

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Крюков Вадим Николаевич

Должность: Проректор по образовательной деятельности и инновационной политике

Дата подписания: 17.06.2026 18:11:53

Уникальный программный ключ:

1b0adb7fd710f6a0705d90c58682bd0c5f2f25b2

**Министерство науки и высшего образования РФ**  
**Федеральное государственное бюджетное образовательное учреждение**  
**высшего образования**  
**«Заполярный государственный университет им. Н. М. Федоровского»**

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ**  
**по дисциплине**  
**Технология создания web-приложений**

Направление подготовки 09.03.02 «Информационные системы и технологии»  
Профиль «Проектирование и реализация информационных систем и технологий»  
Уровень образования: бакалавриат

Кафедра «Информационные системы и технологии»

Разработчик ФОС:

кандидат экономических наук, доцент, Беляев И.С.

\_\_\_\_\_ Беляев И.С.

Оценочные материалы по дисциплине рассмотрены и одобрены на заседании кафедры, протокол от 10.04.2026г. № 5.

Заведующий кафедрой \_\_\_\_\_ к.э.н., Беляев И.С.

Фонд оценочных средств по дисциплине Технология создания web-приложений для текущей/ промежуточной аттестации разработан в соответствии с Федеральным государственным образовательным стандартом высшего образования по специальности / направлению подготовки 09.03.02 Информационные системы и технологии на основе Рабочей программы дисциплины Технология создания web-приложений, утвержденной решением ученого совета от г., Положения о формировании Фонда оценочных средств по дисциплине (ФОС), Положения о проведении текущего контроля успеваемости и промежуточной аттестации обучающихся ЗГУ, Положения о государственной итоговой аттестации (ГИА) выпускников по образовательным программам высшего образования в ЗГУ им. Н.М. Федоровского.

**1. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами образовательной программы**

Таблица 1. Компетенции и индикаторы их достижения

Код и наименование компетенции	Индикаторы достижения
УК-2 Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений	УК-2.1 Определяет круг задач в рамках поставленной цели, определяет связи между ними
	УК-2.2 Предлагает способы решения поставленных задач и ожидаемые результаты; оценивает предложенные способы с точки зрения соответствия цели проекта
	УК-2.3 Планирует реализацию задач в зоне своей ответственности с учетом имеющихся ресурсов и ограничений, действующих правовых норм
УК-3 Способен осуществлять социальное взаимодействие и реализовывать свою роль в команде	УК-3.1 Определяет свою роль в социальном взаимодействии и командной работе, исходя из стратегии сотрудничества для достижения поставленной цели
	УК-3.2 При реализации своей роли в социальном взаимодействии и командной работе учитывает особенности поведения и интересы других участников
	УК-3.3 Анализирует возможные последствия личных действий в социальном взаимодействии и командной работе, и строит продуктивное взаимодействие с учетом этого

ПК-2 Способность разрабатывать программное обеспечение (ПО), включая проектирование, отладку, проверку работоспособности и модификацию ПО	ПК-2.1 Демонстрирует навыки управления процессами формирования и проверки требований к разрабатываемому программному обеспечению с учетом действующих правовых норм и законодательных актов
	ПК-2.2 Использует навыки планирования процесса разработки программного продукта
	ПК-2.3 Составляет планы процесса разработки программного продукта
ПК-6 Способность выполнять проектирование и графический дизайн интерактивных пользовательских интерфейсов	ПК-6.1 Демонстрирует знания к требованиям, предъявляемым к графическому дизайну интерфейса, тенденциям в графическом дизайне
	ПК-6.2 Выбирает оптимальные характеристики элементов интерфейса, создавать визуальный стиль интерфейса

Таблица 2. Паспорт фонда оценочных средств

№п/п	Контролируемые разделы(темы) дисциплины	Кодрезультатаобучения по дисциплине/ модулю	Оценочные средства текущей		Оценочные средства промежуточной	
			Наименование	Форма	Наименование	Форма
<b>6 семестр</b>						

**2. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующие процесс формирования компетенций в ходе освоения образовательной программы.**

2.1. Задания для текущего контроля успеваемости

1. Глобальные компьютерные сети: основные понятия, принципы функционирования.
2. Каталоги ресурсов. Поисковые системы.
3. Язык гипертекстовой разметки страниц HTML: общая структура документа, абзацы, цвета, ссылки.
4. Язык гипертекстовой разметки страниц HTML: списки, графика (графические форматы, графический объект как ссылка).
5. Язык гипертекстовой разметки страниц HTML: таблицы.
6. Фреймы.
7. Общие подходы к дизайну сайта. Разработка макета страницы
8. Язык гипертекстовой разметки страниц HTML: формы.
9. Использование стиля при оформлении сайта. Возможности CSS.
10. CSS. Свойства текста. Свойства цвета и фона.
11. CSS. Свойства шрифта. Свойства блоков.

12. CSS. Свойства списков. Классы. Псевдоклассы.
  13. Хостинг сайтов. Технология FTP. Размещение Интернет-ресурса на сервере провайдера.
  14. Клиент-серверная архитектура.
  15. PHP. Особенности языка.
  16. PHP. Операторы INCLUDE и REQUIRE. Особенности написания функций.
  17. PHP. Работа с классами.
  18. PHP. Регулярные выражения.
1. Глобальные компьютерные сети: основные понятия, принципы функционирования.
  2. Каталоги ресурсов. Поисковые системы.
  3. Язык гипертекстовой разметки страниц HTML: общая структура документа, абзацы, цвета, ссылки.
  4. Язык гипертекстовой разметки страниц HTML: списки, графика (графические форматы, графический объект как ссылка).
  5. Язык гипертекстовой разметки страниц HTML: таблицы.
  6. Фреймы.
  7. Общие подходы к дизайну сайта. Разработка макета страницы
  8. Язык гипертекстовой разметки страниц HTML: формы.
  9. Использование стиля при оформлении сайта. Возможности CSS.
  10. CSS. Свойства текста. Свойства цвета и фона.
  11. CSS. Свойства шрифта. Свойства блоков.
  12. CSS. Свойства списков. Классы. Псевдоклассы.
  13. Хостинг сайтов. Технология FTP. Размещение Интернет-ресурса на сервере провайдера.
  14. Клиент-серверная архитектура.
  15. PHP. Особенности языка.
  16. PHP. Операторы INCLUDE и REQUIRE. Особенности написания функций.
  17. PHP. Работа с классами.
  18. PHP. Регулярные выражения.

## 2.2 Темы письменных работ (эссе, рефераты, курсовые работы и др.)

### 1. Разработка веб-приложений

1. Разработка системы управления задачами с использованием React.js и Node.js.
2. Создание интернет-магазина с функционалом корзины и оплаты через API платежных систем.
3. Разработка социальной сети для обмена знаниями с использованием Django.
4. Создание блог-платформы с поддержкой Markdown и системой комментариев.
5. Разработка CRM-системы для малого бизнеса на базе PHP и MySQL.
6. Создание системы онлайн-бронирования отелей с использованием RESTful API.
7. Разработка платформы для онлайн-обучения с тестированием и сертификатами.
8. Создание системы управления проектами с использованием Agile-методологий.
9. Разработка приложения для анализа поведения пользователей на сайте.
10. Создание системы голосования и опросов с использованием JavaScript и Firebase.

### 2. Работа с базами данных

11. Разработка системы управления базой данных для клиники с возможностью записи пациентов.
12. Создание приложения для анализа данных о продажах с использованием SQL.
13. Разработка системы учета товаров на складе с использованием MongoDB.
14. Создание системы для автоматизации создания отчетов из базы данных.
15. Разработка системы для анализа логов сервера с использованием базы данных.
16. Создание приложения для работы с NoSQL базами данных (например, Cassandra).
17. Разработка системы для миграции данных между различными СУБД.
18. Создание системы для анализа больших данных с использованием Hadoop.
19. Разработка приложения для импорта и экспорта данных в формате JSON/XML.

20. Создание системы для визуализации данных из базы данных.

### 3. Адаптивный и кроссплатформенный дизайн

21. Разработка адаптивного сайта для туристической компании.
22. Создание кроссплатформенного веб-приложения для управления задачами.
23. Разработка мобильной версии интернет-магазина с использованием Bootstrap.
24. Создание адаптивного портфолио-сайта для фрилансера.
25. Разработка системы для автоматизации создания адаптивных шаблонов сайтов.
26. Создание интерактивного лендинга с использованием CSS-анимаций.
27. Разработка адаптивного приложения для онлайн-кинотеатра.
28. Создание системы для автоматического тестирования адаптивности сайтов.
29. Разработка многостраничного сайта с динамической загрузкой контента.
30. Создание адаптивного приложения для управления расписанием занятий.

### 4. Безопасность и защита данных

31. Разработка системы авторизации и аутентификации пользователей с использованием JWT.
32. Создание безопасного API для управления данными пользователей.
33. Разработка системы для защиты от SQL-инъекций в веб-приложениях.
34. Создание приложения для шифрования данных перед отправкой на сервер.
35. Разработка системы для анализа уязвимостей веб-приложений.
36. Создание системы для защиты от DDoS-атак на веб-сервер.
37. Разработка приложения для управления правами доступа пользователей.
38. Создание системы для мониторинга безопасности веб-приложений.
39. Разработка системы для автоматического резервного копирования данных.
40. Создание приложения для анализа логов безопасности сервера.

### 5. Интеграция с внешними сервисами

41. Разработка системы для интеграции с API Google Maps.
42. Создание приложения для анализа погоды с использованием API

### OpenWeatherMap.

43. Разработка системы для работы с API платежных систем (например, Stripe).
44. Создание приложения для анализа данных с социальных сетей (например, Twitter API).
45. Разработка системы для интеграции с облачными хранилищами (например, AWS S3).
46. Создание системы для автоматизации рассылки email-уведомлений.
47. Разработка приложения для работы с API видеохостингов (например, YouTube).
48. Создание системы для анализа данных с IoT-устройств через API.
49. Разработка приложения для работы с API новостных агрегаторов.
50. Создание системы для автоматизации работы с API банковских сервисов.

### 6. Использование современных технологий

51. Разработка PWA (Progressive Web Application) для управления задачами.
52. Создание приложения для анализа данных с использованием WebSocket.
53. Разработка системы для работы с микросервисной архитектурой.
54. Создание приложения для анализа данных в реальном времени с использованием GraphQL.
55. Разработка системы для автоматизации деплоя веб-приложений в облако.
56. Создание приложения для работы с Docker-контейнерами.
57. Разработка системы для анализа производительности веб-приложений.
58. Создание приложения для автоматической генерации документации к коду.
59. Разработка системы для работы с Serverless-архитектурой.
60. Создание приложения для анализа поведения пользователей с использованием машинного обучения.

### 1. Текущий контроль знаний

Тестовые вопросы

1. Что такое Web-приложение? Приведите примеры.
2. Какие основные технологии используются при разработке Web-приложений?
3. В чем заключается разница между клиентской и серверной стороной Web-приложения?
4. Какие языки программирования чаще всего используются на стороне клиента?
5. Какие языки программирования чаще всего используются на стороне сервера?
6. Что такое жизненный цикл Web-приложения? Опишите его этапы.
7. Как работают RESTful API? Для чего они нужны?
8. Что такое адаптивный дизайн? Как он реализуется?
9. Как организуется безопасность данных в Web-приложениях?
10. Что такое WebSocket? Где он применяется?

#### Практические задания

11. Создайте простую HTML-страницу с использованием CSS для стилизации.
12. Напишите JavaScript-скрипт для обработки данных формы.
13. Разработайте RESTful API для управления данными о пользователях.
14. Реализуйте систему авторизации и аутентификации с использованием JWT.

15. Создайте адаптивную версию сайта с использованием Bootstrap.

#### Открытые вопросы

16. Как искусственный интеллект меняет подходы к разработке Web-приложений?
17. Какие ошибки чаще всего допускаются при разработке Web-приложений?
18. Как Web-приложения помогают в снижении затрат на бизнес?
19. Какие факторы следует учитывать при выборе архитектуры Web-приложения?
20. Как Web-приложения поддерживают удаленную работу сотрудников?

## 2. Промежуточная аттестация

#### Тестовые вопросы

21. Что такое микросервисная архитектура? Где она применяется?
22. Какие методы защиты данных используются в Web-приложениях?
23. В чем заключается концепция "Progressive Web Applications" (PWA)?
24. Как Web-приложения поддерживают глобализацию продуктов?
25. Что такое KPI? Для чего они используются в Web-приложениях?

#### Практические задания

26. Разработайте систему KPI для анализа эффективности Web-приложения.
27. Создайте план управления рисками для проекта Web-приложения.
28. Реализуйте систему уведомлений для пользователей Web-приложения.
29. Настройте интеграцию облачного хранилища с Web-приложением.
30. Разработайте план миграции данных из старой системы в новое Web-приложение.

#### Ситуационные задачи

31. Компания хочет автоматизировать процесс обработки данных через Web-приложение. Как вы будете строить модель системы?
32. У вас есть набор данных о поведении пользователей. Как вы будете их анализировать через Web-приложение?
33. Вам нужно разработать Web-приложение для прогнозирования рыночных трендов. Какие технологии вы выберете?
34. Вы работаете над проектом управления рисками. Как Web-приложения могут быть применены в этой задаче?
35. Компания хочет автоматизировать процесс анализа данных. Как вы будете использовать Web-приложения для решения этой задачи?

## 3. Итоговый контроль

#### Защита курсового проекта

36. Разработка Web-приложения для системы управления проектами.
37. Проектирование Web-приложения для анализа данных о клиентах.
38. Создание Web-приложения для управления складскими запасами.
39. Разработка Web-приложения для системы рекомендаций клиентам интернет-магазина.
40. Проектирование Web-приложения для управления качеством продукции.

#### Экзамен

41. Теоретический блок: тестовые вопросы по основным темам курса.
42. Практический блок: выполнение задания на разработку Web-приложения.
43. Анализ предоставленного кода и исправление ошибок.

#### Оценка портфолио

44. Сборник выполненных лабораторных работ.
45. Документация по разработанным проектам.
46. Отчеты по практическим заданиям.

4. Дополнительные виды оценочных средств
47. Устное собеседование по ключевым темам курса.
48. Анализ кода и исправление ошибок в предоставленном проекте.
49. Выполнение заданий на оптимизацию производительности Web-приложения.
50. Разработка прототипа Web-приложения с использованием современных технологий.

#### 1. Текущий контроль знаний:

- Тесты на знание основ HTML, CSS и JavaScript.
- Практические задания по верстке веб-страниц.
- Написание скриптов для обработки данных на клиентской стороне.
- Лабораторные работы по созданию форм и их обработке.

#### 2. Промежуточная аттестация:

- Выполнение контрольной работы (например, создание простой веб-страницы с использованием HTML, CSS и JavaScript).
- Рефераты или эссе на заданные темы.
- Мини-проект (например, разработка адаптивного сайта или веб-приложения).

#### 3. Итоговый контроль:

- Защита курсового проекта (разработка полноценного веб-приложения или сайта).
- Экзамен в виде теста или выполнения практического задания.
- Оценка портфолио выполненных лабораторных и практических работ.

#### 4. Дополнительные виды оценочных средств:

- Устное собеседование по ключевым темам курса.
- Анализ кода и исправление ошибок в предоставленном проекте.
- Выполнение заданий на оптимизацию производительности сайта или приложения.