

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

УТВЕРЖДАЮ

Проректор по учебной работе
и образовательным инновациям

О.Н. Здрок

« 20 » 2020 г.

Регистрационный № УД- 8931/уч.

АРХИТЕКТУРА РАСПРЕДЕЛЕННЫХ ПРИЛОЖЕНИЙ

**Учебная программа учреждения высшего образования
по учебной дисциплине для специальности:**

1-31 80 03 Математика и компьютерные науки

Профилизация: Веб-программирование и интернет-технологии

2020 г.

Учебная программа составлена на основе образовательного стандарта ОСВО 1-31 80 03-2019 и учебных планов: №G31-033/уч., №G31з-034/уч. от 11.04.2019.

СОСТАВИТЕЛЬ:

Борис Михайлович ДУБРОВ, доцент кафедры Веб-технологий и компьютерного моделирования, кандидат физико-математических наук, доцент.

РЕЦЕНЗЕНТ:

Ирина Львовна КОВАЛЕВА, доцент кафедры Программное обеспечение информационных систем и технологий, ФИТР, БНТУ, кандидат технических наук, доцент

РЕКОМЕНДОВАНА К УТВЕРЖДЕНИЮ:

Кафедрой веб-технологий и компьютерного моделирования (протокол № 9 от 20.05.2020г.);

Научно-методическим Советом БГУ
(протокол № 5 от 17.06.2020)

Заведующий кафедрой



Волков В.М.

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Цели и задачи учебной дисциплины

Цель дисциплины — изучение подходов к проектированию и построению распределенных приложений.

Задачи учебной дисциплины:

- изучение современных методов и технологий построения архитектуры распределенных приложений;
- развитие навыков анализа многокомпонентных систем и методов их взаимодействия;
- развитие способностей практического применения облачных технологий;
- формирование представления о технологиях масштабирования распределенных приложений.

Дисциплина «Архитектура распределенных приложений» покрывает важный набор тем компьютерного моделирования и необходима в первую очередь разработчикам программного обеспечения, решающим практические задачи производственного уровня.

Место учебной дисциплины в системе подготовки специалиста с высшим образованием (магистра).

Учебная дисциплина «Архитектура распределенных приложений» является дисциплиной компонента учреждения высшего образования и входит в состав модуля «Распределенные приложения и криптотехнологии». Её преподавание тесно связано с дисциплинами «.NET Технологии», «Технологии Java EE» и «Язык Python в промышленном окружении».

Требования к компетенциям

Освоение учебной дисциплины «Архитектура распределенных приложений» должно обеспечить формирование следующей специализированной компетенции:

СК-4. Быть способным применять ключевые методы проектирования и защиты информационных систем для реализации устойчивых распределенных и криптоприложений.

В результате изучения учебной дисциплины студент магистратуры должен:

знать:

- основные шаблоны проектирования архитектуры распределенных приложений;
- протоколы взаимодействия многокомпонентных систем;
- стандартные строительные блоки распределенных приложений;
- основы построения облачных сервисов;

уметь:

- проектировать архитектуру распределенных приложений;
- корректно определять топологию взаимодействия компонент распределенных систем;
- использовать современные методы контейнеризации микросервисов;

владеть:

- навыками композиции распределенных систем на основе семейств микросервисов;
- методами масштабирования распределенных систем.

Структура учебной дисциплины

Форма получения высшего образования очная (дневная) и заочная.

Дисциплина изучается в 3 семестре. Всего на изучение учебной дисциплины «Архитектура распределенных приложений» отведено:

- для очной формы получения высшего образования – 198 часов, в том числе 70 аудиторных часа, из них: лекции – 36 часа, лабораторные занятия – 34 часов.
- для заочной формы получения высшего образования – 16 аудиторных часов, из них: лекции - 8 часов, лабораторные занятия – 8 часов.

Трудоемкость учебной дисциплины составляет 6 зачетных единиц.

Формой текущей аттестации по учебной дисциплине является экзамен.

СОДЕРЖАНИЕ УЧЕБНОГО МАТЕРИАЛА

1. Технологии контейнеризации

Понятия образов и контейнеров. Запуск стандартных контейнеров. Сборка нового образа и размещение его в репозитории.

Управление ресурсами контейнеров. Файловые системы контейнера и хоста. Создание выделенных подсетей. Принципы логгирования. Выполнение команд внутри контейнера.

Композиция контейнеров. Создание кластеров и задание параметров масштабирования сервисов.

2. Примеры распределенных приложений

Примеры CMS платформ. Развертывание CMS на основе существующих образов. Взаимодействие с базами данных. Шаблоны разделения бизнес-логики. Примеры многокомпонентных распределенных приложений

3. Протоколы взаимодействия

Структура HTTP сообщений и основные методы. Представление данных в виде JSON. Понятие REST API. Реализация CRUD операций в REST API. Пример практической реализации REST API.

4. Компоненты облачных сервисов

Использование облачных кластеров для запуска и тестирования контейнеров. Практическое знакомство с облачными сервисами на примере AWS. Развертывание облачного кластера. Запуск образов в облаке и задание параметров масштабирования. Использование облачных СУБД. Бессерверные вычисления в облаке.

УЧЕБНО-МЕТОДИЧЕСКАЯ КАРТА УЧЕБНОЙ ДИСЦИПЛИНЫ
(очная форма обучения)

Номер раздела, темы	Название раздела, темы	Количество аудиторных часов					Количество часов УСР	Форма контроля знаний
		Лекции	Практические занятия	Семинарские занятия	Лабораторные занятия	Иное		
1	1 Технологии контейнеризации	2	3	4	5	6 14	8	9 Письменный отчет с устной защитой, устный опрос
1.1	Понятия образов и контейнеров. Знакомство с Docker.	2						
1.2	Запуск стандартных контейнеров. Управление контейнерами. Синтаксис Dockerfile	2			2			
1.3	Управление ресурсами контейнеров. Файловые системы контейнера и хоста. Создание выделенных подсетей.	2			2			
1.4	Принципы логгирования. Выполнение команд внутри контейнера.	2			2			
1.5	Композиция контейнеров. Синтаксис docker-compose	2			2			
1.6	Создание кластеров и задание параметров масштабирования серверов.	2			2			
2	Примеры распределенных приложений	8			8			Письменный отчет с устной защитой, устный опрос

2.1	Примеры SMS платформ. Развертывание SMS на основе существующих образов.	2			2				
2.2	Взаимодействие с базами данных. Примеры развертывания MySQL, PostgreSQL и MongoDB	2			2				
2.3	Шаблоны разделения бизнес-логики. Промежуточные сервисы. Пример использования redux.	2			2				
2.4	Разбор примера многокомпонентное распределенное приложения на основе 5 контейнеров	2			2				
3	Протоколы взаимодействия	6			6				Письменный отчет с устной защитой, устный опрос
3.1	Структура HTTP сообщений и основные методы. Представление данных в виде JSON.	2			2				
3.2	Понятие REST API. Реализация CRUD операций в REST API.	2			2				
3.3	Пример практической реализации REST API на основе Node.js и MongoDB	2			2				
4	Компоненты облачных сервисов	6			6				Письменный отчет с устной защитой, устный опрос
4.1	Использование облачных кластеров для запуска и тестирования контейнеров. Практическое знакомство с облачными сервисами на примере AWS.	2			2				
4.2	Развертывание облачного кластера. Запуск образов в облаке и заданные параметров масштабирования.	2			2				
4.3	Использование облачных СУБД. Пример развертывания MongoDB в облаке.	2			2				
4.4	Бессерверные вычисления в облаке. Lambda-сервисы	2			2				
ВСЕГО		36			34				Экзамен

УЧЕБНО-МЕТОДИЧЕСКАЯ КАРТА УЧЕБНОЙ ДИСЦИПЛИНЫ
(заочная форма обучения)

Номер раздела, темы	Название раздела, темы	Количество аудиторных часов					Количество часов УСР	Форма контроля знаний	
		Лекции	Практические занятия	Семинарские занятия	Лабораторные занятия	Иное			
1		2	3	4	5	6	7	8	9
1	Технологии контейнеризации	2.5				2.5			Письменный отчет с устной защитой, устный опрос
1.1	Понятия образов и контейнеров. Знакомство с Docker. Запуск стандартных контейнеров. Управление контейнерами. Синтаксис Dockerfile	0.5				0.5			
1.2	Управление ресурсами контейнеров. Файловые системы контейнера и хоста. Создание выделенных подсетей.	0.5				0.5			
1.3	Принципы логгирования. Выполнение команд внутри контейнера.	0.5				0.5			
1.4	Композиция контейнеров. Синтаксис docker-compose.	0.5				0.5			
1.5	Создание кластеров и задание параметров масштабирования сервисов.	0.5				0.5			
2	Примеры распределенных приложений	2				2			Письменный отчет с устной защитой, устный опрос
2.1	Примеры SMS платформ. Развертывание SMS на основе суще-	0.5				0.5			

	связующих образов.						
2.2	Взаимодействие с базами данных. Примеры развертывания MySQL, PostgreSQL и MongoDB	0.5		0.5			
2.3	Шаблоны разделения бизнес-логики. Промежуточные сервисы. Пример использования гедух.	0.5		0.5			
2.4	Разбор примера многокомпонентное распределенное приложения на основе 5 контейнеров	0.5		0.5			
3	Протоколы взаимодействия	1.5		1.5			Письменный отчет с устной запиской, устный опрос
3.1	Структура HTTP сообщений и основные методы. Представление данных в виде JSON.	0.5		0.5			
3.2	Понятие REST API. Реализация CRUD операций в REST API.	0.5		0.5			
3.3	Пример практической реализации REST API на основе Node.js и MongoDB	0.5		0.5			
4	Компоненты облачных сервисов	2		2			Письменный отчет с устной запиской, устный опрос
4.1	Использование облачных кластеров для запуска и тестирования контейнеров. Практическое знакомство с облачными сервисами на примере AWS.	0.5		0.5			
4.2	Развертывание облачного кластера. Запуск образов в облаке и заданные параметров масштабирования.	0.5		0.5			
4.3	Использование облачных СУБД. Пример развертывания MongoDB в облаке.	0.5		0.5			
4.4	Бессерверные вычисления в облаке. Lambda-сервисы	0.5		0.5			
ВСЕГО		8		8			Экзамен

ИНФОРМАЦИОННО-МЕТОДИЧЕСКАЯ ЧАСТЬ ЛИТЕРАТУРА

Основная литература

1. Б. Брендан. Распределенные системы. Паттерны проектирования. O’Raily Media, 2019.
2. S. Goasguen. Docker Cookbook. O’Reilly Media, 2015.
3. S. Gulabani. Practical Amazon EC2, SQS, Kinesis, and S3. A Hands-On Approach to AWS. Springer Nature Customer Service Center, 2017.
4. Docker 101 Tutorial. Электронный ресурс: <https://www.docker.com/101-tutorial>.

Дополнительная литература

1. С.Л. Бабичев. Распределенные системы. М.:Юрайт, 2020.

Перечень рекомендуемых средств диагностики и методика формирования итоговой оценки

Контроль работы магистранта проходит в форме опроса на лекциях и лабораторных занятиях, а также во время устной защиты отчета по лабораторным работам. Оценка за ответы на лекциях и лабораторных занятиях включает в себя полноту ответа, наличие аргументов, примеров из практики, глубину понимания терминов, используемых магистрантом при ответе на вопросы. При защите лабораторных работ ценится оригинальность кода, самостоятельность выполнения (самостоятельно выполненное задание, но с некоторыми шероховатостями, ценится выше, чем вышколенный отчет, похожий на отчет соседа).

Формой текущей аттестации по дисциплине «Архитектура распределенных приложений» учебным планом предусмотрен экзамен. Экзамен по дисциплине проходит в устной форме.

При формировании итоговой оценки используется рейтинговая оценка знаний магистранта, дающая возможность проследить и оценить динамику процесса достижения целей обучения. Рейтинговая оценка предусматривает использование весовых коэффициентов для текущего контроля знаний и текущей аттестации студентов по дисциплине.

Формирование оценки за текущую успеваемость:

- отчеты по лабораторным работам – 60 %;
- опросы на лекциях, практических и лабораторных занятиях – 40%.

Рейтинговая оценка по дисциплине рассчитывается на основе оценки текущей успеваемости и экзаменационной оценки с учетом их весовых ко-

эфициентов. Вес оценки по текущей успеваемости составляет 40 %, экзаменационной оценки – 60 %.

Описание инновационных подходов и методов к преподаванию учебной дисциплины

При организации образовательного процесса используется *эвристический подход*, который предполагает демонстрацию многообразия решений большинства профессиональных задач и жизненных проблем.

При организации образовательного процесса используется *практико-ориентированный подход*, который предполагает освоение содержания через решения практических задач.

При организации образовательного процесса *используются методы и приемы развития критического мышления*, которые представляют собой систему, формирующую навыки работы с информацией в процессе чтения и письма; понимания информации как отправного, а не конечного пункта критического мышления.

Методические рекомендации по организации самостоятельной работы обучающихся

Для организации самостоятельной работы студентов по учебной дисциплине рекомендовано разместить на образовательном портале или сайте кафедры учебно-методические материалы: методические указания к лабораторным и практическим занятиям, вопросы для подготовки к экзамену, перечень рекомендуемой литературы, информационных ресурсов.

Примерный перечень заданий для самостоятельной работы студентов

Тема 1. Технологии контейнеризации (8ч.)

Создать Dockerfile для сборки образа на основе данного Node.js (Python, C#) приложения.

Создать docker-compose.yaml для запуска заданного набора контейнеров (Node.js+MySQL, Python+PostgreSQL)

Форма контроля – письменный отчет с устной защитой.

Тема 3. Протоколы взаимодействия (6ч.)

Разработать CRUD API на основе Node+MySQL (Python+PostgreSQL).

Форма контроля – письменный отчет с устной защитой.

Примерный перечень вопросов к экзамену

1. Понятия образов и контейнеров в Docker.
2. Запуск стандартных контейнеров. Управление образами и контейнерами.
3. Сборка нового образа и размещение его в репозитории.
4. Управление ресурсами контейнеров.
5. Файловые системы контейнера и хоста.
6. Управление подсетями микросервисов.
7. Принципы логгирования микросервисов.
8. Выполнение команд внутри контейнера.
9. Понятие композиции контейнеров.
10. Создание кластеров и задание параметров масштабирования сервисов.
11. Понятие CMS платформы.
12. Развертывание CMS на основе существующих образов.
13. Взаимодействие с микросервисов с базами данных.
14. Шаблоны разделения бизнес-логики.
15. Структура HTTP сообщений и основные методы.
16. Представление данных в виде JSON.
17. Понятие REST API.
18. CRUD операций в REST API.
19. Использование облачных кластеров для запуска и тестирования контейнеров.
20. Ресурсы облачного кластера.
21. Масштабирование приложений в облаке.
22. Использование облачных СУБД.
23. Бессерверные вычисления в облаке.

**ПРОТОКОЛ
СОГЛАСОВАНИЯ УЧЕБНОЙ ПРОГРАММЫ ПО ИЗУЧАЕМОЙ
УЧЕБНОЙ ДИСЦИПЛИНЕ С ДРУГИМИ ДИСЦИПЛИНАМИ СПЕ-
ЦИАЛЬНОСТИ**

Название дисциплины, с которой требуется согласование	Название кафедры	Предложения об изменениях в содержании учебной программы учреждения высшего образования по учебной дисциплине	Решение, принятое кафедрой, разработавшей учебную программу (с указанием даты и номера протокола)
.NET технологии	Веб-технологий и компьютерного моделирования	Нет	Вносить изменения не требуется (протокол № 9 от 20.05.2020г.)
Технологии Java EE	Веб-технологий и компьютерного моделирования	Нет	Вносить изменения не требуется (протокол № 9 от 20.05.2020г.)

ДОПОЛНЕНИЯ И ИЗМЕНЕНИЯ К УЧЕБНОЙ ПРОГРАММЕ
на _____ / _____ учебный год

№ п/п	Дополнения и изменения	Основание

Учебная программа пересмотрена и одобрена на заседании кафедры Веб-технологий и компьютерного моделирования (протокол № ____ от ____ 201_ г.)

Заведующий кафедрой

доктор физ.-мат. наук, доцент

(ученая степень, ученое звание)

(подпись)

В.М. Волков

(И.О.Фамилия)

УТВЕРЖДАЮ

Декан факультета

канд. физ.-мат. наук, доцент

(ученая степень, ученое звание)

(подпись)

С.М. Босяков

(И.О.Фамилия)