

Белорусский государственный университет

УТВЕРЖДАЮ

Проректор по учебной работе
и образовательным инновациям

О.Н. Здрок

«*30 апреля*» 2020 г.

Регистрационный № УД-*86571* уч.

**ТЕХНОЛОГИИ ПРОЕКТИРОВАНИЯ И РАЗРАБОТКИ
ВЫСОКОНАГРУЖЕННЫХ ВЕБ-СИСТЕМ**

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

1-31 80 09 Прикладная математика и информатика
профилизация

Алгоритмы и системы обработки больших данных

2020 г.

Учебная программа составлена на основе ОСВО 1-31 80 09-2019 и учебного плана G31-072/уч. от 11.04.2019 г.

СОСТАВИТЕЛИ:

В.И. Отлига – старший преподаватель кафедры дискретной математики и алгоритмики факультета прикладной математики и информатики Белорусского государственного университета;

И.С. Терлиженко – ассистент кафедры дискретной математики и алгоритмики факультета прикладной математики и информатики Белорусского государственного университета.

РЕЦЕНЗЕНТЫ:

А.Ю. Лобанов, инженер-программист ООО «ЯндексБел».

РЕКОМЕНДОВАНА К УТВЕРЖДЕНИЮ:

Кафедрой дискретной математики и алгоритмики
(протокол № 14 от 19 марта 2020 года);

Научно-методическим Советом БГУ
(протокол № 4 от 25 марта 2020 года).

Заведующий кафедрой
дискретной математики и алгоритмики  В.М. Котов

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Цели и задачи учебной дисциплины

Учебная дисциплина «Технологии проектирования и разработки высоконагруженных веб-систем» знакомит студентов магистратуры с базовыми подходами проектирования высоконагруженных веб-систем, а также основными инструментами и технологиями, необходимыми для их разработки. Рассмотрение основных технологий и протоколов для веб-систем сопровождается изучением программных библиотек и инструментов, необходимых для их практического использования. Учебная программа учебной дисциплины также содержит ряд разделов, посвященных современным подходам и парадигмам, необходимым для разработки эффективного и надежного программного кода. Дополнительно, учебная дисциплина базовыми принципами тестирования веб-сервисов.

Цель учебной дисциплины – создание базы для разработки эффективных алгоритмов решения прикладных задач, возникающих при обработке больших объемов информации и формирование у магистрантов умения использовать современные научные и технические достижения в области разработки эффективных алгоритмов для решения конкретных прикладных задач.

Задачи учебной дисциплины:

1. Ознакомление с общими принципам разработки API веб-сервисов и базовыми подходами к построению отказоустойчивых архитектур для веб-систем.
2. Приобретение практических навыков разработки таких систем.

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

Учебная дисциплина относится к компоненту учреждения высшего образования и входит **в модуль** «Избранные главы компьютерных наук».

Программа составлена с учетом **межпредметных связей** с учебными дисциплинами. Основой для изучения учебной дисциплины является учебная дисциплина первой ступени высшего образования «Программирование» и дисциплины второй ступени высшего образования «Эксплуатация и администрирование UNIX-систем», «Внутреннее устройство ОС семейства UNIX» вузовского компонента модуля «Большие данные». Знания, полученные в учебной дисциплине, используются при изучении учебной дисциплины вузовского компонента «Разработка высокопроизводительных приложений на C и C++» модуля «Избранные главы компьютерных наук» и дают студентам магистратуры знания и практические навыки, необходимые в дальнейшем для успешной работы.

Требования к компетенциям

Освоение учебной дисциплины «Технологии проектирования и разработки высоконагруженных веб-систем» должно обеспечить формирование следующих углубленных профессиональных и специализированных и компетенций:

специализированные компетенции:

СК-24. Использовать современные научные и технические достижения в области разработки эффективных алгоритмов для решения конкретных прикладных задач

СК-25. Развивать способность выбирать методологию и технологию проектирования компьютерных систем

углубленные профессиональные компетенции:

УПК-5. Владеть перспективными технологиями программирования.

В результате изучения дисциплины студент магистратуры должен:

знать:

- базовые принципы построения API для веб-сервисов;
- основные подходы к построению архитектуры веб-систем;
- современные стандартные библиотеки в области параллельной обработки потока данных;

уметь:

- проектировать распределённые веб-сервисы;
- разрабатывать веб-сервисы с использованием протокола FastCGI;
- производить нагрузочное тестирование веб-сервиса;

владеть:

- архитектурными подходами построения модели веб-сервиса;
- подходами и парадигмами, необходимыми для разработки эффективного и надежного программного кода.

Структура учебной дисциплины

Дисциплина изучается в 3-м семестре. Всего на изучение учебной дисциплины «Технологии проектирования и разработки высоконагруженных веб-систем» отведено:

– для очной формы получения высшего образования – 90 часов, в том числе 40 аудиторных часов, из них: лекции – 20 часов, практические занятия – 20 часов (в том числе – 10 часов дистанционного обучения).

Трудоемкость учебной дисциплины составляет 3 зачетные единицы.

Форма текущей аттестации по учебной дисциплине – зачет.

СОДЕРЖАНИЕ УЧЕБНОГО МАТЕРИАЛА

Раздел 1. Проектирование Web API

Тема 1.1. Введение

Понятие веб-сервиса. Сервис-ориентированная архитектура. Особенности разработки веб-сервисов. Понятие Web API.

Тема 1.2. Подход RPC

Основные принципы RPC. Протоколы XML-RPC и JSON-RPC. Протокол SOAP, WSDL и UDDI.

Тема 1.3. Подход REST

Архитектурные ограничения REST. Унифицированный интерфейс в REST. Особенности практического использования REST.

Тема 1.4. Представление данных

Форматы XML и JSON. Схемы для XML и JSON. Формат Google Protocol Buffers.

Раздел 2. Распределённая архитектура веб-систем

Тема 2.1. Теорема CAP

ACID системы. Модели согласованности. Теорема CAP. Примеры CA, CP и AP систем. Согласованность в конечном счёте. BASE системы. NoSQL базы данных. Структуры данных CRDT.

Тема 2.2. Отказоустойчивость

Точки отказа. Отказоустойчивость регулярных узлов. Отказоустойчивость хранилища. Отказоустойчивость базы данных. Отказоустойчивость генераторов данных.

Тема 2.3. Балансировка нагрузки

Сравнение уровней балансировки нагрузки. Nginx в качестве балансировщика нагрузки.

Тема 2.4. MongoDB

Документо-ориентированные СУБД. Представление данных в MongoDB. Индексы. Репликация и шардирование. Библиотеки C++ для работы с MongoDB.

Раздел 3. Разработка узла веб-системы

Тема 3.1. Протокол FastCGI

История технологий CGI и FastCGI. Протокол FastCGI. Обзор библиотек для написания FastCGI приложений.

Тема 3.2. Устройство веб-сервера

История веб-серверов. Проблема 10000 соединений. Сравнение веб-серверов на основе процессов, потоков и событий. Блокирующий, неблокирующий и асинхронный ввод/вывод. Устройство nginx. Особенности конфигурирования nginx.

Тема 3.3. Нагрузочное тестирование веб-сервиса

Принципы нагрузочного тестирования. Метрики производительности. Инструмент yandex-tank.

УЧЕБНО-МЕТОДИЧЕСКАЯ КАРТА УЧЕБНОЙ ДИСЦИПЛИНЫ

Дневная форма получения образования с применением дистанционных образовательных технологий

№ п/п	Название раздела, темы	Количество аудиторных часов				Количество часов УСР	Форма контроля знаний
		Лекции	Семинарские занятия	Практически е занятия	Иное		
1	Проектирование Web API	6		2 2(ДО)			
1.1	Введение	1					Устный опрос
1.2	Подход RPC	2					Устный опрос
1.3	Подход REST	2		2			Отчеты по домашним практическим заданиям с их устной защитой.
1.4	Представление данных	1		2 (ДО)			Отчеты по домашним практическим заданиям с их устной защитой. Контрольная работа 1.
2	Распределённая архитектура веб-систем	8		4 2(ДО)			
2.1	Теорема CAP	2		2			Отчеты по домашним практическим заданиям с их устной защитой.
2.2	Отказоустойчивость	2		2			Отчеты по домашним практическим заданиям с их

							устной защитой. Контрольная работа 2.
2.3	Балансировка нагрузки	2					
2.4	MongoDB	2		2 (ДО)			Отчеты по домашним практическим заданиям с их устной защитой.
3	Разработка узла веб-системы	6		4 6(ДО)			
3.1	Протокол FastCGI	2		2			Отчеты по домашним практическим заданиям с их устной защитой.
3.2	Устройство веб-сервера	2		2 2(ДО)			Отчеты по домашним практическим заданиям с их устной защитой.
3.3	Нагрузочное тестирование веб-сервиса	2		2 2(ДО)			Отчеты по домашним практическим заданиям с их устной защитой. Коллоквиум. Контрольная работа 3.

ИНФОРМАЦИОННО-МЕТОДИЧЕСКАЯ ЧАСТЬ

Перечень основной литературы

1. Burns B. Designing Distributed Systems: Patterns and Paradigms for Scalable, Reliable Services. – O'Reilly Media, Inc., 2018. — 147 p.
2. Cheng JinSugih JaminDanny RazYuval Shavitt. Building Scalable Network Services — Springer, 2004 — 193 p.
3. Principled Design of the Modern Web Architecture / Fielding, Roy T.; Taylor, Richard N. — O'Reilly Media, 2009. — 330 p.
4. ОБэнкер К. MongoDB в действии. — ДМК Пресс, 2012. — 395 с.
5. Nedelcu C. Nginx HTTP Server. — Birmingham-Mumbai: Packt Publishing, 2010. — 327 p.

Перечень дополнительной литературы

6. Fielding, Roy Thomas (2000), Architectural Styles and the Design of Network-based Software Architectures, Doctoral dissertation, University of California, Irvine
7. Stroustrup B. C++ Programming Language, 4 edition. – 2013. – 1368.
8. Simon St. Laurent, Joe Johnston, Edd Dumbill. (June 2001) Programming Web Services with XML-RPC. O'Reilly. First Edition.
9. Wilson G. The Architecture Of Open Source Applications. – Lulu. – 345.
10. Майерс С. Эффективное использование C++. 50 рекомендаций по улучшению ваших программ и проектов. — СПб: Питер, 2006. — 240 с.
11. Мейерс С. Эффективный и современный C++: 42 рекомендации по использованию C++11 и C++14. — М.: Вильямс, 2016. — 304 с.
12. Karlsson B. Beyond the C++ Standard Library: An Introduction to Boost. – Addison Wesley Professional, 2005. – 432.
13. Прата С. Язык программирования C++. Лекции и упражнения (6-е изд.). — Вильямс, 2012. — 1248 стр.
14. Brewer E. Spanner, truetime and the cap theorem. – 2017.
15. Gupta A. et al. NoSQL databases: Critical analysis and comparison //2017 International Conference on Computing and Communication Technologies for Smart Nation (IC3TSN). – IEEE, 2017. – С. 293-299.

Перечень рекомендуемых средств диагностики и методика формирования итоговой оценки

Для диагностики компетенции в рамках учебной дисциплины рекомендуется использовать следующие формы:

1. Устная форма: устный опрос, коллоквиум.
2. Письменная форма: контрольные работы.
3. Устно-письменная форма: отчеты по домашним практическим заданиям с их устной защитой, оценивание на основе проектного метода.

Формой текущей аттестации по дисциплине «Технологии проектирования и разработки высоконагруженных веб-систем» учебным планом предусмотрен – зачет.

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

Примерные весовые коэффициенты, определяющие вклад текущего контроля знаний и текущей аттестации в рейтинговую оценку (формирование оценки за текущую успеваемость):

- отчёты по практическим домашним заданиям с их устной защитой – 60 %;
- контрольные работы – 20 %;
- коллоквиум – 10 %;
- устный опрос – 10%;

Рейтинговая оценка по дисциплине рассчитывается на основе оценки текущей успеваемости и зачетной оценки с учетом их весовых коэффициентов. Вес оценки по текущей успеваемости составляет 30 %, зачетная оценка – 70%.

Примерная тематика практических занятий

Занятие № 1. Проектирование API веб сервиса

Занятие № 2. Схемы ответов веб сервиса

Занятие № 3. Проектирование сервиса с точки зрения CAP теоремы

Занятие № 4. Отказоустойчивость сервиса

Занятие № 5. Проектирование базы данных

Занятие № 6. Разработка сервиса «Hello world»

Занятие № 7. Конфигурирование nginx

Занятие № 8. Запросы модификации данных

Занятие № 9. Взаимодействие между узлами

Занятие № 10. Нагрузочное тестирование

Описание инновационных подходов и методов к преподаванию учебной дисциплины

При организации образовательного процесса большинства практических занятий используется *практико-ориентированный подход*, который предполагает освоение содержания учебного материала через решение практических задач, а также приобретение навыков эффективного выполнения разных видов профессиональной деятельности.

Кроме этого, при организации образовательного процесса используется комбинация *методов группового обучения, проектного обучения и учебной дискуссии*. Комбинация методов предполагает: ориентацию на генерирование идей, приобретение навыков для решения исследовательских, творческих и коммуникационных задач, появление нового уровня понимания изучаемой темы, применение знаний (теорий, концепций) при решении проблем, определение способов их решения.

Методические рекомендации по организации самостоятельной работы обучающихся, подготовка к экзамену

Для организации самостоятельной работы студентов магистратуры по учебной дисциплине следует использовать информационно коммуникационные технологии:

Образовательный портал БГУ (<https://edufpmi.bsu.by>);

Образовательный портал InsightRunner (<https://acm.bsu.by>);

систему AnyTask (<https://anytask.org/school/bsu>);

разместить в сетевом доступе комплекс учебных и учебно-методических материалов (учебно-программные материалы, учебное издание для теоретического изучения дисциплины, презентации лекций, методические указания к практическим занятиям, электронные версии домашних заданий, материалы текущего контроля и текущей аттестации, позволяющие определить соответствие учебной деятельности обучающихся требованиям образовательных стандартов высшего образования и учебно-программной документации, в том числе вопросы для подготовки к экзамену, задания, вопросы для самоконтроля, список рекомендуемой литературы, информационных ресурсов и др.).

Примерный перечень вопросов к зачету

Теоретическая часть:

1. Подход RPC в проектировании API.
2. Подход REST в проектировании API.
3. ACID и BASE системы
4. Теорема CAP для распределенных систем; CP системы.

5. Теорема CAP для распределенных систем; AP системы.
6. Теорема CAP для распределенных систем; CA системы.
7. Отказоустойчивость фронтенда
8. Отказоустойчивость бэкенда
9. Отказоустойчивость базы данных
- 10.Отказоустойчивость хранилища
- 11.Отказоустойчивость генератора
- 12.Протокол FastCGI.
- 13.Проблема 10000 соединений для веб-серверов.
- 14.Веб-сервер Nginx.
- 15.Нагрузочное тестирование.
- 16.База данных MongoDB

Практическая часть:

1. Разработать REST веб-сервис с помощью FastCGI с поддержкой модификации данных пользователями.
2. Разработать REST веб-сервис с помощью FastCGI с поддержкой обновления данных в процессе работы.
3. Разработать RPC веб-сервис с помощью FastCGI с поддержкой модификации данных пользователями.
4. Разработать REST веб-сервис в виде плагина к Nginx.
5. Разработать веб-сервер, поддерживающий 10000 одновременных соединений.

Рекомендуемая тематика контрольных работ

Контрольная работа №1. Проектирование Web API.

Контрольная работа №2. Распределённая архитектура веб-систем.

Контрольная работа №3. Разработка узла веб-системы.

ПРОТОКОЛ СОГЛАСОВАНИЯ УЧЕБНОЙ ПРОГРАММЫ

Название учебной 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)
Разработка высокопроизводительных приложений на С и С++	Дискретной математики и алгоритмики	Нет	Оставить содержание учебной дисциплины без изменения (протокол № 14 от 19 марта 2020 г.)

ДОПОЛНЕНИЯ И ИЗМЕНЕНИЯ К УЧЕБНОЙ ПРОГРАММЕ

на ____/____ учебный год

№№ пп	Дополнения и изменения	Основание

Учебная программа пересмотрена и одобрена на заседании кафедры дискретной математики и алгоритмики (протокол № ____ от _____ 202_ г.)

Заведующий кафедрой

(ученая степень, звание)

(подпись)

(И.О. Фамилия)

УТВЕРЖДАЮ

Декан факультета

(ученая степень, звание)

(подпись)

(И.О.Фамилия)