Федеральное государственное бюджетное образовательное учреждение высшего образования
«РОССИЙСКАЯ АКАДЕМИЯ НАРОДНОГО ХОЗЯЙСТВА И ГОСУДАРСТВЕННОЙ СЛУЖБЫ ПРИ ПРЕЗИДЕНТЕ РОССИЙСКОЙ ФЕДЕРАЦИИ»

ИНСТИТУТ ЭКОНОМИКИ, МАТЕМАТИКИ И ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ
ФАКУЛЬТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И АНАЛИЗА ДАННЫХ
КАФЕДРА СИСТЕМНОГО АНАЛИЗА И ИНФОРМАТИКИ

УТВЕРЖДЕНА
решением кафедры Системного анализа и информатики
Протокол от «03 » сентября 2020 г. № 01

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Б1.В.11 «DevOps инжиниринг»
(индекс и наименование дисциплины (модуля), в соответствии с учебным планом)

09.03.03 Прикладная информатика _
(код и наименование направления подготовки (специальности))

" Веб-разработка"
направленность (профиль/специализация)

бакалавр
квалификация

очная форма обучения
форма(ы) обучения

Год набора - 2021

Москва, 2020 г.

Автор составитель:

Профессор кафедры Системного анализа и информатики
д-р техн. наук, профессор Ромашкова О.Н.

Заведующий кафедрой
Системного анализа и информатики Маруев С.А.
(*наименование кафедры)* *(ученая степень и(или) ученое звание)* *(подпись)* *(Ф.И.О.)*

СОДЕРЖАНИЕ

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы…….....

2. Объем и место дисциплины в структуре образовательной программы…….....

3. Содержание и структура дисциплины …….

4. Материалы текущего контроля успеваемости обучающихся и фонд оценочных средств промежуточной аттестации по дисциплине …….

5. Методические указания для обучающихся по освоению дисциплины…….....

6. Учебная литература и ресурсы информационно-телекоммуникационной сети "Интернет", учебно-методическое обеспечение самостоятельной работы обучающихся по дисциплине……..…….

6.1. Основная литература……..……..……..……

6.2. Дополнительная литература……...……..……..

6.3. Учебно-методическое обеспечение самостоятельной работы……

6.4. Нормативные правовые документы…….……..……

6.5. Интернет-ресурсы…….……..……

6.6. Иные источники…….…….…….…

7. Материально-техническая база, информационные технологии, программное обеспечение и информационные справочные системы …..…….

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения программы

1.1. Дисциплина Б1.В.11 «DevOps инжиниринг» обеспечивает овладение следующими компетенциями с учетом этапа:

Код компетенции	Наименование компетенции	Код этапа освоения компетенции	Наименование этапа освоения компетенции
ПКр – 2	Способность разрабатывать, внедрять и адаптировать прикладное программное обеспечение	ПКр-2.1 ПКр-2.2 ПКр-2.3	Знает принципы, методы и стандарты разработки архитектуры ИС Умеет профессионально выполнять разработку баз данных ИС Владеет навыками разработки архитектуры и баз данных ИС
ПКр-3	Способность проектировать ИС в соответствии с профилем подготовки по видам обеспечения	ПКр-3.1 ПКр-3.2. ПКр-3.3.	Знает основные методологии и средства проектирования и дизайна ИС Умеет выполнять проектирование и дизайн ИС Владеет навыками проектирования и разработки дизайна ИС
ПКр-4	способность документировать процессы создания информационных систем на стадиях жизненного цикла	ПКр-4.1 ПКр-4.2.	Знает принципы документирования существующих бизнес-процессов организации заказчика (реверс-инжиниринг бизнес-процессов организации)

		ПКр-4.3.	Умеет выполнять документирование существующих бизнес-процессов организации заказчика Владеет навыками создания пользовательской документации к ИС

1.2 В результате освоения дисциплины у обучаемых должны быть сформированы:

ОТФ/ТФ *(при наличии профстандарта)*/ трудовые или профессиональные действия	Код этапа освоения компетенции	Результаты обучения

06.015 СПЕЦИАЛИСТ ПО ИНФОРМАЦИОННЫМ СИСТЕМАМ С Выполнение работ и управление работами по созданию (модификации) и сопровождению ИС, автоматизирующих задачи организационного управления и бизнес-процессы С/14.6 Разработка архитектуры ИС С/17.6 Разработка баз данных ИС	ПКр-2.1

06.015 СПЕЦИАЛИСТ ПО ИНФОРМАЦИОННЫМ СИСТЕМАМ С Выполнение работ и управление работами по созданию (модификации) и сопровождению ИС, автоматизирующих задачи организационного управления и бизнес-процессы С/16.6 Проектирование и дизайн ИС	ПКр-3.1 ПКр-3.2. ПКр-3.3.	ПКр-3.1.1 Знает основные методологии проектирования и дизайна ИС ПКр-3.1.2 Знает основные средства проектирования и дизайна ИС ПКр-3.2.1 Умеет применять основные методологии проектирования и дизайна ИС ПКр-3.2.2 Умеет применять основные средства проектирования и дизайна ИС ПКр-3.3.1 Владеет навыками проектирования ИС ПКр-3.3.2 Владеет навыками разработки дизайна ИС

06.015 СПЕЦИАЛИСТ ПО ИНФОРМАЦИОННЫМ СИСТЕМАМ С Выполнение работ и управление работами по созданию (модификации) и сопровождению ИС, автоматизирующих задачи организационного управления и бизнес-процессы С/07.6 Документирование существующих бизнес-процессов организации заказчика (реверс-инжиниринг бизнес-процессов организации) С/22.6 Создание пользовательской документации к ИС	ПКр-4.1 ПКр-4.2. ПКр-4.3.	ПКр-4.1.1 Знает правила и стандарты разработки пользовательской документации к ИС ПКр-4.1.2 Знает основные принципы документирования и реинжиниринга бизнес-процессов организации ПКр-4.2.1 Умеет выполнять документирование существующих бизнес-процессов организации заказчика ПКр-4.2.2 Умеет применять правила и стандарты разработки пользовательской документации к ИС ПКр-4.3.1 Владеет навыками создания пользовательской документации к ИС ПКр-4.3.2 Владеет навыками реинжиниринга бизнес-процессов организации

2. Объем и место дисциплины в структуре ОП ВО

Объем дисциплины

Объем дисциплины Б1.В.11 «DevOps инжиниринг»в ЗЕ и академических/астрономических часах составляет 2 ЗЕ (72/54 ч). Дисциплина реализуется с применением дистанционных образовательных технологий.

Количество академических/астрономических часов по очной форме обучения, выделенных на контактную работу 16/12 часов (в т.ч. практ. – 16/12 ч.); на самостоятельную работу обучающихся - 56/42 часа.

Место дисциплины в структуре ОП ВО

В соответствии с учебным планом дисциплина Б1.В.11 «DevOps инжиниринг» относится к обязательной части учебного плана по направлению подготовки 09.03.03 Прикладная информатика. и изучается на 4 курсе в 7 семестре.
Дисциплины опирается на объём знаний информатики, вычислительных машин, сетей и систем телекоммуникаций, программирования;

Форма промежуточной аттестации для очной дистанционной формы обучения – зачёт.

3. Содержание и структура дисциплины

№ п/п	Наименование тем и/или разделов	Объем дисциплины, час.						Форма текущего контроля успеваемости**, промежуточной аттестации ***
		Всего	Контактная работа обучающихся с преподавателем по видам учебных занятий				СР	
			Л/ЭО, ДОТ*	ЛР/ ЭО, ДОТ*	ПЗ/ ЭО, ДОТ*	КСР		
Тема 1	DevOps и инфраструктурная инженерия.	9			2		7	О, К, Т
Тема 2	Системы управления версиями.	9			2		7	Т, О, К
Тема 3	Виртуализация.	9			2		7	О, К, Р, Т
Тема 4	Администрирование баз данных.	9			2		7	О, Т
Тема 5	Мониторинг и логи.	9			2		7	О, К, КР
Тема 6	Система управления конфигурациями	9			2		7	Т, О, КР
Тема 7	Анализ частых множеств признаков и ассоциативных правил.	9			2		7	Т, О, КР
Тема 8	Кластеризация данных	9			2		7	Т, О, КР
Промежуточная аттестация								За
Всего:		**72/54**			16/12		56/42	

Примечание:

** – при применении электронного обучения, дистанционных образовательных технологий в соответствии с учебным планом;*

*** – формы текущего контроля успеваемости: опрос (О), тестирование (Т), контрольная работа (КР), коллоквиум (К), реферат (Р).*

**** - разработчик указывает необходимые формы промежуточной аттестации: зачет (ЗаО).*

Содержание дисциплины

Тема 1. DevOps и инфраструктурная инженерия.

В этом модуле вы овладеете базовой терминологией, используемой в процессе создания программного обеспечения. Познакомитесь с популярными методологиями разработки и поймёте их различия. Узнаете, зачем нужны инфраструктурные архитекторы и чем они занимаются. Знакомство с основной терминологие. Различие DevOps и DevSecO. Цели методологии DevOps. Общее

представление об инфраструктуре современной разработки. Обзор используемых инструментов. Циклы и этапы разработки ПО. Гибкие методологии разработки: Agile, Scrum, Kanban.

Тема 2. Системы управления версиями.

В этому модуле вы узнаете, почему и как возникли системы управления версиями исходного кода. Получите практические навыки глубокой работы с Git репозиториями. Научитесь работать одновременно с несколькими репозиториями, синхронизировать их и создавать резервные копии. Познакомитесь со способами интеграции репозиториев со внешними системами. Узнаете, как контролировать процесс разработки ещё на уровне отправки кода в репозиторий. Обзор систем управления версиями. Подробное знакомство с Git. Установка Gitlab на собственный сервер.

Тема 3. Виртуализация.

Узнаете различия видов виртуализации и контейнеризации. Научитесь управлять виртуальными машинами с помощью libvirtd. Напишете несколько Dockerfile, которые можно будет использовать в дальнейших проектах как примеры. Научитесь запускать несколько контейнеров одновременно и объединять их в виртуальную сеть. Изучение задач, которые решает виртуализация. Технологии виртуализации. Системы управления виртуализацией. Контейнеризация на примере Docker и ContainerD. Практические навыки работы с Docker. Написание Dockerfile и использование Docker Compose.

Тема 4. Администрирование баз данных.

Узнаете, чем различаются SQL и NoSQL базы данных, и научитесь выбирать нужную из всего многообразия существующих решений. Познакомитесь с решениями для полнотекстового поиска. Овладеете навыками установки и настройки кеш-систем. Научитесь устанавливать и настраивать базы данных для нужд разработки. Познакомитесь с технологиями создания отказоустойчивых кластеров баз данных и кеш систем при помощи кластеризации и шардинга. Научитесь писать простые SQL-запросы и запросы для работы с NoSQL-базами данных.

Типы и структура СУБД. Администрирование PostgreSQL. Администрирование MySQL. Знакомство с NoSQL-базами данных на примере MongoDB. Кеш-системы Redis и Memcached. Elasticsearch. Кластеризация, шардинг и отказоустойчивость.

Тема 5. Мониторинг и логи.

Подробно разберётесь, зачем нужен мониторинг и какие параметры нужно контролировать. Узнаете, как организовать систему оповещения о различных событиях, чтобы узнавать о сбоях первым, а не от заказчика. Научитесь организовывать логирование всех действий приложений и анализировать эти логи. Овладеете навыками работы с elasticsearch, Logstash, Kibana и Graylog. Научитесь настраивать связку Prometehus + Grafana + Alertmanager. Познакомитесь с Zabbix для мониторинга физических и виртуальных машин. Зачем и что нужно мониторить. Системы для мониторинга. Системы оповещения о проблемах ELK: Elasticsearch, Logstash и Kibana. Graylog. Zabbix. Prometehus, Grafana, Alertmanager.

Тема 6. Система управления конфигурациями.

Научитесь описывать инфраструктуру в виде кода. Сможете настроить удаленный сервер и восстановить его конфигурацию в случае необходимости. Узнаете набор уже готовых шаблонов для решения типовых задач конфигурирования серверов. Обзор систем управления конфигурациями. Концепция «инфраструктура как код». Знакомство с Ansible. Практика использования Ansible.

Тема 7. Анализ частых множеств признаков и ассоциативных правил.

Задача анализа потребительской корзины. Поддержка и достоверность. Частые, замкнутые и максимальные частые множества. Алгоритм Априори. Меры «интересности правил».

Тема 8. Кластеризация данных.

Простые эвристические подходы. Алгоритм K-Means. Проблема устойчивости результатов и важность грамотной инициализации, алгоритм K-Means++. Выбор числа кластеров. Оценка качества кластеризации.

4. Материалы текущего контроля успеваемости обучающихся и фонд оценочных средств промежуточной аттестации по дисциплине

4.1. Формы и методы текущего контроля успеваемости обучающихся и промежуточной аттестации

4.1.1. В ходе реализации дисциплины Б1.В.11 «DevOps инжиниринг» используются следующие методы текущего контроля успеваемости обучающихся[1]:

Тема и/или раздел		Методы текущего контроля успеваемости в аудитории	Методы текущего контроля успеваемости с применением ДОТ
Тема 1	DevOps и инфраструктурная инженерия.	Опрос, коллоквиум	Тестирование
Тема 2	Системы управления версиями.	Опрос, коллоквиум	Тестирование
Тема 3	Виртуализация.	Опрос, коллоквиум	Тестирование, Реферат
Тема 4	Администрирование баз данных.	Опрос	Тестирование
Тема 5	Мониторинг и логи.	Опрос	Тестирование, Контрольная работа
Тема 6	Система управления конфигурациями.	Опрос	Тестирование, Контрольная работа
Тема 7	Анализ частых множеств признаков и ассоциативных правил.	Опрос, коллоквиум	Тестирование, Контрольная работа
Тема 8	Кластеризация данных.	Опрос, коллоквиум	Тестирование, Контрольная работа

4.1.2. Зачет проводится с применением следующих методов (средств):

Зачет проводится с применением следующих методов (средств): без использования дистанционных образовательных технологий в форме выполнения практических заданий на компьютере с использованием инструментария веб-технологий, а именно LAMP/LEMP/FAMP/FEMP. HTML, CSS, JavaScript, TypeScript. PHP 7, MySQL. Symfony, Laravel, Zend Framework, Yii, Kohana. LESS, SASS, PostCSS. Gulp, Webpack, Babel. БЭМ. React, Vue, Angular. Git, Mercurial. Jenkins, Gitlab.

[1] *Разработчик программы самостоятельно выбирает форму заполнения пункта 1.1.*

4.2. Материалы текущего контроля успеваемости обучающихся.

Преподаватель оценивает уровень подготовленности обучающихся к занятию по следующим показателям:

- опрос - устные ответы на вопросы преподавателя по теме занятия. Критерии оценивания устных ответов на вопросы преподавателя по теме занятия - правильность и полнота ответа;

- выступление с презентациями по реферату темы №3, 4 отчетные доклады с презентациями по предлагаемым темам. Критериями оценивания реферата:

степень усвоения понятий и категорий по теме;
умение работать с документальными и литературными источниками;
грамотность изложения материала;
самостоятельность работы, наличие собственной обоснованной позиции;
Критерии оценивания презентации
оформлен титульный лист;
количество слайдов соответствует требованиям (мин.10 слайдов);
использование дополнительных эффектов в Power point, инфографики;
слайды представлены в логической последовательности.

- выполнение практических заданий на компьютере. Критерии оценивания выполнения заданий:

правильность выполнения задания;
оригинальность решений;
полнота выполнения задания.

- количество правильных ответов при тестировании. Критерии оценки выполненных студентами тестов определяются преподавателем самостоятельно. Рекомендуются следующие критерии оценки:

1. 85% – 100% правильных ответов – «отлично»;
2. 66% – 84% правильных ответов – «хорошо»;
3. 50% – 65% правильных ответов – «удовлетворительно»;
4. менее 50% правильных ответов – «неудовлетворительно».

- выполнение контрольной работы на компьютере. Критерии оценивания обучаемого в ходе выполнения:

3 балла – задание выполнено полностью, студент показал отличные знания, проявил активную позицию при ответах на вопросы, отлично владеет инструментарием

2 балла – задание выполнено с недочетами, студент показал хорошие знания, хорошо ответил на вопросы, владеет инструментарием;

1 балл – задание выполнено не полностью, студент показал слабые знания, не полностью ответил на вопросы, слабо владеет инструментарием;

0 баллов – задание не выполнено, студент не знает, не может ответить на вопросы, не владеет инструментарием.

Оценка знаний, умений, навыков проводится на основе балльно-рейтинговой системы 70% из 100% (70 баллов из 100) - вклад по результатам посещаемости занятий, активности на занятиях, выступления с докладами, выполнение всех практических заданий и контрольных работ, правильность ответов в ходе

тестирования и контрольных практических заданий.

Детализация баллов и критерии оценки текущего контроля успеваемости утверждается на заседании кафедры.

Примеры типовых оценочных материалов исходя из методов текущего контроля успеваемости

1. Типовые вопросы для проведения опроса
2. Что такое Load Average, SWAP, inode?
3. Опишите модель OSI и все ее уровни?
4. Чем отличается TCP от UDP, как устанавливается соединение TCP?
5. Когда нужно использовать SWAP, а когда нет?
6. Чем отличается git pull от git fetch?
7. Чем отличается транзакция от запроса, в контексте БД?
8. Как починить chmod -x /bin/chmod?
9. Что такое DevOps, Agile?
10. С какими методологиями разработки ПО работал? Что знаешь о Scrum, Kanban и т.д?
11. Чем виртуальные машины отличаются от контейнеров?
12. Клиенты жалуются на то, что веб-сервис стал медленно работать, как последовательно устранить проблемы? Как обнаружить bottleneck?
13. Чем отличаются реляционные БД от нереляционных и key-value?
14. Что такое балансировщик?
15. Какие типы балансировки вы знаете?
16. Что такое blue-green deployment, канареечные релизы?
17. Как организовать поднятие инфраструктуры в облаке, в случае если отвалилась целая зона или регион?
18. Как налету мигрировать работающую базу из одного региона в другую?
19. Как строить ту или иную архитектуру проекта?
20. Как рабртать с облаками, AWS, GCP, Azure, OpenStack?
21. Что представляет собой докер-контейнер? Из каких двух базовых вещей состоят контейнеры (имеется ввиду namespaces и cgroups)?
22. Какие инструменты оркестрации контейнеров использовал?
23. Для чего нужен Kubernetes, Nomad, Swarm, Compose?
24. Как работает стек ELK, со стеком ПО от Hashicorp (Vault, Consul, Nomad, Terraform и т.д)?
25. Как работает с Prometheus?
26. В чем отличия Jenkins от TeamCity и других аналогичных систем?
27. Что такое Continuous Integration/Delivery/Deployment и чем они друг от друга отличаются?
28. Описать инфраструктуру java-приложений (ant/maven/gradle)?
29. Описать системы конфигурации (Ansible/Puppet/Chef и т.д)?
30. В чем между описанными выше системами разница?
31. Как в access-логе Nginx посмотреть самые активные IP-адреса за последние сутки с помощью BASH?
32. Как отсортировать массив в Python?
33. Написать код в реальном времени.

Вварианты тестовых заданий

1. **Что такое DevOps?**

- ☐ Концепция, которая обеспечивает непрерывную поставку обновлений продукта конечным пользователям
- ☐ Язык программирования
- ☐ Сотрудники, занимающиеся разработкой
- ☐ Модель разработки программного обеспечения

2. Почему раньше DevOps не было и зачем эта концепция появилась?

- ☐ Раньше разработка была более трудоемким процессом
- ☐ Раньше новые релизы не выходили так часто, как сейчас
- ☐ Увеличилось количество запросов на разработку
- ☐ Все ответы правильные

3. **Зачем компании DevOps-методология?**

- ☐ Помогает продукту быстрее выйти на рынок
- ☐ Повышает лояльность конечных пользователей
- ☐ Увеличивает производительность компаний
- ☐ Всё вышеперечисленное;

4. **Что изменилось с внедрением DevOps?**

- ☐ Разработчики постоянно объединяют изменения программного кода в центральном репозитории, а сборка и тестирование ПО выполняются автоматически
- ☐ За каждое обновление отвечает конкретный разработчик. Он контролирует, чтобы апдейт происходил без ошибок
- ☐ Разработчики регулярно сохраняют только важные изменения
- ☐ Стало меньше тестировщиков — разработчики теперь тестируют сами;

5. **Непрерывная доставка – ещё одна методика DevOps. При любых изменениях в программном коде выполняется автоматическая сборка, тестирование и подготовка к окончательному выпуску.Чем она полезна?**

- ☐ Разработчики могут писать код с ошибками, а непрерывная доставка будет их проверять и исправлять
- ☐ Задерживает сложные обновления и тестирует их, а к конечным пользователям допускает только мелкие апдейты
- ☐ У разработчиков всегда есть готовый к развертыванию собранный экземпляр ПО, прошедший стандартизированную процедуру тестирования
- ☐ Сигнализирует тимлиду, когда кто-то из разработчиков пытается внести в репозиторий код с ошибками;

6. **Зачем нужна архитектура микросервисов?**

- ☐ Чтобы сделать стоимость разработки более дешёвой
- ☐ Чтобы модули приложения могли работать отдельно друг от друга. У каждого из них своя задача
- ☐ Чтобы проще было тестировать ПО
- ☐ Чтобы отдать разработку каких-то микросервисов на аутсорс;

7. **Инфраструктура как код — модель, по которой настройка инфраструктуры аналогична процессу создания ПО.**
Это основа облачных вычислений. Но вот почему это неотъемлемая часть DevOps?

- ☐ Инфраструктура как код позволяет управлять виртуальными машинами на программном уровне. Не нужно вручную настраивать и обновлять отдельные компоненты оборудования. Это быстро, экономично и здорово уменьшает риски
- ☐ Потому что появилась новая профессия — «разработчик в облаке». И он как раз отвечает за разработку инфраструктур. Удобно, когда есть отдельный специалист. А ведь DevOps — это про удобство
- ☐ Масштабировать инфраструктуру сложно, но зато её настройка происходит гораздо быстрее и легче, чем раньше
- ☐ Дело идёт к созданию одного универсального языка программирования. Так разработчики смогут быстрее создавать приложения и сервисы;

8. **Почему мониторинг и ведение журналов играет особую роль в DevOps?**

- ☐ Методика помогает интегрировать обновления в продукт и быстро понимать, где происходит сбой в работе
- ☐ Методика помогает пользователям определять обновления, которые будут интегрированы в проект
- ☐ Методика не имеет никакой ценности для разработчиков. Она нужна отделу маркетинга для лучшей продажи продукта: понять, чего хочет пользователь и дать ему это
- ☐ Методика помогает сэкономить бюджет компании на разработку;

Варианты контрольных работ

1. Описать методологию devops.
2. Инструменты devops
3. Планирование и сотрудничество
4. Описать один из популярных инструментов управления проектами JIRA.
5. Описать бесплатный инструмент для совместной работы на основе облака, для групповой коммуникации и совместной работе в одном месте – Slack.
6. Рассказать про платформу для веб-конференций и мгновенного обмена экранами - Zoom.
7. Программное обеспечение для совместной работы и управления проектами, которое помогает отслеживать проблемы, управлять задачами и управлять портфелем проектов Clarizen.
8. Мобильное и веб-приложение, которое помогает командам эффективно и результативно организовывать, отслеживать и управлять своей работой Asana.
9. Управление исходным кодом
10. Централизованный инструмент контроля версий и исходного кода, разработанный Apache SVN.
11. Распределенная система контроля версий, которая нацелена на скорость, целостность данных, поддержку распределенных, нелинейных рабочих процессов Git.
12. Веб-хостинговая платформа, разработанная Atlassian Bitbucket.
13. Платформа для размещения кода, предназначенная для контроля версий и совместной работы GitHub.
14. Инструменты для сборки
15. Инструмент для сборки и развертывания на основе Java с открытым исходным кодом Apache Ant.
16. Инструмент для автоматизации сборки, который в основном используется для Java-проектов Maven.

Вопросы и задания для самостоятельной подготовки

Познакомиться с платформами управления и разработки проектами и познакомиться с программным обеспечением:

1. Инструменты devops
2. Инструменты планирования и сотрудничества
3. Инструмент управления проектами JIRA.
4. Инструмент для совместной работы на основе облака, для групповой коммуникации и совместной работе в одном месте – Slack.
5. Платформа для веб-конференций и мгновенного обмена экранами - Zoom.
6. Программное обеспечение для совместной работы и управления проектами, которое помогает отслеживать проблемы, управлять задачами и управлять портфелем проектов Clarizen.
7. Мобильное и веб-приложение, которое помогает командам эффективно и результативно организовывать, отслеживать и управлять своей работой Asana.
8. Инструменты управления исходным кодом
9. Централизованный инструмент контроля версий и исходного кода, разработанный Apache SVN.
10. Распределенная система контроля версий, которая нацелена на скорость, целостность данных, поддержку распределенных, нелинейных рабочих процессов Git.
11. Веб-хостинговая платформа, разработанная Atlassian Bitbucket.
12. Платформа для размещения кода, предназначенная для контроля версий и совместной работы GitHub.
13. Инструменты для сборки
14. Инструмент для сборки и развертывания на основе Java с открытым исходным кодом Apache Ant.
15. Инструмент для автоматизации сборки, который в основном используется для Java-проектов Maven.

Примерные темы для написания рефератов

1. В чем состоит суть DevOps-культуры?
2. Зачем нужен DevOps?
3. Какие бывают DevOps-практики?
4. Как соотносятся понятия «системный инженер», «билд-инженер» и «DevOps-инженер»?
5. Что должен знать DevOps-инженер?
6. Особенности реализации информационных систем в различных предметных областях.
7. Структурный системный анализ. Методы структурного анализа. Классификация структурных методологий.
8. Понятие предметной области (ПО) информационной системы. Анализ исходных данных. Моделирование ПО ИС. Необходимость и возможность формали-зованного представления ПО.
9. Методологии моделирования предметной области. Структурная модель предметной области. Цели и задачи предпроектной стадии создания ИС. Модели деятельности организации ("как есть" и "как должно быть").
10. Методология структурного моделирования SADT (IDEF0): состав функцио-нальной модели, построение иерархии диаграмм –требования, правила.

11. Типы функциональных связей: случайная, логическая, временная, процедур-ная, коммуникационная, последовательная, функциональная.
12. Понятие модели данных. Модели и проблемы человеко-машинного взаимодей-ствия в информационных системах. Концептуальные средства описания.
13. Модель сущность-связь (CASE-метод Баркера). Этапы построения модели. Ос-новные компоненты модели данных. Структуры данных. Операции над данны-ми. Ограничения целостности.
14. Основные и дополнительные конструкции. Типы связей. Методология постро-ения логической структуры данных.
15. Типы моделей данных. Реляционная модель данных. Структура данных. Огра-ничения целостности. Язык манипулирования данными. Нормализация отно-шений.
16. Иерархическая модель данных. Структуры данных. Ограничения целостности и манипулирование данными.
17. Сетевая модель данных.
18. Обзор графических средств представления проектных решений. Основы CASE-технологии. CASE-средства: обзор, классификация. Применение CASE-технологий на всех этапах жизненного цикла ИС.
19. Правовые, экономические, социальные и психологические аспекты информа-ционных систем. Тенденции и перспективы развития информационных систем.
20. Архитектура ИС в понятиях отечественных стандартов.
21. Нотация представления архитектурыстандарты открытой распределенной об-работки информации.
22. Zachman Framework-общая схема Захмана.
23. TOGAF(The Open Group Architectural Framework) - ОбщаясхемаорганизацииOpen Group.
24. FEAF(Federal Enterprise Architecture Framework) - общаясхемаФедеральнойархитектурыСША.
25. Метод ARIS.
26. Язык моделирования архитектуры ArchiMate. Назначение, возможности, осо-бенности использования.
27. Язык моделирования архитектуры ArchiMate. Моделирование бизнес-архитектуры.
28. Язык моделирования архитектуры ArchiMate. Моделирование архитектуры приложений.
29. Инструментарий моделирования CIM-UML.

Шкала оценивания

Уровень освоения компетенций по дисциплине «DevOps инжиниринг» определяется:

- знанием содержания процессов самоорганизации и самообразования, их особенностей и технологий реализации, исходя из целей совершенствования профессиональной деятельности;
- умением найти необходимую информацию, самостоятельно решать стандартные задачи профессиональной деятельности, выполнять действия в изученной последовательности, в том числе в новых условиях, на новом содержании;
- навыками использования современных информационных визуальных

цифровых технологий и способами их реализации;

- способностью самостоятельно строить процесс овладения информацией, отобранной и структурированной для выполнения профессиональной деятельности;
- умением обеспечивать сохранность здоровья себе и сотрудникам при выполнении профессиональных задач.

Низкий «неудовлетворительно/незачет» - компетенция не освоена или освоена в недостаточной мере. Студент не знает, либо знает на слабом уровне теоретический материал по дисциплине. Не владеет терминологией и основными понятиями из профессиональной сферы или называет неуверенно, с ошибками.

Пороговый (базовый) «удовлетворительно/зачет» - компетенция освоена удовлетворительно, но достаточно. Студент освоил основную базу теоретических знаний. Владеет терминологией и основными понятиями из профессиональной сферы.

Продвинутый «хорошо/зачет» - компетенция освоена достаточно хорошо. Студент знает теоретический материал по дисциплине, умеет применить эти знания на практике. Чётко и ясно формулирует свои мысли. Знает специальную и публицистическую литературу по профессиональным вопросам.

Высокий «отлично/зачет» - компетенция освоена в полной мере или на продвинутом уровне. Студент знает теоретический материал, умеет применить эти знания на практике и имеет опыт в профессионально-практической деятельности. Приводит актуальные примеры из сферы профессиональной деятельности; демонстрирует способности к нестандартной интерпретации поставленного вопроса.

Наименование темы (раздела)	**Показатели оценивания**	**Критерии оценивания**	**Оценка (баллы)**
DevOps и инфраструктурная инженерия.	Вопрос на зачете, Р	Зачет	Удовлетворительно -41 -60, хорошо 61-80, отлично 81-100, Не зачтено – 40-0
Системы управления версиями.	Вопрос на зачете, Р	Зачет	Удовлетворительно -41 -60, хорошо 61-80, отлично 81-100, Не зачтено – 40-0
Виртуализация.	Вопрос на зачете, Р	Зачет	Удовлетворительно -41 -60, хорошо 61-80, отлично 81-100, Не зачтено – 40-0
Администрирование баз данных.	Вопрос на зачете, Р	Зачет	Удовлетворительно -41 -60, хорошо 61-80, отлично 81-100, Не зачтено – 40-0
Мониторинг и логи.	Вопрос на зачете, выполнение КР	Зачет	Удовлетворительно -41 -60, хорошо 61-80, отлично 81-100, Не зачтено – 40-0
Система управления конфигурациями.	Вопрос на зачете, выполнение КР	Зачет	Удовлетворительно -41 -60, хорошо 61-80, отлично 81-100,

			Не зачтено – 40-0
Анализ частых множеств признаков и ассоциативных правил.	Вопрос на зачете, выполнение КР	Зачет	Удовлетворительно -41 -60, хорошо 61-80, отлично 81-100, Не зачтено – 40-0
Кластеризация данных.	Вопрос на зачете, выполнение КР	Зачет	Удовлетворительно -41 -60, хорошо 61-80, отлично 81-100, Не зачтено – 40-0

В начале каждого семестра до студентов доводится информация о структуре набора баллов за семестр:

Максимальный балл за посещение и активную работу в семестре – 60 баллов.

Максимальный балл за каждую контрольную работу – 7 баллов.

Максимальный балл за каждую решенную задачу – 7 баллов.

Максимальный балл за реферат по выбранной теме – 12 баллов.

Вес текущих контрольных работ различен и зависит от этапов формирования компетенций. Решение и постановка прикладных задач, формирующих компетенции, выше, чем простое владение информационными технологиями.

В результате каждый студент четко представляет свое число набранных баллов.

Текущая аттестация обучаемых. Текущая аттестация обучаемых по дисциплине «DevOps инжиниринг» проводится в соответствии с Уставом Академии, Положением о текущей аттестации студентов по программам ВО и является обязательной.

Текущая аттестация по дисциплине «DevOps инжиниринг» проводится в форме опроса и контрольных мероприятий по оцениванию фактических результатов обучения студентов и осуществляется ведущим преподавателем.

Объектами оценивания выступают:

учебная дисциплина (активность на занятиях, своевременность выполнения различных видов заданий, посещаемость практических занятий по аттестуемой дисциплине);

уровень овладения практическими умениями и навыками по всем видам учебной работы, уровень освоения компетенций;

результаты самостоятельной работы.

Активность обучаемого на занятиях оценивается по скорости и правильности выполнения практических заданий на компьютерах.

Кроме того, оценивание обучаемого проводится на контрольной неделе в соответствии с распоряжением проректора по учебной работе. Оценивание обучаемого на контрольной неделе проводится преподавателем независимо от наличия или отсутствия обучаемого (по уважительной или неуважительной причине) на занятии. Оценка носит комплексный характер и учитывает достижения обучаемого по основным компонентам учебного процесса за текущий период.

Оценивание обучаемого на занятиях осуществляется с использованием балльно-рейтинговой системы в соответствии с технологической картой дисциплины. Оценивание обучаемого на контрольной неделе также осуществляется по балльно-рейтинговой системе с выставлением оценок в ведомости и указанием количества пропущенных занятий.

4.3. Оценочные средства для промежуточной аттестации

4.3.1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы. Показатели и критерии оценивания компетенций с учетом этапа их формирования

Код компетенции	Наименование компетенции	Код этапа освоения компетенции	Наименование этапа освоения компетенции
ПКр – 2	Способность разрабатывать, внедрять и адаптировать прикладное программное обеспечение	ПКр-2.1 ПКр-2.2 ПКр-2.3	Знает принципы, методы и стандарты разработки архитектуры ИС Умеет профессионально выполнять разработку баз данных ИС Владеет навыками разработки архитектуры и баз данных ИС
ПКр-3	Способность проектировать ИС в соответствии с профилем подготовки по видам обеспечения	ПКр-3.1 ПКр-3.2. ПКр-3.3.	Знает основные методологии и средства проектирования и дизайна ИС Умеет выполнять проектирование и дизайн ИС Владеет навыками проектирования и разработки дизайна ИС
ПКр-4	способность документировать процессы создания информационных систем на стадиях жизненного цикла	ПКр-4.1 ПКр-4.2.	Знает принципы документирования существующих бизнес-процессов организации заказчика (реверс-инжиниринг бизнес-процессов организации) Умеет выполнять документирование существующих бизнес-

		ПКр-4.3.	процессов организации заказчика Владеет навыками создания пользовательской документации к ИС

ОТФ/ТФ *(при наличии профстандарта)*/ трудовые или профессиональные действия	Код этапа освоения компетенции	Результаты обучения

06.015 СПЕЦИАЛИСТ ПО ИНФОРМАЦИОННЫМ СИСТЕМАМ С Выполнение работ и управление работами по созданию (модификации) и сопровождению ИС, автоматизирующих задачи организационного управления и бизнес-процессы С/14.6 Разработка архитектуры ИС С/17.6 Разработка баз данных ИС	ПКр-2.1

06.015 СПЕЦИАЛИСТ ПО ИНФОРМАЦИОННЫМ СИСТЕМАМ С Выполнение работ и управление работами по созданию (модификации) и сопровождению ИС, автоматизирующих задачи организационного управления и бизнес-процессы С/16.6 Проектирование и дизайн ИС	ПКр-3.1 ПКр-3.2. ПКр-3.3.	ПКр-3.1.1 Знает основные методологии проектирования и дизайна ИС ПКр-3.1.2 Знает основные средства проектирования и дизайна ИС ПКр-3.2.1 Умеет применять основные методологии проектирования и дизайна ИС ПКр-3.2.2 Умеет применять основные средства проектирования и дизайна ИС ПКр-3.3.1 Владеет навыками проектирования ИС ПКр-3.3.2 Владеет навыками разработки дизайна ИС

06.015 СПЕЦИАЛИСТ ПО ИНФОРМАЦИОННЫМ СИСТЕМАМ С Выполнение работ и управление работами по созданию (модификации) и сопровождению ИС, автоматизирующих задачи организационного управления и бизнес-процессы С/07.6 Документирование существующих бизнес-процессов организации заказчика (реверс-инжиниринг бизнес-процессов организации) С/22.6 Создание пользовательской документации к ИС	ПКр-4.1 ПКр-4.2. ПКр-4.3.	ПКр-4.1.1 Знает правила и стандарты разработки пользовательской документации к ИС ПКр-4.1.2 Знает основные принципы документирования и реинжиниринга бизнес-процессов организации ПКр-4.2.1 Умеет выполнять документирование существующих бизнес-процессов организации заказчика ПКр-4.2.2 Умеет применять правила и стандарты разработки пользовательской документации к ИС ПКр-4.3.1 Владеет навыками создания пользовательской документации к ИС ПКр-4.3.2 Владеет навыками реинжиниринга бизнес-процессов организации

Промежуточная аттестация обучаемых по дисциплине «DevOps инжиниринг» проводится в соответствии с Уставом Академии, Положением о промежуточной аттестации студентов по программам ВО. Промежуточная аттестация по дисциплине «DevOps инжиниринг» проводится в соответствии с Учебным планом в 7 семестре в форме зачета. Обучаемые допускаются к зачету, экзамену по дисциплине в случае выполнения ими учебного плана по дисциплине: выполнения всех заданий и мероприятий, предусмотренных программой дисциплины по формам текущего контроля и выполненных рефератах. В случае наличия учебной задолженности обучаемый отрабатывает пропущенные занятия в соответствии с требованиями преподавателя. Экзамен и зачет принимает ведущий преподаватель.

Оценка знаний обучаемого на зачете складывается из:

ответа на один вопрос зачета;

выполнения всех текущих контрольных работ;

защиты рефератов по двум темам;

Оценка знаний обучаемого на зачете носит комплексный характер, является балльной и определяется его:

- ответом на зачете;
- учебными достижениями в семестровый период.

Знания, умения, навыки обучаемого на зачете оцениваются оценками: «отлично» – 5, «хорошо» – 4, «удовлетворительно» – 3, «неудовлетворительно» – 2. Кроме того, обучаемому выставляется оценка в соответствии с балльно-рейтинговой системой. Основой для определения оценки служит уровень усвоения обучаемыми материала, предусмотренного данной рабочей программой.

4.3.2 Типовые оценочные средства

Темы коллоквиумов по дисциплине «DevOps инжиниринг»

Коллоквиум организуется для проверки самостоятельной работы по теме, предложенную преподавателем (обучаемым). Коллоквиум позволяет автору научиться четко и грамотно формулировать мысли, структурировать информацию, использовать основные категории анализа, выделять причинно-следственные связи, иллюстрировать понятия соответствующими примерами, аргументировать свои выводы; овладеть научным стилем речи. Коллоквиум содержит: четкое изложение сути поставленной проблемы, включать самостоятельно проведенный анализ этой проблемы с использованием концепций и аналитического инструментария, рассматриваемого в рамках дисциплины, выводы, обобщающие авторскую позицию по поставленной проблеме.

1. Задачи DevOps-инженера сейчас — это обеспечение комфортного и эффективно выстроенного процесса взаимодействия команд разработки и инфраструктуры продукта. В чем трудности?
2. Эксплуатация программного кода.
3. Зачем необходимо знать обо всех элементах системы, конфигурациях, потенциальных точках нестабильности и т. п?
4. Что тпкое методология ITIL и ему подобные методологии.
5. Умение пользоваться существующими решениями.
6. Разница компетенций разработчиков команды.
7. Процессы и объекты девопс.
8. Цели и задачи DevOps.
9. Главные принципы DevOps.
10. Достоинства девопс.
11. Критика и недостатки DevOps.
12. Методы и средства реализации: как работает DevOps.
13. Кто и где используется DevOps и сколько за это стоит.

Критерии оценки коллоквиумов:

Оценка «удовлетворительно» предполагает, что полученные результаты в значительной степени соответствуют поставленной цели (цель работы достигнута в основном). Обоснована актуальность работы. В процессе анализа литературы отобраны наиболее важные источники, продемонстрировано понимание решаемой проблемы. Выбраны адекватные цели научный подход, методы, процедуры. Они в значительной степени реализованы в работе. Выводы имеют наглядный и проверяемый характер. Требования по оформлению работы в основном выполнены.

Оценка «хорошо» ставится, когда полученные результаты преимущественносоответствуют поставленной цели и задачам. Обоснована практическая и теоретическая актуальность работы. В процессе анализа литературы отобран и проанализирован широкий круг теоретических и эмпирических источников. Выбраны и обоснованы применяемые научные подходы, методы и

процедуры. Полученные результаты в целом логичны, доказательны и систематизированы. Оформление работы в целом соответствует существующим требованиям.

Оценка «отлично» предполагает: полученные результаты полностью соответствуют поставленной цели. Обоснована практическая и теоретическая значимость работы. Проведен детальный анализ теоретических и эмпирических источников, выводы автора самостоятельны и аргументированы. Выбраны и подробно описаны применяемые в работе научные подходы, методы и процедуры. Содержание работы полностью отражает узловые проблемы темы, исследовательская часть (в курсовой работе) выполнена самостоятельно, методологически корректно и содержит достоверные и интересные выводы и положения. Оформление работы полностью отвечает всем требованиям

Вопросы к зачету по дисциплине «DevOps инжиниринг»

1. Простые эвристические подходы.
2. Алгоритм K-Means.
3. Проблема устойчивости результатов и важность грамотной инициализации, алгоритм K-Means++.
4. Выбор числа кластеров.
5. Оценка качества кластеризации.
6. Системы оповещения о проблемах ELK: Elasticsearch, Logstash и Kibana. Graylog. Zabbix. Prometehus, Grafana, Alertmanager.
7. Практика использования Ansible
8. Понятие информационной системы (ИС). Свойства ИС. Качество ИС. Параметры качества ИС. Требования, предъявляемые к информационным системам.
9. Классификация информационных систем.
10. Принципы построения информационных систем.
11. Типовые функциональные компоненты информационной системы.
12. Понятие архитектуры системы. Определение архитектуры ИС согласно стандартов ANSI\IEEE и ГОСТ. Типовые функциональные компоненты ИС. Уров-ни описания архитектуры ИС.
13. Архитектурный подход к реализации информационных систем: понятия и определения.
14. Методология «архитектуры предприятия». Переход от бизнес-архитектуры к ИТ-архитектуре.
15. Архитектуры аппаратных средств вычислительных систем. Четыре класса структур аппаратных средств ЭВМ.
16. Централизованная архитектура ИС.
17. Файл-серверная архитектура ИС. Модель файлового сервера.
18. Клиент-серверная архитектура ИС. Модель сервера СУБД.
19. Переходная к трехслойной архитектуре (2.5 слоя). Трехуровневая клиент-серверная архитектура. Модель сервера приложений.
20. Архитектура на основе Internet/Intranet
21. Специализированные подсистемы (СУБД, SAN и т. д.). Основные функции СУБД. Классификации СУБД.
22. Распределенные информационные системы.
23. Сервис–ориентированная архитектура (SOA) информационных систем.

24. Функциональные уровни информационной системы. Основные критерии де-композиции. Декомпозиция информационных систем на слои и уровни. Выде-ление подсистем в архитектуре.
25. Понятие интерфейса. Интерфейс ИС и интерфейс пользователя. Виды пользовательского интерфейса.
26. Типы интерфейсов. Методы и средства разработки пользовательского интерфейса. Стандартизация.
27. Классы информационных систем.
28. Корпоративные информационные системы.
29. Облачные технологии для ИС.
30. Архитектурные и проектные решения для интеграцииразличных информационных систем между собой. Интерфейсы и протоколы обмена данными.
31. Архитектуры масштабируемых информационных систем. Параллельные ин-формационные системы.
32. Жизненный цикл информационных систем. Понятие жизненного цикла ИС. Процессы жизненного цикла: основные, вспомогательные, организационные.
33. Модели ЖЦ ИС.
34. Проектирование архитектур информационных систем. Установление требова-ний.
35. Проектирование архитектур информационных систем. Спецификация требова-ний.
36. Методы и модели моделирования архитектуры ИС. Обзор существующих под-ходов. Стандарт IEEE 1471. Современные методологии моделирования архи-тектуры ИС.
37. Модель Захмана. Общие принципы модели Захмана. Интеграция различных моделей в рамках матрицы Захмана.
38. Методология TOGAF. Общие принципы TOGAF. Фазы процесса моделирова-ния. Состав моделей TOGAF.
39. Обзор графических средств представления проектных решений. Основы CASE-технологии. CASE-средства: обзор, классификация.
40. Применение CASE-технологий на всех этапах жизненного цикла ИС.
41. Правовые, экономические, социальные и психологические аспекты информационных систем.
42. Тенденции и перспективы развития информационных

Критерии оценки компетенций

Промежуточная аттестация обучаемых по дисциплине «DevOps инжиниринг» проводится в соответствии с Уставом Академии, Положением о промежуточной аттестации студентов по программам ВПО. Промежуточная аттестация по дисциплине «DevOps инжиниринг» проводится в соответствии с Учебным планом в 7 семестре в форме зачета. Обучаемые допускаются к зачету по дисциплине в случае выполнения ими учебного плана по дисциплине: выполнения всех заданий и мероприятий, предусмотренных программой дисциплины по формам текущего контроля и выполненных заданий. В случае наличия учебной задолженности обучаемый отрабатывает пропущенные занятия в соответствии с требованиями преподавателя. Оценка знаний обучаемого на зачете носит комплексный характер, является балльной и определяется его:

- ответом на зачете;
- учебными достижениями в семестровый период.

Знания, умения, навыки студента на экзамене оцениваются оценками: «отлично» – 5, «хорошо» – 4 , «удовлетворительно» – 3, «неудовлетворительно» – 2. Кроме того, обучаемому выставляется оценка в соответствии с балльно-рейтинговой системой. Основой для определения оценки служит уровень усвоения обучаемыми материала, предусмотренного данной рабочей программой.

Темы рефератов по дисциплине «DevOps инжиниринг»

1. В чем состоит суть DevOps-культуры?
2. Зачем нужен DevOps?
3. Какие бывают DevOps-практики?
4. Как соотносятся понятия «системный инженер», «билд-инженер» и «DevOps-инженер»?
5. Что должен знать DevOps-инженер?
6. Особенности реализации информационных систем в различных предметных областях.
7. Структурный системный анализ. Методы структурного анализа. Классификация структурных методологий.
8. Понятие предметной области (ПО) информационной системы. Анализ исходных данных. Моделирование ПО ИС. Необходимость и возможность формали-зованного представления ПО.
9. Методологии моделирования предметной области. Структурная модель предметной области. Цели и задачи предпроектной стадии создания ИС. Модели деятельности организации ("как есть" и "как должно быть").
10. Методология структурного моделирования SADT (IDEF0): состав функцио-нальной модели, построение иерархии диаграмм –требования, правила.
11. Типы функциональных связей: случайная, логическая, временная, процедур-ная, коммуникационная, последовательная, функциональная.
12. Понятие модели данных. Модели и проблемы человеко-машинного взаимодей-ствия в информационных системах. Концептуальные средства описания.
13. Модель сущность-связь (CASE-метод Баркера). Этапы построения модели. Ос-новные компоненты модели данных. Структуры данных. Операции над данны-ми. Ограничения целостности.
14. Основные и дополнительные конструкции. Типы связей. Методология постро-ения логической структуры данных.
15. Типы моделей данных. Реляционная модель данных. Структура данных. Огра-ничения целостности. Язык манипулирования данными. Нормализация отно-шений.
16. Иерархическая модель данных. Структуры данных. Ограничения целостности и манипулирование данными.
17. Сетевая модель данных.
18. Обзор графических средств представления проектных решений. Основы CASE-технологии. CASE-средства: обзор, классификация. Применение CASE-технологий на всех этапах жизненного цикла ИС.
19. Правовые, экономические, социальные и психологические аспекты информационных систем. Тенденции и перспективы развития информационных систем.
20. Архитектура ИС в понятиях отечественных стандартов.
21. Нотация представления архитектурыстандарты открытой распределенной обработки информации.

22. Zachman Framework-общая схема Захмана.
23. TOGAF(The Open Group Architectural Framework) - ОбщаясхемаорганизацииOpen Group.
24. FEAF(Federal Enterprise Architecture Framework) - общаясхемаФедеральнойархитектурыСША.
25. Метод ARIS.
26. Язык моделирования архитектуры ArchiMate. Назначение, возможности, осо-бенности использования.
27. Язык моделирования архитектуры ArchiMate. Моделирование бизнес-архитектуры.
28. Язык моделирования архитектуры ArchiMate. Моделирование архитектуры приложений.
29. Инструментарий моделирования СИМ-UML.

Критерии оценки рефератов:

Оценка «удовлетворительно» предполагает, что полученные результаты в значительной степени соответствуют поставленной цели (цель работы достигнута в основном). Обоснована актуальность работы. В процессе анализа литературы отобраны наиболее важные источники, продемонстрировано понимание решаемой проблемы. Выбраны адекватные цели научный подход, методы, процедуры. Они в значительной степени реализованы в работе. Выводы имеют наглядный и проверяемый характер. Требования по оформлению работы в основном выполнены.

Оценка «хорошо» ставится, когда полученные результаты преимущественносоответствуют поставленной цели и задачам. Обоснована практическая и теоретическая актуальность работы. В процессе анализа литературы отобран и проанализирован широкий круг теоретических и эмпирических источников. Выбраны и обоснованы применяемые научные подходы, методы и процедуры. Полученные результаты в целом логичны, доказательны и систематизированы. Оформление работы в целом соответствует существующим требованиям.

Оценка «отлично» предполагает: полученные результаты полностью соответствуют поставленной цели. Обоснована практическая и теоретическая значимость работы. Проведен детальный анализ теоретических и эмпирических источников, выводы автора самостоятельны и аргументированы. Выбраны и подробно описаны применяемые в работе научные подходы, методы и процедуры. Содержание работы полностью отражает узловые проблемы темы, исследовательская часть (в курсовой работе) выполнена самостоятельно, методологически корректно и содержит достоверные и интересные выводы и положения. Оформление работы полностью отвечает всем требованиям

Шкала оценивания по результатам зачета:

Планируемые результаты обучения (показатели достижения заданного уровня освоения компетенций)	Критерии оценивания результатов обучения			
	неудовлетворительно	удовлетворительно	хорошо	отлично

ПКр-2. Способность разрабатывать, внедрять и адаптировать прикладное программное обеспечение				
на уровне знаний*:* Знает особенности всех этапов жизненного цикла информационн ой системы Знает основные стандарты оформления технической документации для информационн ой системы Знает перечень стандартной технической документации для каждого этапа жизненного цикла информационн ой системы.	Не знает возможности применения основных законов естественнонауч ных дисциплин, а также современных информационно-коммуникационн ых технологий	Демонстрирует частичные знания основных законов естественнонауч н ых дисциплин, а также современных информационно-коммуникационн ых технологий для решения комплекса задач на основе информационно й и библиографичес кой культуры	Демонстрирует системные знания основных законов естественнонауч ных дисциплин, а также современных информацион но-коммуникационн ых технологий для решения стандартных задач профессиональн ой деятельности на основе информационно й и библиографичес кой культуры	Проявляет высокий уровень знаний всего учебного материала
на уровне умений *(типовые действия выполняются по заданному алгоритму)*: Умеет применять стандарты оформления технической документации на ИС	Не умеет применять знания на практике для разработки алгоритмов решения задач	Демонстрирует частичный уровень умений выявлять экономическую сущность задач	Демонстрируе т достаточный уровень умений выявлять экономическу ю сущность задач, использовать экономико-математически й аппарат для разработки алгоритмов решения задач	Демонстр ируе т высокий уровень умений для разработк и алгоритмо в решения задач
на уровне навыков *(типовые действия*	Не владеет практическими навыками обработки текстовой,	Владеет практическими навыками работы на компьютере и	Владеет практическим и навыками работы на компьютере и	Владеет практичес ким и навыками

выполняются автоматически, без воспроизведения алгоритма): Владеет навыками оформления технической документации для информационной системы. Владеет навыками составления технической документации на различных этапах жизненного цикла информационной системы в соответствии с действующими стандартами	табличной и графической информации	использования стандартных информационно-коммуникационных технологий частично	использования стандартных информационно-коммуникационных технологий достаточно свободно	работы на компьютере и использования стандартных информационно-коммуникационных технологий на высоком уровне
ПКр – 4. способность документировать процессы создания информационных систем на стадиях жизненного цикла				
на уровне знаний: принципов разработки планов коммуникаций с заказчиком в проектах создания (модификации) ИС; Знает принципы, методы и стандарты разработки архитектуры ИС; Знает основные	Не знает как проводить описание прикладных процессов и информационного обеспечения решения разработки планов коммуникаций с заказчиком в проектах создания (модификации) ИС; методологии и средства проектировани	Демонстрирует частичные знания в описании прикладных процессов и информационного обеспечения решения разработки планов коммуникаций с заказчиком в проектах создания (модификации) ИС; методологии и средства	Демонстрирует системные знания в описании прикладных процессов и информационного обеспечения решения разработки планов коммуникаций с заказчиком в проектах создания (модификации) ИС; методологии и средства	Проявляет высокий уровень знаний всего учебного материала

методологии и средства проектирования и дизайна ИС; Знает основные методологии и средства разработки баз данных ИС; Знает правила и стандарты разработки пользовательской документации к ИС; Знает принципы и приемы командообразования.	я и дизайна ИС; правила и стандарты разработки пользовательской документации к ИС; принципы и приемы командообразования.	проектирования и дизайна ИС; правила и стандарты разработки пользовательской документации к ИС; принципы и приемы командообразования.	проектирования и дизайна ИС; правила и стандарты разработки пользовательской документации к ИС; принципы и приемы командообразования.	
на уровне умений: Умеет выполнять документирование существующих бизнес-процессов организации заказчика; Умеет применять принципы, методы и стандарты разработки архитектуры ИС; Умеет профессионально выполнять разработку баз данных ИС; Умеет организовывать программы	Не умеет применять знания на практике	Демонстрирует частичный уровень умений	Демонстрирует достаточный уровень умений	Демонстрирует высокий уровень умений для разработки алгоритмов решения задач

обучения и развития персонала.				
на уровне навыков: Владеет навыками выстраивания коммуникаций с заказчиком в проектах создания (модификации) и ввода ИС в эксплуатацию; Владеет навыками документирования и реинжиниринга бизнес-процессов организации; Владеет навыками проектирования и разработки дизайна ИС; Владеет навыками создания пользовательской документации к ИС; Владеет навыками организации заключения договоров на выполняемые работы, связанные с ИС	Не владеет практическими навыками работы с технологиями и программным инструментарие м	Владеет практическими навыками частично работы с технологиями и программным инструментарием	Владеет практическими навыками формирования требований к информационной системе достаточно свободно	Владеет практическим и навыками работы на компьютере и использования стандартных информационно-коммуникационных технологий на высоком уровне

4.4. Методические материалы

4.4.1. Методические материалы, определяющие процедуру оценивания ответов обучающихся на вопросы на понимание лекционного материала

Критериями оценки ответа обучающихся на лекционном занятии выступают:

- правильность ответов на вопросы преподавателя по изученному

- материалу;
- полнота и лаконичность ответа;
- степень понимания тематики предмета;
- логика и аргументированность изложения материала;
- приведение примеров, демонстрирующих умение и владение
- полученными знаниями по темам предмета в раскрытии поставленных вопросов.

4.4.2. Методические материалы, определяющие процедуру оценивания при проведении опроса на практическом занятии

Оценки "отлично" заслуживает обучающийся, обнаруживший всестороннее, систематическое и глубокое знание вопроса, умение свободно ориентироваться в теме, усвоивший основную, и знакомый с дополнительной литературой, рекомендованной программой. Как правило, оценка "отлично" выставляется обучающимся, усвоившим взаимосвязь основных понятий в их значении для приобретаемой профессии, проявившим творческие способности в понимании, изложении и использовании учебно-программного материала;

Оценки "хорошо" заслуживает обучающийся, обнаруживший полное знание темы, успешно выполняющий предусмотренные программой задания, усвоивший основную литературу, рекомендованную в программе. Как правило, оценка "хорошо" выставляется обучающимся, показавшим систематический характер знаний по пройденному материалу и способным к их самостоятельному пополнению и обновлению в ходе дальнейшей учебы и профессиональной деятельности;

Оценки "удовлетворительно" заслуживает обучающийся, обнаруживший знание основного материала в объеме, необходимом для дальнейшего усвоения материала и предстоящей работы по профессии, знакомый с основной литературой, рекомендованной программой.

Оценка "неудовлетворительно" выставляется обучающемуся, обнаружившему пробелы в знаниях основного материала темы, допустившему принципиальные ошибки в понимании и изложении учебного материала.

4.4.3. Методические материалы, определяющие процедуру оценивания промежуточной аттестации по дисциплине

Зачет проводится путем распределения по времени в течение всего срока обучения проведения мониторинга уровня освоения компетенции по результатам выполнения заданий по подготовке к опросу. При условии положительных ответов на вопросы при зачете, компетенции считаются сформированными, и по данной дисциплине выставляется оценка. В ином случае, компетенции считаются несформированными, и по данной дисциплине выставляется оценка «не зачтено».

Зачет принимается в устной форме, по вопросам. Задание включает теоретический и практический вопрос. Оценка знаний обучающегося на зачете носит комплексный характер и определяется его:

- ответом на зачете;
- учебными достижениями в семестровый период.

Знания, умения, навыки обучающегося на экзамене оцениваются оценками: «отлично», «хорошо», «удовлетворительно», «неудовлетворительно». Основой для определения оценки служит уровень усвоения обучающимися материала, предусмотренного данной рабочей программой.

Оценивание студента на зачете по дисциплине «DevOps инжиниринг»

Оценка	Требования к знаниям
Отлично	Оценка «отлично» выставляется обучающемуся, если он глубоко и прочно усвоил программный материал, исчерпывающе, последовательно, четко и логически стройно его излагает на экзамене, умеет тесно увязывать теорию с практикой, свободно справляется с задачами, вопросами и другими видами применения знаний, причем не затрудняется с ответом при видоизменении заданий, правильно обосновывает принятое решение. Учебные достижения в семестровый период демонстрируют высокую степень овладения программным материалом.
Хорошо	Оценка «хорошо» выставляется обучающемуся, если он твердо знает материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, правильно применяет теоретические положения при решении практических вопросов и задач, владеет необходимыми навыками и приемами их выполнения. Учебные достижения в семестровый период демонстрируют хорошую степень овладения программным материалом.
Удовлетворительно	Оценка «удовлетворительно» выставляется обучающемуся, если он имеет знания только основного материала, но не усвоил его деталей, допускает неточности, недостаточно правильные формулировки, нарушения логической последовательности в изложении программного материала, испытывает затруднения при выполнении практических работ. Учебные достижения в семестровый период демонстрируют достаточную (удовлетворительную) степень овладения программным материалом.
Неудовлетворительно	Оценка «неудовлетворительно» выставляется обучающемуся, который не знает значительной части программного материала, допускает существенные ошибки, неуверенно, с большими затруднениями выполняет практические работы. Как правило, оценка «неудовлетворительно» ставится обучающимся, которые не могут продолжить обучение без дополнительных занятий по соответствующей дисциплине. Учебные достижения в семестровый период не демонстрировали достаточную степень овладения программным материалом на пороговом уровне.

Зачет проводится во время экзаменационной сессии по традиционной шкале оценивания.
Студентам, не выполнившим требования по промежуточной аттестации, по уважительным причинам, предоставляется возможность их выполнения и сдачи.

5. Методические указания для обучающихся по освоению дисциплины

Домашние задания соответствуют приведенным ранее темам для коллоквиумов и опросов при самостоятельной подготовке, и могут быть опубликованы:
1. На общих ресурсах академии (в сетевой папке);
2. Высланы на общую почту группы

Конкретный способ коммуникации со студентами определяется преподавателем. Для подготовки к практическим занятиям необходимо ознакомиться с лекциями, рекомендованной литературой, иными источниками, интернет ресурсами, повторить материал предыдущих практических занятий. Самостоятельно разобрать вопросы к практическим занятиям по темам. При возникновении вопросов - обратиться к преподавателю по электронной почте с указанием конкретной проблемы и (или) прийти к преподавателю на консультацию в установленное время.

5.1. Методические указания по вопросам на понимание лекционного материала

На лекциях рекомендуется слушать предлагаемый лектором материал, при этом параллельно конспектировать основные положения, поскольку это дает наибольший результат в усвоении материала. Предоставляется возможность задавать вопросы на уточнение понимания темы и принимать участие в ее обсуждении. Кроме этого, для лучшего освоения материала и систематизации знаний по дисциплине, необходимо постоянно разбирать материалы лекций по конспектам и учебным пособиям. Во время самостоятельной проработки лекционного материала особое внимание следует уделять возникшим вопросам, непонятным терминам, спорным точкам зрения. Все такие моменты следует выделить или выписать отдельно для дальнейшего обсуждения на семинарском занятии. В случае необходимости обращаться к преподавателю за консультацией. Полный список литературы по дисциплине приведен в разделе 6 программы.

5.2. Методические указания по подготовке вопросов для самостоятельного изучения к занятиям практического (семинарского) типов

Подготовка обучающегося к практическому занятию осуществляется на основании плана раскрытия темы практического занятия, которое разрабатывается преподавателем на основе рабочей программы и своевременно доводится до сведения обучающегося.

При подготовке к практическому занятию, обучающемуся необходимо внимательно изучить основные вопросы темы семинара. Важным условием успешной подготовки к практическому занятию является четкая организация самостоятельной работы студентов по изучению учебной и дополнительной литературы. Умение анализировать и применять для ответов на вопросы и решения задач и заданий полученные знания при самостоятельной подготовке в значительной степени определяет успешность освоения материала по дисциплине и формирование у обучающихся соответствующих компетенций.

Подготовка вопросов для самостоятельного изучения включает: изучение необходимой литературы (обязательной, дополнительной литературы, специальных периодических изданий, Интернет-ресурсов), подготовку конспекта ответа, ответы на вопросы.

При подготовке к практическим занятиям важно:

- использовать достаточно широкий диапазон массива информации, провести обзор литературы и специальных изданий, составить каталог Интернет-ресурсов;
- представить различные подходы, четко и полно определить рассматриваемые понятия, выявить взаимосвязи понятий и явлений, взаимозависимости и связи с другими вопросами;
- грамотно структурировать материал, ясно, четко и логично его излагать, приводить соответствующие примеры из практики, для иллюстрации положений, тезисов и выводов использовать таблицы, схемы, графики, диаграммы.

Вопросы для самостоятельной подготовки к занятиям практического (семинарского) типа указаны в разделе 4.2.

5.3. Методические рекомендации по подготовке к зачету по дисциплине

Ответ на экзамене предусматривает устный ответ на теоретические вопросы. При подготовке к зачету обучающийся обращается к пройденному материалу, сосредоточенному в конспектах лекций, учебниках и других источниках информации.

Повторяя, обобщая, закрепляя и дополняя полученные знания, поднимает их на качественно-новый уровень — уровень системы совокупных данных, что позволяет ему понять логику всего предмета в целом. Новые знания обучающийся получает в ходе самостоятельного изучения того, что не было изложено в лекциях и на семинарских занятиях.

Зачет как особая форма учебного процесса имеет свои особенности, специфические черты и некоторые аспекты, которые необходимо обучающемуся знать и учитывать в своей работе. Это, прежде всего:

• что и как запоминать при подготовке к зачету;
• по каким источникам и как готовиться;
• на чем сосредоточить основное внимание;
• каким образом в максимальной степени использовать программу курса;
• что и как записать, а что выучить дословно и т. п.

На зачете, как правило, проверяется не столько уровень запоминания обучающимся учебного материала, сколько то, насколько успешно он оперирует теми или иными научными понятиями и категориями, систематизирует факты, как умеет мыслить, аргументировано отстаивать определенную позицию, объясняет и пересказывает заученную информацию.

Программу курса необходимо максимально использовать как в ходе подготовки, так и на самом зачете. Ведь она включает в себя разделы, темы и основные проблемы, в рамках которых и формируются вопросы для зачета.

Оптимальным для подготовки к экзамену является вариант, когда обучающийся начинает подготовку к нему с первых занятий по данному курсу.

При подготовке к экзамену по наиболее сложным вопросам, ключевым проблемам и важнейшим понятиям необходимо сделать краткие письменные записи в виде тезисов, планов, определений. Особое внимание в ходе подготовки к экзамену следует уделять конспектам лекций, ибо они обладают рядом преимуществ по сравнению с печатной продукцией. Как правило, они более детальные, иллюстрированные, что позволяет оценивать современную ситуацию, отражать самую свежую научную и оперативную информацию, отвечать на вопросы, интересующие аудиторию, в данный момент, тогда как при написании и опубликовании печатной продукции проходит определенное время, и материал быстро устаревает.

В то же время, подготовка по одним конспектам лекций недостаточна, необходимо использовать и иную учебную литературу. Не следует бояться дополнительных и уточняющих вопросов на экзамене. Они, как правило, задаются или помимо экзаменационного вопроса для выявления общей подготовленности, или в рамках билета для уточнения высказанной мысли.

6. Учебная литература и ресурсы информационно-телекоммуникационной сети "Интернет", включая перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

6.1. Основная литература

1. Гарнаев А.Ю., Гарнаев С.Ю. Web-программирование на Java и JavaScript. – СПб.: БХВ-Петербург, 2018. – 1040 с.
2. Котеров Д.В., Костарев А.Ф. PHP 5. - СПб.: БХВ-Петербург, 2019. – 1120 с.
3. Костин С.П. Самоучитель создания Web-сайтов. – М.: Триумф, 2018. - 176с.
4. Дейтел Х.М., Дейтел П.Дж., Нието Т.Р. Как программировать для Internet & WWW. – М.: Бином, 2018. – 1178.
5. Питер Морвиль, Луис Розенфельд. Информационная архитектура в Интернете. – М.: Символ-Плюс, 2019. – 608 с.

6.2. Дополнительная литература

1. Gene Kim The Phoenix Project: A Novel About IT, DevOps, and Helping Your Business Win / Gene Kim, Kevin Behr, George Spafford. - М.: IT Revolution Press, 2019. - 346 с.
2. Jesse Russell DevOps / Jesse Russell. - М.: Книга по Требованию, 2019. - 103 с.
3. Kyle Rankin DevOps Troubleshooting: Linux Server Best Practices / Kyle Rankin. - Москва: Гостехиздат, 2018. - 240 с.
4. Paul Swartout Continuous Delivery and DevOps / Paul Swartout. - М.: Книга по Требованию, 2018. - 154 с.
5. Sanjeev Sharma The DevOps Adoption Playbook. A Guide to Adopting DevOps in a Multi-Speed IT Enterprise / Sanjeev Sharma. - Москва: Огни, 2016. - 880 с.
6. Ноа Гифт и Джереми М. Джонс. Python в системном администрировании UNIX и Linux – Пер. с англ. – СПб.: Символ□Плюс, 2019. – 512 с., ил.

6.3. Учебно-методическое обеспечение самостоятельной работы

6. Положение об организации самостоятельной работы студентов федерального государственного бюджетного образовательного учреждения высшего образования «Российская академия народного хозяйства и государственной службы при Президенте Российской Федерации» (в ред. приказа РАНХиГС от 11.05.2016 г. № 01-2211). http://www.ranepa.ru/images/docs/prikazy-ranhigs/Pologenie_o_samostoyatelnoi_rabote.pdf

Самостоятельная работа студентом осуществляется для закрепления изученного материала после практических занятий или лабораторных работ, для выполнения домашних заданий, для подготовки к контрольным работам, для изучения дополнительных материалов.

№ пп	Тип занятия	Указания
	Тема 1	DevOps и инфраструктурная инженерия.
1	СРС	Знакомство с основной терминологие. Различие DevOps и DevSecO. Цели методологии DevOps.
2	СРС	Получить общее представление об инфраструктуре современной разработки. Обзор используемых инструментов. Циклы и этапы разработки ПО. Гибкие методологии разработки: Agile, Scrum, Kanban.
3	СРС	Для подготовки к занятиям использовать лекции, интернет ресурсы, основную и дополнительную литературу.
	Тема 2	Системы управления версиями.

1	СРС	Изучить системы управления версиями исходного кода. Получите практические навыки глубокой работы с Git репозиториями.
2	СРС	Научиться работать одновременно с несколькими репозиториями, синхронизировать их и создавать резервные копии. Познакомитесь со способами интеграции репозиториев со внешними системами.
3	СРС	Для подготовки к занятиям использовать лекции, интернет ресурсы, основную и дополнительную литературу.
	Тема 3	Виртуализация.
1	СРС	Научиться контролировать процесс разработки ещё на уровне отправки кода в репозиторий. Обзор систем управления версиями. Подробное познакомиться с Git. Установка Gitlab на собственный сервер.
2	СРС	Научитесь управлять виртуальными машинами с помощью libvirtd. Напишете несколько Dockerfile, которые можно будет использовать в дальнейших проектах как примеры.
3	СРС	Для подготовки к занятиям использовать лекции, интернет ресурсы, основную и дополнительную литературу
	Тема 4	Администрирование баз данных.
1	СРС	Познакомитесь с решениями для полнотекстового поиска. Овладеете навыками установки и настройки кеш-систем. Научитесь устанавливать и настраивать базы данных для нужд разработки. Познакомитесь с технологиями создания отказоустойчивых кластеров баз данных и кеш систем при помощи кластеризации и шардинга.
2	СРС	Научитесь писать простые SQL-запросы и запросы для работы с NoSQL-базами данных. Типы и структура СУБД. Администрирование PostgreSQL. Администрирование MySQL. Знакомство с NoSQL-базами данных на примере MongoDB. Кеш-системы Redis и Memcached. Elasticsearch. Кластеризация, шардинг и отказоустойчивость.
3	СРС	Для подготовки к занятиям использовать лекции, интернет ресурсы, основную и дополнительную литературу
	Тема 5	Мониторинг и логи.
1	СРС	Научиться организовывать логирование всех действий приложений и анализировать эти логи. Овладеете навыками работы с elasticsearch, Logstash, Kibana и Graylog.
2	СРС	Для подготовки к занятиям использовать лекции, интернет ресурсы, основную и дополнительную литературу
	Тема 6	Система управления конфигурациями.
1	СРС	Научиться настраивать связку Prometehus + Grafana + Alertmanager. Познакомиться с Zabbix для мониторинга физических и виртуальных машин.
2	СРС	Научиться описывать инфраструктуру в виде кода. Сможете настроить удаленный сервер и восстановить его конфигурацию в случае необходимости. Узнать набор уже готовых шаблонов для решения типовых задач конфигурирования серверов.

3	СРС	Для подготовки к занятиям использовать лекции, интернет ресурсы, основную и дополнительную литературу
	Тема 7	Анализ частых множеств признаков и ассоциативных правил.
1	СРС	Знать концепцию «инфраструктура как код». Знакомство с Ansible. Практика использования Ansible
2	СРС	Алгоритм Априори. Меры «интересности правил».
3	СРС	Для подготовки к занятиям использовать лекции, интернет ресурсы, основную и дополнительную литературу
	Тема 8	Кластеризация данных
1	СРС	Простые эвристические подходы. Алгоритм K-Means.
2	СРС	Проблема устойчивости результатов и важность грамотной инициализации, алгоритм K-Means++.
3	СРС	Для подготовки к занятиям использовать лекции, интернет ресурсы, основную и дополнительную литературу

Рекомендации по организации самостоятельной работы студентов:

Вид учебных занятий	Организация деятельности студента
Лекция	Написание конспекта лекций: кратко, схематично, последовательно фиксировать основные положения, выводы, формулировки, обобщения; помечать важные мысли, выделять ключевые слова, термины. Проверка терминов, понятий с помощью энциклопедий, словарей, справочников с выписыванием толкований в тетрадь. Обозначить вопросы, термины, материал, который вызывает трудности, пометить и попытаться найти ответ в рекомендуемой литературе. Если самостоятельно не удается разобраться в материале, необходимо сформулировать вопрос и задать преподавателю на консультации, на практическом занятии.
Практические занятия	Проработка рабочей программы, уделяя особое внимание целям и задачам, структуре и содержанию дисциплины. Анализ литературы и/или Интернет – ресурсов, решение типовых задач, проведение консультаций с преподавателем по проблемным вопросам лекционного материала, подготовка к опросу и/или коллоквиуму, постановка задачи для практического задания, оформление отчета по заданию, работа в малой группе.
Контрольная работа / индивидуальное задание	Ознакомление с постановкой задачи, разработка алгоритма решения, выбор программно-аппаратных средств реализации. Реализация в выбранной инструментальной среде. Подготовка выводов и/или короткого отчета в виде презентации.
Подготовка к дискуссиям, докладам, написанию реферата	Поиск литературы и составление библиографии, использование от 3 до 5 научных работ, изложение мнения авторов и своего суждения по выбранному вопросу; изложение основных аспектов проблемы. Изучение научной, учебной, нормативной и другой литературы. Отбор необходимого материала; формирование выводов

	и разработка конкретных алгоритмов и/или рекомендаций по решению поставленной цели и задачи; проведение практических исследований по данной теме. Разработка презентации для доклада основных результатов проведенной работы. Желательно оформление результатов в виде статьи в сборник трудов конференции или тематический научный журнал.

6.4. Нормативные правовые документы

7. IEEE P1003.0 «Руководство по окружению открытых систем POSIX».
8. ГОСТ Р ИСО/МЭК 12207. Процессы жизненного цикла программных средств.

6.5. Интернет-ресурсы, справочные системы.

9. http://smarty.php.net – веб-технологии
10. http://yandex.ru/cgi/vesna - примеры скриптов веб-приложений
11. http://belitsoft.ru/ - инструментарий разработки интернет-приложений
12. http://www.virtech.ru – инструментарий разработки интернет-приложений
13. http://www.oracle.sun.com – инструментарий разработки интернет-приложений
14. 1. http://www.consultant.ru/ - Официальный сайт компании «Консультант Плюс».
15. http://www.cfin.ru/software/index.shtml - Сайт «Корпоративный менеджмент». Обзор информационных систем для основных функций управления и бизнеса.
16. http://citforum.ru/ - «Сервер информационных технологий» - on-line библиотека информационных материалов по компьютерным технологиям.
17. http://www.intuit.ru/ - Образовательный портал дистанционного обучения.
18. www.coursera.org - Платформа для бесплатных онлайн - лекций (проект по публикации образовательных материалов в интернете, в виде набора бесплатных онлайн - курсов).

6.6. Иные источники

1. https://ru.atlassian.com/devops
2. https://ru.wikipedia.org/wiki/DevOps
3. https://vc.ru/hr/50165-chto-takoe-metodologiya-devops-i-komu-ona-nuzhna
4. https://habr.com/ru/post/349796/
5. https://habr.com/ru/company/itglobalcom/blog/447742/
6. https://insights.stackoverflow.com/survey/2019
7. https://yandex.ru/company/researches/2019/it-jobs

7. Материально-техническое и программное обеспечение дисциплины (модуля)

Название лаборатории/класса,	Наименование оборудования	Перечень лицензионного

оснащенного необходимым, в соответствии с требованиями ФГОС/ СУОС, оборудованием		программного обеспечения
учебная аудитория для проведения занятий лекционного типа, текущего контроля и промежуточной аттестации	Рабочие места студентов: столы и стулья – соответственно количеству студентов. Рабочее место для инвалида и лиц с ОВЗ: парта с телескопической столешницей на электромеханическом приводе - 1 шт., кресло-коляска для инвалидов 18" - 1 шт., индукционная петля - 1 шт., компьютер с версией для слабовидящих - 1 шт., кнопка вызова сотрудников – 1 шт. Рабочее место преподавателя: стол – 1 шт., стул – 1 шт, кафедра - 1 шт. Доска меловая и маркерная. Экран, ноутбук Lenovo ideapad 100/15, проектор	Мультимидийный проектор Консультант Плюс
информационно – аналитическая лаборатория – учебная аудитория для проведения занятий семинарского типа, курсового проектирования (выполнения курсовых работ), групповых и индивидуальны х консультаций	Рабочие места: столы компьютерные – в соотвествии с количеством студентов, кресло Престиж Profi -B-20 Самба бордо в рубчик - 15 шт., подставка для ног Fellowes FS-48121 Standard черный - 15 шт. Рабочее место преподавателя: стол компьютерный - 1 шт., стул - 1 шт. Доска меловая или маркерная Персональные компьютеры	Мультимидийный проектор Консультант Плюс
библиотека - помещение для самостоятельной работы	Рабочие места: столы и стулья. Рабочее место преподавателя: стол – 1 шт., стул - 1 шт., кафедра библиотечная - 1 шт. Телефон – 1 шт., цифровой многофункциональный копир - 1 шт., копировальный аппарат МФУ – 1 шт., принтер - 1 шт., сканер – 1 шт. Шкаф – 7 шт, стеллаж-33 шт, библиотечная стойка – 2 шт., стенд – 2 шт. Меловая или маркерная доска. Персональные компьютеры	Мультимидийный проектор Консультант Плюс

Программное обеспечение:

В процессе лекционных и семинарских занятий используется следующее программное обеспечение:

- программы, обеспечивающие навигацию в сети Интернет: «Google chrome»;
- программы, демонстрации видео материалов: проигрыватель «Windows Media»;
- программы для демонстрации и создания презентаций: «Microsoft Power Point».

Информационные справочные системы:

1. www.nnir.ru / - Российская национальная библиотека
2. www.nns.ru / -Национальная электронная библиотека
3. www.rsi.ru / - Российская государственная библиотека
4. www.biznes-karta.ru / -Агентство деловой информации «Бизнес-карта»
5. www. rbs.ru / - Информационное агентство «РосБизнесКонсалтинг»
6. www.aport.ru / - Поисковая система
7. www.rambler.ru / - Поисковая система
8. www.yandex.ru / - Поисковая система
9. www.busineslearning.ru / - Система дистанционного бизнес образования
10. www.test.specialist.ru / - Центр компьютерного обучения МГТУ им. Н. Э. Баумана
11. http://www.consultant.ru/ - Консультант плюс
12. http://www.garant.ru/ - Гарант
13. Научная библиотека РАНХиГС. URL: http://lib.ranepa.ru/;
14. Научная электронная библиотека eLibrary.ru. URL: http://elibrary.ru/defaultx.asp;
15. Национальная электронная библиотека. URL: www.nns.ru; Российская государственная библиотека. URL: www.rsl.ru;
16. Российская национальная библиотека. URL: www.nnir.ru; Электронная библиотека Grebennikon. URL: http://grebennikon.ru/;
17. Электронно-библиотечная система Издательства «Лань». URL: http://e.lanbook.com;
18. Электронно-библиотечная система ЮРАЙТ. URL: http://www.biblio-online.ru/.