка альтернативных объяснений; - подготовка выводов о рисках, ограничениях и необходимых изменениях.
15	План улучшения отказоустойчивости Рассматриваемые вопросы: - ранжирование выявленных рисков по влиянию, вероятности и трудоемкости устранения; - выбор архитектурных, эксплуатационных и организационных мер повышения устойчивости; - повторная проверка после внесения изменений.
16	Документирование проверки отказоустойчивости Рассматриваемые вопросы: - структура технической документации по гипотезам, сценариям, метрикам и рискам; - оформление результатов, временной линии, выводов и плана улучшений; - критерии готовности материалов к передаче команде сопровождения.

4.2. Занятия семинарского типа.

Практические занятия

№ п/п	Тематика практических занятий/краткое содержание
1	Анализ архитектуры распределенной системы Студент выбирает серверное приложение и описывает его сервисы, базы данных, очереди сообщений, внешние зависимости и пользовательские сценарии. На схеме фиксируются потоки данных и точки возможного отказа. Для каждого компонента определяется влияние на работу системы.
2	Выделение критичных пользовательских сценариев Студент выбирает ключевые сценарии и задает для них допустимое время ответа, долю ошибок, доступность и время восстановления. Для каждого сценария описывается последствие отказа. Итоговая таблица используется как основа для экспериментов.
3	Формулирование гипотез устойчивости Студент формулирует гипотезы о поведении системы при отказе зависимости, сетевой задержке, потере контейнера и исчерпании ресурса. Для каждой гипотезы задаются ожидаемые метрики и допустимая реакция системы. Отдельно фиксируются критерии остановки эксперимента.
4	Проектирование безопасного плана эксперимента Студент определяет область воздействия, подготовительные проверки, окно проведения, ответственных лиц и действия при нарушении критерия остановки. Для эксперимента задаются ограничения по данным и пользовательским сценариям. План проверяется на отсутствие лишнего воздействия на систему.
5	Выбор метрик и источников наблюдения Студент выбирает метрики, журналы событий, трассировки и проверки доступности для каждого сценария отказа. Для показателей задаются нормальные значения и пороги деградации. Составляется таблица наблюдения за экспериментом.

№ п/п	Тематика практических занятий/краткое содержание
6	Проектирование нагрузочной проверки Студент описывает профиль нагрузки, частоту запросов, длительность проверки и ожидаемое поведение системы. Для k6 или Apache JMeter проектируется набор запросов и контрольных данных. Сценарий связывается с выбранными метриками качества сервиса.
7	Проектирование отказа контейнерного компонента Студент описывает воздействие на Kubernetes-компонент через удаление модуля, ограничение ресурсов или недоступность узла. Для воздействия указываются ожидаемое самовосстановление и влияние на пользовательский сценарий. Результат оформляется как спецификация эксперимента.
8	Проектирование сетевого нарушения Студент задает задержку, потерю пакетов или недоступность внешней службы при помощи Toxiproxy или Chaos Mesh. Для сценария описываются тайм-ауты, повторные попытки и ожидаемая безопасная деградация. Отдельно фиксируются признаки каскадного отказа.
9	Проектирование отказа хранилища данных Студент моделирует задержку базы данных, потерю соединения или недоступность очереди сообщений на уровне архитектурного сценария. Для каждого воздействия описывается влияние на целостность данных и повторную обработку. Предлагаются проверки резервного копирования и восстановления.
10	Анализ устойчивости к каскадному отказу Студент прослеживает путь распространения отказа от одной зависимости к нескольким пользовательским сценариям. На схеме отмечаются повторные запросы, накопление очередей и исчерпание соединений. Формулируются меры ограничения распространения отказа.
11	Проектирование архитектурных мер устойчивости Студент подбирает тайм-ауты, повторные попытки, предохранитель, ограничение скорости, идемпотентность и безопасную деградацию для выбранных отказов. Для каждой меры указывается ожидаемое изменение поведения системы. Меры связываются с исходными гипотезами.
12	Проектирование плана восстановления Студент задает порядок восстановления после отказа базы данных, очереди сообщений, внешней службы или контейнерного компонента. Для каждого случая фиксируются ответственные действия, контроль целостности данных и момент возврата к штатному режиму. План сопоставляется с целевым временем восстановления.
13	Оценка безопасности эксперимента Студент анализирует риски воздействия на данные, пользователей, смежные сервисы и команду сопровождения. Для каждого риска задается мера снижения и условие прекращения эксперимента. Итоговая карта рисков прикладывается к плану проверки.
14	Разбор результатов отказного сценария Студент получает набор метрик, журналов событий и трассировок для заданного отказа и восстанавливает временную линию события. Выявляется первопричина деградации и проверяются альтернативные объяснения. Выводы оформляются в таблицу результатов.
15	Планирование улучшений отказоустойчивости Студент ранжирует выявленные риски по влиянию, вероятности и трудоемкости устранения. Для приоритетных рисков выбираются архитектурные, эксплуатационные и организационные меры. Для каждой меры задается способ повторной проверки.
16	Подготовка технической документации по проверке отказоустойчивости Студент собирает архитектурную схему, критичные сценарии, гипотезы, планы экспериментов, метрики, риски, результаты разбора и план улучшений. Документация проверяется на полноту и воспроизводимость. Итоговый материал оформляется как связное портфолио проверки отказоустойчивости.

4.3. Самостоятельная работа обучающихся.

№ п/п	Вид самостоятельной работы
1	Изучение рекомендованной литературы.
2	Подготовка к промежуточной аттестации.

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

№ п/п	Библиографическое описание	Место доступа
1	Баланов, А. Н. Построение микросервисной архитектуры и разработка высоконагруженных приложений : учебное пособие для вузов / А. Н. Баланов. — Санкт-Петербург : Лань, 2024. — 244 с. — ISBN 978-5-507-48747-9. — Текст : электронный	Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/394538 (дата обращения: 18.06.2026)
2	Морган, Д. Linkerd на практике. Работа с приложениями в Kubernetes : руководство / Д. Морган, Флинн ; пер. с англ. В. И. Бахура. — Москва : ДМК Пресс, 2025. — 230 с. — ISBN 978-6-01123-649-2. — Текст : электронный	Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/514966 (дата обращения: 18.06.2026)
3	Аймен, Э. А. Облачные микросервисы в Kubernetes : руководство / Э. А. Аймен ; пер. с англ. В. С. Яценкова. — Москва : ДМК Пресс, 2025. — 276 с. — ISBN 978-5-93700-324-9. — Текст : электронный	Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/514873 (дата обращения: 18.06.2026)
4	Программные инструменты обработки и визуализации данных. Elasticsearch, Logstash, Kibana, Grafana, Prometheus : учебное пособие / И. В. Никифоров, О. А. Юсупова, Н. В. Воинов [и др.] ; под редакцией И. В. Никифорова [и др.]. — Санкт-Петербург : СПбГПУ, 2023. — 140 с. — ISBN 978-5-7422-8075-0. — Текст : электронный	Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/430157 (дата обращения: 18.06.2026)

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/.

Принципы инженерии хаоса – https://principlesofchaos.org/.

Документация Kubernetes – https://kubernetes.io/docs/.

Документация LitmusChaos – https://litmuschaos.io/docs/.

Документация Chaos Mesh – https://chaos-mesh.org/docs/.

Документация Toxiproxy – https://github.com/Shopify/toxiproxy.

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

Документация Grafana – https://grafana.com/docs/.

Документация k6 – https://grafana.com/docs/k6/.

Документация Apache JMeter – https://jmeter.apache.org/usermanual/.

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.

Контейнерная среда – Kubernetes, Podman.

Проверка отказоустойчивости – LitmusChaos, Chaos Mesh, Toxiproxy.

Нагрузочная проверка и наблюдаемость – k6, Apache JMeter, Prometheus, Grafana.

Сопровождение проекта – Git, Markdown.

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 занятий – наличие персональных компьютеров вычислительного класса.

9. Форма промежуточной аттестации:

Зачет в 3 семестре.

10. Оценочные материалы.

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

Авторы:

старший преподаватель кафедры
«Цифровые технологии управления
транспортными процессами» Е.А. Заманов

Согласовано:

Заведующий кафедрой ЦТУТП В.Е. Нутович

Председатель учебно-методической
комиссии Н.А. Андриянова