

МИНОБРНАУКИ РОССИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ОБРАЗОВАНИЯ
«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «ВГУ»)

УТВЕРЖДАЮ

Заведующий кафедрой
электроники
Усков Г.К.



31.01.2024 г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ
Б1.В.ДВ.03.02 Администрирование серверов для веб проектов

1. Код и наименование направления подготовки/специальности:

09.03.01 Информатика и вычислительная техника

2. Профиль подготовки/специализация:

Программно-аппаратные средства информационных систем

3. Квалификация (степень) выпускника:

бакалавр

4. Форма обучения:

очная

5. Кафедра, отвечающая за реализацию дисциплины:

электроники

6. Составители программы:

Ряполов М.П., к.ф.-м.н.

7. Рекомендована:

НМС физического факультета 24.06.2021, № протокола: 6

8. Учебный год: *2027/2028*

Семестр(ы): *8*

9. Цели и задачи учебной дисциплины

Целью учебной дисциплины является знакомство студентов с современными подходами к организации инфраструктуры для веб проектов, её мониторинга производительности и масштабирования.

Задачами является обучение установки и администрирования веб серверов и реляционных СУБД для использования в веб проектах, поиск узких мест в производительности, настройка балансировщиков запросов и кластеров для распределения нагрузки, обеспечения безопасности передаваемых данных.

10. Место учебной дисциплины в структуре ООП:

Дисциплина входит в вариативную часть и является дисциплиной по выбору. Для её изучения студенты должны пройти курсы Б1.О.11 Информатика, Б1.О.13 Вычислительная техника, Б1.О.14 Операционные системы, Б1.О.15 Программирование, Б1.О.16 Сети и телекоммуникации, Б1.О.17 Защита информации, Б1.О.18 Базы данных и Б1.В.17 Системное администрирование.

11. Планируемые результаты обучения по дисциплине/модулю (знания, умения, навыки), соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями) и индикаторами их достижения:

Код	Название компетенции	Код(ы)	Индикатор(ы)	Планируемые результаты обучения
ПК-2	Способен осуществлять концептуальное, функциональное и логическое проектирование систем среднего и крупного масштаба и сложности	ПК-2.1	Знает возможности современных и перспективных средств разработки программных продуктов, технических средств, методы и средства проверки работоспособности программного обеспечения, методы и средства рефакторинга и оптимизации программного кода	Использует инструменты для автоматизированного развертывания приложений корректно
		ПК-2.2	Проводит анализ требований, вырабатывает варианты реализации программного продукта с учетом требований	На основе требований к программному обеспечению проектирует линии непрерывной интеграции и доставки приложений до потребителя
		ПК-2.3	Проводит оценку и обоснование рекомендуемых решений	Пишет обоснование по использованию инструментов для развертывания и тестирования приложений
		ПК-2.4	Анализирует возможности удовлетворения требований к программному обеспечению	Анализирует пути доставки приложений и вырабатывает рекомендации по оптимизации и улучшению
		ПК-2.5	Оценивает время и трудоемкость реализации программного обеспечения с учетом требований	Оценивает трудоемкость работ по настройке непрерывной интеграции и развертыванию приложений
ПК-4	Способен разрабатывать компоненты системных программных продуктов	ПК-4.4	Применяет методы и средства сборки модулей и компонентов программного обеспечения, разработки процедур для развертывания программного обеспечения, миграции и преобразования данных, создания программных интерфейсов	Использует инструменты сборки и доставки программного обеспечения в рамках облачной инфраструктуры

		ПК-4.5	Разрабатывает и документирует программные интерфейсы	Проектирует, реализует и тестирует поддержку проектов через SSH
		ПК-4.6	Разрабатывает процедуры сборки модулей и компонентов программного обеспечения	Проектирует, реализует и тестирует системы сборки программных модулей с помощью инструментов на сервере

12. Объем дисциплины в зачетных единицах/час.— 4/144.

Форма промежуточной аттестации **экзамен**

13. Трудоемкость по видам учебной работы

Вид учебной работы	Трудоемкость			
	Всего	По семестрам		
		8		
Аудиторные занятия	32	32		
в том числе:	лекции	22	22	
	практические			
	лабораторные	10	10	
Самостоятельная работа	76	76		
в том числе: курсовая работа (проект)				
Форма промежуточной аттестации (экзамен – __ час.)	36	36		
Итого:	144	144		

13.1. Содержание дисциплины

п/п	Наименование раздела дисциплины	Содержание раздела дисциплины	Реализация раздела дисциплины с помощью онлайн-курса, ЭУМК *
1. Лекции			
1.1	Платформа для веб приложений на языке PHP	Развёртывание стека Apache→php→mysql и установка веб-приложения на языке php (CMS Drupal). Подготовка рецептов для puppet/ansible для автоматической подготовки стека.	Эл. курс «Администрирование серверов для WEB проектов»
1.2	Нагрузочное тестирование	Использование системы мониторинга zabbix и инструмента нагрузочного тестирования JMeter для проведения тестирования и интерпретация его результатов.	Эл. курс «Администрирование серверов для WEB проектов»
1.3	Преодоление узких мест	Методы преодоления узких мест в производительности, найденные на этапе нагрузочного тестирования: вертикальное масштабирование, кеширование, кластеризация СУБД, кластеризация интерпретатора PHP, балансировка нагрузки. Обеспечение скорости доступа и задержки в географически распределённых проектах (использование различных дата-центров и CDN).	Эл. курс «Администрирование серверов для WEB проектов»
1.4	Платформа для веб приложений на языке	Установка стека Apache→ruby→Postgresql и установка веб приложения на языке ruby (Gitlab)	Эл. курс «Администрирование серверов для WEB проектов»

	Ruby		рование серверов для WEB проектов»
1.5.	Использование непрерывной интеграции и непрерывной доставки для веб приложений.	Использование CI/CD инструментов Gitlab для разворачивания веб приложений. Подготовка рецептов для puppet/ansible для автоматической подготовки стека и необходимой инфраструктуры для работы CI/CD инструментов.	Эл. курс «Администрирование серверов для WEB проектов»
1.6.	Резервное копирование	Обеспечение резервного копирования без остановки работы веб проектов. Планирование восстановления.	Эл. курс «Администрирование серверов для WEB проектов»
2. Лабораторные занятия			
2.1	Стек LAMP	Установка и настройка Apache2, php 7.x, MariaDB 10.x	Эл. курс «Администрирование серверов для WEB проектов»
2.2	Установка веб приложения на PHP	Установка CMS Drupal и заполнение её информацией для проведения тестирования.	Эл. курс «Администрирование серверов для WEB проектов»
2.3.	Система мониторинга Zabbix	Установка сервера и клиентов системы мониторинга, настройка экранов для получения оперативной информации о тестировании.	Эл. курс «Администрирование серверов для WEB проектов»
2.4.	JMeter	Установка и методы построение тестового плана. Составление плана тестирования для сайта и проведение нагрузочного тестирования созданного ранее сайта.	Эл. курс «Администрирование серверов для WEB проектов»
2.5.	Кеширование данных	Использование веб-сервера nginx для кеширования данных и запуск интерпретатора PHP в виде сервиса для уменьшения накладных расходов.	Эл. курс «Администрирование серверов для WEB проектов»
2.6.	Горизонтальное масштабирование	Кластер СУБД и кластер PHP служб. Балансировка нагрузки.	Эл. курс «Администрирование серверов для WEB проектов»
2.7.	Установка веб приложения на Ruby.	Установка и настройка Apache2, ruby и postgresql, веб приложения Gitlab.	Эл. курс «Администрирование серверов для WEB проектов»
2.8.	Системы непрерывной интеграции	Настройка рабочего сервера для публикации веб приложения из Gitlab. Разработка рабочего процесса для проведения публикации приложений.	Эл. курс «Администрирование серверов для WEB

			проектов»
2.9.	Резервное копирование	Методы резервного копирования файлов и базы веб приложений. Резервное копирование в «горячем» режиме без остановки приложения.	Эл. курс «Администрирование серверов для WEB проектов»

13.2. Темы (разделы) дисциплины и виды занятий

№ п/п	Наименование темы (раздела) дисциплины	Виды занятий (количество часов)				
		Лекции	Практические	Лабораторные	Самостоятельная работа	Всего
1.1	Платформа для веб приложений на языке PHP	2			10	12
1.2	Нагрузочное тестирование	4		2	14	20
1.3	Преодоление узких мест	4		2	12	18
1.4	Платформа для веб приложений на языке Ruby	4		2	14	20
1.5.	Использование непрерывной интеграции и непрерывной доставки для веб приложений.	4		2	14	20
1.6.	Резервное копирование	4		2	12	18
	Итого:	22		10	76	108

14. Методические указания для обучающихся по освоению дисциплины

Работа по освоению дисциплины состоит из изучения основного материала, освещаемого на лекциях, выполнения лабораторных работ и выполнения самостоятельных практических работ. При проведении контактной работы (лекции и лабораторные) может организовываться как очная работа в аудитории, так и дистанционная с помощью видеоконференц связи. За основную часть заданий студенты могут получить оценки, сдавая их в портале электронного обучения, финальные задания обязательно требуют собеседования с обучающимся.

15. Перечень основной и дополнительной литературы, ресурсов интернет, необходимых для освоения дисциплины

а) основная литература:

№ п/п	Источник
1.	Олифер В. Г. Сетевые операционные системы / В. Г. Олифер, Н. А. Олифер. — СПб. : Питер, 2002. - 538 с.
2.	Таненбаум Э. Современные операционные системы / Э. Таненбаум, Б. Херберт – СПб. : Питер, 2015. – 1120 с.

б) дополнительная литература:

№ п/п	Источник
3.	Маклин Й. Администрирование Windows Server 2008. Учебный курс Microsoft / Й. Маклин, О. Томас — М. Русская редакция, 2013. — 688 с.
4.	Sander van Vugt Red Hat RHCSA/RHCE 7 Cert Guide: Red Hat Enterprise Linux 7 (EX200 and EX300) / Sander van Vugt — Pearson IT Certification, 2015. — 1008 p.
5.	Bresnahan C. CompTIA Linux+ Powered by Linux Professional Institute Study Guide: Exam LX0-103 and Exam LX0-104, 3rd Edition / Christine Bresnahan — Linux Professional Institute, 2015. — 696 p.
6.	Минаси М. Windows Server 2012 R2. Полное руководство. Том 1. Установка и конфигурирование сервера, сети, DNS страниц / Марк Минаси, Кевин Грин, Кристиан Бус, Роберт Батлер, и др. - М. Диалектика, 2014. - 960с.
7.	Минаси М. Windows Server 2012 R2. Полное руководство. Windows Server 2012 R2. Полное руководство. Том 2. Дистанционное администрирование, установка среды с несколькими доменами, виртуализация, мониторинг и обслуживание сервера страниц / Марк Минаси, Кевин Грин, Кристиан Бус, Роберт Батлер, и др. - М. Диалектика, 2014. - 864с.
8.	Служба Active Directory. Ресурсы Windows Server 2008 = Windows Server 2008. Active Directory. Resourst Kit / Стэн Раймер [и др.].— М. ; СПб. : Русская Редакция : Питер, 2009. — 793 с.

9.	Дейтел Х. Операционные системы. Основы и принципы / Дейтел Харви, Дейтел Пол Дж., Чофнес Дэвид Р. - М. Бином, 2013. - 1024с.
----	--

в) информационные электронно-образовательные ресурсы (официальные ресурсы интернет)*:

№ п/п	Ресурс
10.	Электронный курс «Администрирование серверов для WEB проектов»: URL - https://edu.vsu.ru/course/view.php?id=3777
11.	Zabbix documentation: URL - https://www.zabbix.com/documentation/
12.	JMeter user manual: URL - https://jmeter.apache.org/usermanual/index.html
13.	Gitlab docs: URL - https://docs.gitlab.com/
14.	Pro Git: URL - https://git-scm.com/book/ru/v2

16. Перечень учебно-методического обеспечения для самостоятельной работы

№ п/п	Источник
15.	LAMP Linux Apache MySQL PHP: URL - https://wiki.debian.org/ru/LaMp
16.	Gitlab CI/CD: URL - https://docs.gitlab.com/ee/ci/

17. Образовательные технологии, используемые при реализации учебной дисциплины, включая дистанционные образовательные технологии (ДОТ), электронное обучение (ЭО), смешанное обучение):

При реализации учебной дисциплины лекции и практические занятия могут проводиться очно в аудитории или с использованием видеоконференцсвязи в дистанционном формате. Все задания в обязательном порядке сдаются на портале электронного обучения. Текущая и промежуточные аттестации в дополнение к отправке ответов на задания могут сопровождаться собеседованием. Собеседование, так же как и занятия, может проводиться как очно, так и в дистанционном режиме с помощью видеоконференций.

18. Материально-техническое обеспечение дисциплины:

Для проведения занятий требуется компьютерный класс с проектором или большеформатным экраном. На компьютерах должно быть установлено программное обеспечение для создания виртуальных машин (VirtualBox, KVM либо HyperV). Возможна работа с виртуальными машинами не локально, а через портал самообслуживания в ЦОД вуза. Необходим доступ в интернет для использования внешних общедоступных сервис.

19. Оценочные средства для проведения текущей и промежуточной аттестаций

Порядок оценки освоения обучающимися учебного материала определяется содержанием следующих разделов дисциплины:

№ п/п	Наименование раздела дисциплины (модуля)	Компетенция(и)	Индикатор(ы) достижения компетенции	Оценочные средства
1.1	Платформа для веб приложений на языке PHP	ПК-2	ПК-2.1, ПК-2.2	Тесты, практикоориентированные задания
1.2	Нагрузочное тестирование	ПК-2	ПК-2.1, ПК-2.2	Тесты, практикоориентированные задания
1.3	Преодоление узких мест	ПК-2	ПК-2.3, ПК-2.4, ПК-2.5	Тесты, практикоориентированные задания
1.4	Платформа для веб приложений на языке Ruby	ПК-4	ПК-4.4, ПК-4.5, ПК-4.6	Тесты, практикоориентированные задания, индивидуальные проекты
1.5	Использование непрерывной	ПК-4	ПК-4.4, ПК-4.5, ПК-4.6	Тесты, практикоориентированные задания, индивидуальные проекты

№ п/п	Наименование раздела дисциплины (модуля)	Компетенция(и)	Индикатор(ы) достижения компетенции	Оценочные средства
	интеграции и непрерывной доставки для веб приложений.			
1.6.	Резервное копирование	ПК-4	ПК-4.4, ПК-4.5, ПК-4.6	Тесты, практикоориентированные задания, индивидуальные проекты
Промежуточная аттестация форма контроля - экзамен				Набор практических заданий

20 Типовые оценочные средства и методические материалы, определяющие процедуры оценивания

20.1 Текущий контроль успеваемости

Контроль успеваемости по дисциплине осуществляется с помощью практикоориентированных заданий, тестовых заданий и индивидуальных проектов.

Примеры практикоориентированных заданий:

1. Создание репозитория для проекта на GitLab
2. Установка веб приложения на одном сервере
3. Установка веб приложения с трёхзвенной архитектурой на группе серверов
4. Настройка правил установки приложения.
5. Резервное копирование трёхзвенных приложений
6. Составление тестового плана
7. стек LAMP
8. Установка веб приложения на PHP
9. Система мониторинга Zabbix
10. JMeter
11. Кеширование данных
12. Горизонтальное масштабирование
13. Установка веб приложения на Ruby.
14. Системы непрерывной интеграции
15. Резервное копирование

Примеры тестовых заданий:

Выберите один или несколько ответов:

Основными задачами DevOps являются:

- Автоматизация тестирования программного проекта
- Автоматизация сохранения кода в центральном репозитории
- Автоматизация установки приложения на рабочих станциях разработчиков
- Автоматизация установки проекта на сервере для промышленного использования

Введите короткий ответ:

Хранилище кода проекта, необходимое для организации его автоматического тестирования

СУВ, система управления версиями, version control system, vcs

Соедините термин с его определением:

Commit	Операций отправки кода в локальный репозиторий
Push	Отправка изменения в локальном репозитории на сервер.
Pull	Получение изменений с сервера в локальный репозиторий и применение их в текущий код проекта
Fetch	Получение из серверного репозитория изменений без применения их к

Примеры индивидуальных проектов:

1. Планирование тестирования существующего веб приложения и сравнение его производительности при использовании трёхзвенной архитектуры и при монолитной.
2. Настройка автоматического обновления веб приложения из репозитория исходного кода.
3. Настройка системы «горячего» резервного копирования приложения с трёхзвенной архитектурой при условии хранения резервных в географически распределённых местах.

20.2 Промежуточная аттестация

Промежуточная аттестация по дисциплине осуществляется с помощью практико-ориентированного задания.

Примерные задания:

1. Составление плана нагрузочного тестирования по журналам работы веб сервера
2. Горизонтальное масштабирование веб приложения
3. Настройка процессов CI/CD для публикации и обновления прилоежных.

При проведении промежуточной аттестации обучающийся получает одно случайное задание и выполняет его в течении ограниченного времени. Оценивается функционал полученного в результате решения. Допускается использование подготовленной в течении семестра собственной документации и материалов из открытых источников. По окончании времени или завершения выполнения задания обучающийся демонстрирует работающий функционал.

Задание считается полностью выполненным, если весь описанный в нём функционал работает и обучающийся может это продемонстрировать. В этом случае ему выставляется оценка «отлично». Если функционал реализован не более, чем на 75%, либо обучающийся не может его продемонстрировать самостоятельно, он получает оценку «хорошо». При функционале реализованном на 50% оценка «удовлетворительно». При реализации функционала менее, чем на 30% обучающийся получает оценку «неудовлетворительно».