

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

УТВЕРЖДАЮ

Ректор Белорусского
государственного университета

А.Д.Король

23 декабря 2024 г.
Регистрационный № 2278/б.



WEB-ПРОГРАММИРОВАНИЕ

Учебная программа учреждения образования по учебной дисциплине для
специальности:

6-05-0533-11 Прикладная информатика

2024 г.

Учебная программа составлена на основе ОСВО 6-05-0533-11-2023, учебных планов БГУ: №6-5.3-59/03 от 15.05.2023, №6-5.3-59/04 от 15.05.2023, №6-5.3-59/05 от 15.05.2023, №6-5.3-59/11ин. от 31.05.2023, №6-5.3-59/12ин. от 31.05.2023, №6-5.3-59/13ин. от 31.05.2023.

СОСТАВИТЕЛЬ:

Е.Д.Рафеевко, доцент кафедры многопроцессорных систем и сетей факультета прикладной математики и информатики Белорусского государственного университета, кандидат физико-математических наук, доцент.

РЕЦЕНЗЕНТ:

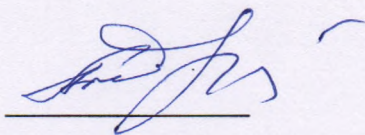
Н.В.Лапицкая, заведующая кафедрой программного обеспечения информационных систем Учреждения образования «Белорусский государственный университет информатики и радиоэлектроники», кандидат технических наук, доцент.

РЕКОМЕНДОВАНА К УТВЕРЖДЕНИЮ:

Кафедрой многопроцессорных систем и сетей
(протокол № 4 от 21.11.2024);

Научно-методическим Советом БГУ
(протокол № 5 от 19.12.2024)

Заведующий кафедрой



И.Е.Андрушкевич

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Цели и задачи учебной дисциплины

Цель учебной дисциплины «Web-программирование»: создание представления о современных технологиях проектирования web приложений. Учебная дисциплина «Web-программирование» предполагает изучение языков и технологий программирования клиентских и серверных приложений.

Учебная дисциплина предусматривает изучение языков программирования JavaScript, TypeScript, фреймворков React, Angular для разработки приложений на клиентской стороне и Java Enterprise платформы, фреймворка Spring для создания приложений на серверной стороне. Также в рамках дисциплины рассматриваются вопросы организации защиты web-приложений. Содержание учебного материала ориентировано на подготовку студентов к практическому использованию полученных знаний, формирование у них широкого кругозора в области информационных технологий.

Задачи учебной дисциплины:

1. Формирование представления об архитектуре web-приложений.
2. Ознакомление с методами проектирования web-приложений.
3. Подготовка к самостоятельной разработке web-приложений с использованием современных технологий.

Место учебной дисциплины в системе подготовки специалиста с высшим образованием.

Учебная дисциплина относится к модулю «Прикладное программирование» компонента учреждения образования.

Программа составлена с учетом межпредметных связей с учебными дисциплинами. Основой для изучения учебной дисциплины являются учебные дисциплины модуля «Программирование» государственного компонента. Знания, полученные в учебной дисциплине, используются при выполнении студентами курсовых и дипломных работ.

Требования к компетенциям

Освоение учебной дисциплины «Web-программирование» должно обеспечить формирование следующей *специализированной* компетенции:

СК. Использовать программные средства и технологии для создания прикладного программного обеспечения.

В результате освоения учебной дисциплины студент должен:

знать:

- концепции, положенные в основу языков для разработки клиентских и серверных приложений;
- методы обеспечения безопасности информационных систем, построенных на основе web-технологий.

уметь:

- анализировать и разрабатывать проекты корпоративных информационных систем (сайтов);
- создавать динамическое содержание web-страниц;

- организовывать доступ к базам данных в web-приложениях;
- обеспечивать безопасность пользователей и защиту информации;

иметь навык:

- разработки web-систем;
- использования программных комплексов для проектирования, создания и управления web-приложений.

Структура учебной дисциплины

Дисциплина изучается в 4-м, 5-м семестрах. В соответствии с учебным планом всего на изучение учебной дисциплины «Web-программирование» отведено для очной формы получения высшего образования – 216 часов, в том числе 132 аудиторных часа, лекции – 66 часов, лабораторные занятия – 66 часов. **Из них:**

4 семестр:

Лекции – 32 часа, лабораторные занятия – 30 часов, управляемая самостоятельная работа – 2 часа.

Трудоемкость учебной дисциплины составляет 3 зачетных единиц.

Формы промежуточной аттестации – зачет.

5 семестр:

Лекции – 34 часа, лабораторные занятия – 30 часов, управляемая самостоятельная работа – 4 часа.

Трудоемкость учебной дисциплины составляет 3 зачетных единиц.

Формы промежуточной аттестации – экзамен.

СОДЕРЖАНИЕ УЧЕБНОГО МАТЕРИАЛА

Раздел 1. Начальные понятия. Языки разработки web-приложений

Тема 1.1. Введение в web разработку

Основы World Wide Web. Общие принципы построения web приложений. Этапы создания web-проекта.

Тема 1.2. Язык HTML, каскадные таблицы стилей CSS

Язык HTML - цвет, шрифт, границы, отступы, размеры. Обтекание. Вёрстка макетов. Селекторы, иерархические селекторы. Каскадирование. Медиазапросы. Адаптивная вёрстка.

Тема 1.3. Язык JavaScript

Язык JavaScript: типы данных, операторы, функции и объекты. Массивы и хэши. Встроенные классы и объекты. Замыкания. Принципы ООП в JavaScript, прототипирование. Работа с DOM. Работа с окном браузера и с документом. События, всплытие событий.

Раздел 2. Front end разработка

Тема 2.1. JavaScript object notation (JSON). Хранение данных на клиенте

Встроенные методы для работы с JSON. JSON и клонирование. Куки (Cookie). Локальное хранилище данных localStorage.

Тема 2.2. Одностраничные приложения (Single page applications). MVC

Особенности одностраничных приложений, сравнение с традиционными web- приложениями. Паттерны Model View Controller (MVC), Model View Presenter (MVP), Model View ViewModel (MVVM).

Технология AJAX. Асинхронные AJAX-запросы.

Тема 2.3 Модульность. Сборка JavaScript приложений

Модульная организация кода в JavaScript приложении. Инструмент Webpack, встроенный webpack-dev-server. Фреймворк Express. Базовая настройка маршрутизации в приложении.

Тема 2.4 Реактивное программирование. Библиотека RxJS

Потоки событий (Stream). Наблюдатели, наблюдаемые, подписки. Протоколы push и pull. Операторы создания, конвейеризации, трансформации, объединения потоков событий.

Тема 2.5 Язык TypeScript

Основные возможности языка, сравнение с JavaScript. Типы, классы, стрелочные функции. Generic – типы. Модули.

Тема 2.6 Фреймворк React

Технология JavaScript XML (JSX). Настраиваемые компоненты. Жизненный цикл и состояние компонента. Обработка событий, формы.

Тема 2.7 Фреймворк Angular

Модули, компоненты, директивы. Привязка данных. Маршрутизация и навигация (Routing). Получение данных с сервера. Http-клиент.

Раздел 3. Получение данных с сервера

Тема 3.1. Организация работы с базами данных - технология Java DataBase Connectivity (JDBC API)

Соединение с базой данных. Подготовленные запросы и хранимые процедуры.

Пул соединений. Выделение ресурсов соединениям. Data Access Object.

Тема 3.2. Доступ к базам данных с использованием ORM фреймворков

Технология Java Persistence API. Hibernate. Сущности, отношения между сущностями. Язык запросов JPQL. CriteriaAPI, MetamodelAPI.

Раздел 4. Серверные web –приложения

Тема 4.1. Протокол HTTP

Схема HTTP-сеанса. Состав HTTP-запроса. Обеспечение безопасности передачи данных HTTP.

Тема 4.2. Web-сервер и серверные технологии

Взаимодействие между браузером и web-сервером. Сценарные языки: классификация по быстродействию. Компонентные технологии разработки web-приложений.

Тема 4.3. Web- компоненты платформы Java Enterprise (Jakarta EE)

Сервлеты (servlets), контекст сервлета, жизненный цикл. Сессии, обработки web событий, фильтры.

Тема 4.4. Технология шаблонов Thymeleaf

Thymeleaf API, сравнение с другими механизмами создания шаблонов. Контекст Thymeleaf приложения. Выражения в Thymeleaf.

Тема 4.5. Фреймворк Spring

Внедрение зависимостей. Bean-компоненты, области видимости. Spring Boot. Работа с данными в Spring. Spring MVC, контроллеры RESTful. WebSockets в Spring.

Раздел 5. Защита web приложений

Тема 5.1. Авторизация и аутентификация

Типы аутентификации. Использование SSL и сертификата клиента.

Тема 5.2. Spring Security

Служба хранения сведений о пользователях. Провайдеры аутентификации. Защита web - запросов.

УЧЕБНО-МЕТОДИЧЕСКАЯ КАРТА УЧЕБНОЙ ДИСЦИПЛИНЫ

Очная (дневная) форма получения высшего образования с применением дистанционных образовательных технологий (ДОТ)

Номер раздела, темы,	Название раздела, темы	Количество аудиторных часов					Количество часов УСР	Формы контроля знаний
		лекции	практические занятия	семинарские занятия	лабораторные занятия	Иное		
1	2	3	4	5	6	7	8	9
4 семестр								
1	Начальные понятия, языки разработки web-приложений.	14			14			
1.1	Введение в web разработку	2						Устный опрос
1.2	Язык HTML, каскадные таблицы стилей CSS	4			4			Контрольная работа №1. Устная защита лабораторных работ
1.3	Язык JavaScript	8			10			Контрольная работа №2. Устная защита лабораторных работ
2	Front end разработка	24			22		2	
2.1	JavaScript object notation (JSON). Хранение данных на клиенте	2			2			Устная защита лабораторных работ
2.2	Одностраничные приложения (Single page applications). MVC	2			2			Устная защита лабораторных работ
2.3	Модульность. Сборка JavaScript приложений	2			2			Устная защита лабораторных работ
2.4	Реактивное программирование.	4			4			Контрольная работа №3.

	Библиотека RxJS							Устная защита лабораторных работ
2.5	Язык TypeScript	2					2	Собеседование по заданиям УСР
2.6	Фреймворк React	6			6			Контрольная работа №4. Устная защита лабораторных работ
5 семестр								
2.7	Фреймворк Angular	6			6			Контрольная работа №5. Устная защита лабораторных работ
3	Получение данных с сервера	8			8			
3.1	Организация работы с базами данных - технология Java DataBase Connectivity (JDBC API)	4			4			Контрольная работа №6. Устная защита лабораторных работ
3.2	Доступ к базам данных с использованием ORM фреймворков	4			4			Устная защита лабораторных работ
4	Серверные web – приложения	16			12		4	
4.1	Протокол HTTP	1						Устный опрос по теме 6.1
4.2	Web-сервер и серверные технологии	1						Устный опрос по теме 4.2
4.3	Web-компоненты платформы Java Enterprise (Jakarta EE)	4			4		2	Контрольная работа №7 Устная защита лабораторных работ, собеседование
4.4	Технология шаблонов Thymeleaf	4			4			Устная защита лабораторных работ
4.5	Фреймворк Spring	6			4		2	Устная защита лабораторных работ

								работ. Собеседование по заданиям УСР
5	Защита приложений web	4			4			
5.1	Авторизация и аутентификация	2						Собеседование
5.2	Spring Security	2			4			Контрольная работа №8 Устная защита лабораторных работ

ИНФОРМАЦИОННО-МЕТОДИЧЕСКАЯ ЧАСТЬ

Основная литература

1. Робсон, Э. Изучаем HTML, XHTML и CSS / Элизабет Робсон, Эрик Фримен ; [пер. с англ. В. Черник]. – 2-е изд. – Москва ; Санкт-Петербург ; Минск : Питер, 2022. – 720 с. – URL: <https://ibooks.ru/bookshelf/377028/reading>.
2. Симпсон, К. Область видимости и замыкания / Кайл Симпсон ; [пер. с англ. Е. Матвеев]. – 2-е междунар. изд. – Санкт-Петербург ; Москва ; Минск : Питер, 2022. – 240 с. – URL: <https://ibooks.ru/reading.php?short=1&productid=379921>.
3. Бэнкс, А. React. Современные шаблоны для разработки приложений / Алекс Бэнкс, Ева Порселло ; [пер. с англ. С. Черников]. – 2-е изд. – Санкт-Петербург ; Москва ; Минск : Питер, 2022. – 317 с. – URL: <https://ibooks.ru/bookshelf/379917/reading>.
4. Дронов, В. А. React 17. Разработка веб-приложений на JavaScript / Владимир Дронов. - Санкт-Петербург : БХВ-Петербург, 2022. - 384 с. - URL: <https://ibooks.ru/bookshelf/385763/reading>.
5. Хеклер, М. Spring Boot по-быстрому. Создаем облачные приложения на Java и Kotlin / М. Хеклер ; [пер. с англ. И. Пальти]. – Санкт-Петербург ; Москва ; Минск : Питер, 2022. – 351 с. – URL: <https://ibooks.ru/reading.php?short=1&productid=385989>.

Дополнительная литература

1. Черный, Б. Профессиональный TypeScript. Разработка масштабируемых JavaScript-приложений – Санкт-Петербург [и др.] : Питер, 2021. – 351 с.
2. Современный учебник JavaScript [Электронный ресурс]. – Режим доступа: <https://learn.javascript.ru/>. Дата доступа: 30.08.2024
3. TypeScript [Электронный ресурс]. – Режим доступа : <https://www.typescriptlang.org/>. Дата доступа: 30.08.2024
4. Фреймворк Angular [Электронный ресурс]. – Режим доступа : <https://angular.io/>. Дата доступа: 30.08.2024
5. Фреймворк React [Электронный ресурс]. – Режим доступа : <https://react.dev/>. Дата доступа: 30.08.2024
6. Фреймворк Spring [Электронный ресурс]. – Режим доступа : <https://spring.io/>. Дата доступа: 30.08.2024
7. Thymeleaf [Электронный ресурс]. – Режим доступа : <https://www.thymeleaf.org/>. Дата доступа: 30.08.2024
8. Блинов, И. Н. Java from EPAM : учеб.-метод. пособие / И. Н. Блинов, В. С. Романчик. – Минск: Четыре четверти, 2021. – 560 с.

Электронные ресурсы

1. Образовательный портал БГУ [Электронный ресурс]. – Режим доступа: <https://edufpmi.bsu.by/course/view.php?id=9>. – Дата доступа: 30.08.2024.
2. Образовательный портал БГУ [Электронный ресурс]. – Режим доступа: <https://edufpmi.bsu.by/course/view.php?id=351>. – Дата доступа: 30.08.2024

Перечень рекомендуемых средств диагностики и методика формирования итоговой отметки

Объектом диагностики компетенций студентов являются знания, умения, полученные ими в результате изучения учебной дисциплины. Выявление учебных достижений студентов осуществляется с помощью мероприятий текущего контроля и промежуточной аттестации.

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

1. Устная форма: опросы, устная защита лабораторных работ.
2. Письменная форма: контрольные работы для оценивания на основе модульно-рейтинговой системы.

В качестве рекомендуемых технических средств диагностики используется обучение, организованное на платформе Moodle (<https://edufpmi.bsu.by>).

Формой промежуточной аттестации по дисциплине «Web-программирование» учебным планом предусмотрен **зачет** и **экзамен**.

Для формирования итоговой отметки по учебной дисциплине используется модульно-рейтинговая система оценки знаний студента, дающая возможность проследить и оценить динамику процесса достижения целей обучения. Рейтинговая система предусматривает использование весовых коэффициентов для текущей и промежуточной аттестации студентов по учебной дисциплине.

Формирование итоговой отметки в ходе проведения контрольных мероприятий текущей аттестации (примерные весовые коэффициенты, определяющие вклад текущей аттестации в отметку при прохождении промежуточной аттестации):

- защита лабораторных работ – 40 %;
- контрольные работы – 40 %;
- опросы – 10 %.
- собеседования по заданиям УСР – 10%.

Итоговая отметка по дисциплине рассчитывается на основе итоговой отметки текущей аттестации (модульно-рейтинговой системы оценки знаний) 40 % и экзаменационной отметки 60 %.

Примерный перечень заданий для управляемой самостоятельной работы

В качестве заданий для управляемой самостоятельной работы могут быть выданы задания для самостоятельного решения задач по следующим темам:

Тема 2.5. Язык TypeScript (2 ч.)

Разработка React и TypeScript компонентов.

Форма контроля – собеседование.

Тема 4.3. Web-компоненты платформы Java Enterprise (Jakarta EE) (2 ч.)

Организация поддержки сессии.

Форма контроля – собеседование.

Тема 4.5. Фреймворк Spring (2 ч.)

Проектирование Spring репозитория.

Форма контроля – собеседование.

Примерная тематика лабораторных занятий

Занятие 1. Разработка html-страниц, таблицы, стили.

Занятие 2. Табличная и блочная верстка.

Занятие 3. Введение в JavaScript. Функции, хэши.

Занятие 4. Введение в JavaScript. Функции, хэши – ООП подход.

Занятие 5. JavaScript. Работа с DOM.

Занятие 6. JavaScript. Обработка событий.

Занятие 7. JavaScript. Хранение данных на клиенте. MVC. Модульность

Занятие 8. Создание приложения с использованием RxJS.

Занятие 9. Создание приложения с использованием RxJS. Обработка событий.

Занятие 10. Настройка Express.

Занятие 11. Разработка React компонентов.

Занятие 12. Создание архитектуры Angular приложения.

Занятие 13. Создание Angular приложения. Обработка данных.

Занятие 14. Создание Http-клиента в Angular приложении.

Занятие 15. Создание Angular приложения – навигация.

Занятие 16. Разработка форм в Angular приложении.

Занятие 17. Развертывание Angular приложения на Firebase.

Занятие 18. Создание приложения, имеющего доступ к базе данных, использование API JDBC. Connection Wrapper.

Занятие 19. Создание приложения, имеющего доступ к базе данных, использование API JDBC. Шаблон DAO.

Занятие 20. Организация пула соединений.

Занятие 21. Создание приложения, имеющего доступ к базе данных, с использованием технологии Java Persistence API.

Занятие 22. Создание приложения, имеющего доступ к базе данных, с использованием технологии Java Persistence API. Criteria API, Metamodel API.

Занятие 23. Программирование сервлетов. Использование Cookies.

Занятие 24. Проектирование приложения с использованием Spring Boot, Spring MVC.

Занятие 25. Проектирование приложения с использованием Spring Boot, Spring Data.

Занятие 26. Проектирование приложения с использованием Spring Boot, Spring Security.

Занятие 27. Проектирование пользовательского интерфейса

Занятие 28. Проектирование RESTful web-сервиса.

Занятие 29. Проектирование клиента RESTful web-сервиса.

Занятие 30. Проектирование клиента RESTful web-сервиса. Взаимодействие с Angular приложением.

Примерные темы реферативных работ:

1. Языки разработки сценариев.
2. Компонентные технологии разработки web-приложений.

Рекомендуемая тематика контрольных работ:

- 1) Контрольная работа № 1. Язык HTML, каскадные таблицы стилей CSS. Язык JavaScript – встроенный объект window.
- 2) Контрольная работа № 2. Язык JavaScript – работа с тегами на web-странице.
- 3) Контрольная работа № 3. Реактивное программирование с использованием библиотеки RxJS.
- 4) Контрольная работа № 4. React компоненты с состоянием.
- 5) Контрольная работа № 5. Фреймворк Angular – маршрутизация в приложении.
- 6) Контрольная работа № 6. Взаимодействие с базами данных - JDBC API.
- 7) Контрольная работа № 7. Сервлеты и фильтры в web-приложении.
- 8) Контрольная работа № 8. Программная аутентификация.

Примерный вариант контрольной работы № 1

Создать страницу Task.html с кнопкой, по которой вызывается JS скрипт, позволяющий выполнить программный переход на другой URL.

Примерный вариант контрольной работы № 2

Создать страницу Task.html с формой, в которой должны быть: текстовое поле, переключатель и кнопка Submit. Написать JS скрипт, который валидирует форму (правила валидации – непустое поле ввода и отмеченный переключатель).

Примерный вариант контрольной работы № 3

Создать страницу Task.html с формой, в которой должны быть: текстовое поле и кнопка Submit. Щелчок по кнопке запускает поток, в котором счетчик увеличивается на единицу каждую секунду. Каждый новый щелчок останавливает предыдущий поток и запускает новый, счетчик обнуляется.

Примерный вариант контрольной работы № 4

Создать React компонент, который будет отображать названия всех областей Беларуси, содержащих заданную подстроку.

Примерный вариант контрольной работы № 5

Создать Angular модуль с названием TransportCompany, в этом модуле разместить компоненты TransportCenter и TransportList. При открытии проекта отображается компонент TransportCenter, для которого дочерним подгружается компонент TransportList.

Примерный вариант контрольной работы № 6

Создать БД с одной таблицей, в которой хранится информация о товаре (название, стоимость, наличие). Приложение получает из БД список товаров, которых нет в наличии.

Примерный вариант контрольной работы № 7

Разработать web –приложение, состоящее из сервлета, двух thymeleaf страниц и dao-класса UserDAO. Первая страница содержит форму с полем ввода возраста и кнопкой, по которой вызывается сервлет, перенаправляющий на вторую страницу.

Примерный вариант контрольной работы № 8

Разработать web –приложение, состоящее из сервлета, двух thymeleaf страниц и dao-класса UserDAO. Первая страница содержит форму с полем ввода имени и пароля и кнопкой, по которой происходит аутентификация пользователя .

Описание инновационных подходов и методов к преподаванию учебной дисциплины

При организации образовательного процесса используются следующие методы:

– **метод учебной дискуссии**, который предполагает участие студентов в целенаправленном обмене мнениями, идеями для предъявления и/или согласования существующих позиций по определенной проблеме. Использование метода обеспечивает появление нового уровня понимания изучаемой темы, применение знаний (теорий, концепций) при решении проблем, определение способов их решения.

– **метод группового обучения**, который представляет собой форму организации учебно-познавательной деятельности обучающихся, предполагающую функционирование разных типов малых групп, работающих как над общими, так и специфическими учебными заданиями.

В качестве технических средств для организации работы в рамках учебной дисциплины рекомендуется использовать Образовательный портал БГУ (<https://edufpmi.bsu.by>) – инструмент с эффективной функциональностью контроля, тренинга и самостоятельной работы.

Используются **методы и приемы развития критического мышления**, которые представляют собой систему, формирующую навыки работы с информацией; понимании информации как отправного, а не конечного пункта критического мышления.

При организации образовательного процесса используется **практико-ориентированный подход**, который предполагает освоение содержания дисциплины через выполнение практических заданий, имеющих элементы учебно-исследовательской деятельности.

Методические рекомендации по организации самостоятельной работы

Для организации самостоятельной работы студентов по учебной дисциплине следует использовать современные информационные ресурсы: разместить на образовательном портале комплекс учебных и учебно-методических материалов (учебно-программные материалы, учебное издание для теоретического изучения дисциплины, методические указания к лабораторным занятиям, материалы текущего контроля и текущей аттестации, позволяющие определить соответствие учебной деятельности обучающихся требованиям образовательного стандарта высшего образования и учебно-программной документации, в т.ч. вопросы для подготовки к зачету, задания, тесты, вопросы для самоконтроля, тематика рефератов и др., список рекомендуемой литературы, информационных ресурсов и др.).

Примерный перечень вопросов к зачету

1. Общие принципы построения web приложений. Этапы создания web-проекта.
2. Клиентские сценарии и приложения.
3. Программы, выполняющиеся на клиент-машине.
4. Язык JavaScript.
5. Замыкания.
6. Принципы ООП в JavaScript, прототипирование.
7. Работа с DOM.
8. Работа с окном браузера и с документом.
9. События, всплытие событий.
10. Особенности одностраничных приложений,

11. Локальное хранилище данных localStorage.
12. Модульная организация кода в JavaScript приложении. Инструмент Webpack.
13. Реактивное программирование. Библиотека RxJS.
14. Язык TypeScript.
15. Фреймворк React.

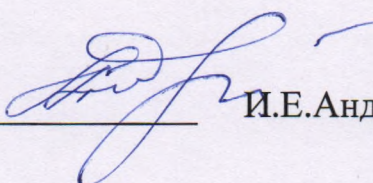
Примерный перечень вопросов к экзамену

1. Архитектура Angular приложения.
2. Связывание данных.
3. Работа с формами.
4. Навигация в Angular приложении.
5. Язык TypeScript. Observables в Angular приложении.
6. Java DataBase Connectivity (JDBC API).
7. Соединение с базой данных.
8. Подготовленные запросы и хранимые процедуры.
9. Шаблон Data Access Object.
10. Пул соединений. Выделение ресурсов соединениям.
11. Транзакции.
12. Технология Java Persistence API.
13. Описание сущностей и отношений между ними.
14. Преобразование наследования.
15. Управление сущностями.
16. Java Persistence Query Language.
17. Использование Criteria API и Metamodel API для создания запросов.
18. Протокол HTTP. Схема HTTP-сеанса. Состав HTTP-запроса, ответа.
19. Web- компоненты Сервлеты.
20. Динамические страницы Thymeleaf.
21. Фильтры.
22. Сессии, куки.
23. Spring Framework
24. Внедрение зависимостей. Dispatcher servlet.
25. Репозитории.
26. REST - контроллер.
27. RESTful service клиент.
28. Безопасность в Spring.

ПРОТОКОЛ СОГЛАСОВАНИЯ УЧЕБНОЙ ПРОГРАММЫ УО

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)
Учебная дисциплина не требует согласования			

Заведующий кафедрой
многопроцессорных систем и сетей
кандидат физико-математических наук,
доцент


И.Е.Андрушкевич

21 ноября 2024 г.

ДОПОЛНЕНИЯ И ИЗМЕНЕНИЯ К УЧЕБНОЙ ПРОГРАММЕ

на ____/____ учебный год

№№ ПП	Дополнения и изменения	Основание

Учебная программа пересмотрена и одобрена на заседании кафедры
многопроцессорных систем и сетей (протокол № ____ от _____ 200_ г.)

Заведующий кафедрой

УТВЕРЖДАЮ

Декан факультета
