

Документ подписан простой электронной подписью.
Информация о владельце:

ФИО: Кандрашина Елена Александровна

Должность: И.о. ректора ФГАОУ ВО «Самарский государственный экономический университет»

Дата подписания: 10.07.2025 10:20:25

Уникальный программный ключ:

2db64eb9605ce27edd3b8e8fdd32c70e0674ddd2

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«Самарский государственный экономический университет»

Институт Институт экономики предприятий

Кафедра Прикладной информатики

УТВЕРЖДЕНО

Ученым советом Университета

(протокол № 10 от 22 мая 2025 г.)

РАБОЧАЯ ПРОГРАММА

Наименование дисциплины	Б1.В.ДЭ.06.02 Блокчейн и его приложения
Основная профессиональная образовательная программа	09.03.03 Прикладная информатика программа Интеллектуальные цифровые системы и сервисы в управлении

Квалификация (степень) выпускника Бакалавр

Самара 2025

Содержание (рабочая программа)

Стр.

- 1 Место дисциплины в структуре ОП
- 2 Планируемые результаты обучения по дисциплине, обеспечивающие достижение планируемых результатов обучения по программе
- 3 Объем и виды учебной работы
- 4 Содержание дисциплины
- 5 Материально-техническое и учебно-методическое обеспечение дисциплины
- 6 Фонд оценочных средств по дисциплине

Целью изучения дисциплины является формирование результатов обучения, обеспечивающих достижение планируемых результатов освоения образовательной программы.

1. Место дисциплины в структуре ОП

Дисциплина Блокчейн и его приложения входит в часть, формируемая участниками образовательных отношений (дисциплина по выбору) блока Б1. Дисциплины (модули)

Предшествующие дисциплины по связям компетенций: Веб-программирование, Облачные технологии и сервисы, Интеллектуальные информационные системы в менеджменте, Технологии больших данных, Архитектура интеллектуальных информационных систем, Технологии работы в социальных сетях, Информационно-коммуникационные технологии в профессиональной деятельности, Разработка распределенных приложений, Облачные ресурсы для разработки интеллектуальных сервисов, Разработка цифровых сервисов на основе искусственного интеллекта, Системы машинного обучения в управлении, Цифровой дизайн, инфографика и визуализация данных в управлении

Последующие дисциплины по связям компетенций: Автоматизированное тестирование программного обеспечения, Управление качеством разработки приложений

2. Планируемые результаты обучения по дисциплине, обеспечивающие достижение планируемых результатов обучения по программе

Изучение дисциплины Блокчейн и его приложения в образовательной программе направлено на формирование у обучающихся следующих компетенций:

Профессиональные компетенции (ПК):

ПК-1 - Способен осуществлять организационно-технологическое сопровождение работ по созданию интеллектуальных информационных систем и разрабатывать их прототипы

Планируемые результаты обучения по программе	Планируемые результаты обучения по дисциплине		
ПК-1	ПК-1.1: Знать:	ПК-1.2: Уметь:	ПК-1.3: Владеть (иметь навыки):
	инструменты и методы организационно-технического сопровождения работ на различных этапах цикла разработки интеллектуальных цифровых систем и сервисов	выбирать инструменты и методы организационно-технического сопровождения работ на различных этапах цикла разработки интеллектуальных цифровых систем и сервисов	навыками выполнения и управления работами по созданию интеллектуальных цифровых систем и сервисов на различных этапах цикла разработки с целью повышения эффективности деятельности организаций

3. Объем и виды учебной работы

Учебным планом предусматриваются следующие виды учебной работы по дисциплине:

Очная форма обучения

Виды учебной работы	Всего час/ з.е.
	Сем 6
Контактная работа, в том числе:	54.15/1.5
Занятия лекционного типа	36/1
Занятия семинарского типа	18/0.5
Индивидуальная контактная работа (ИКР)	0.15/0
Самостоятельная работа:	35.85/1
Промежуточная аттестация	18/0.5
Вид промежуточной аттестации: Зачет	Зач

Общая трудоемкость (объем части образовательной программы): Часы	108
Зачетные единицы	3

4. Содержание дисциплины

4.1. Разделы, темы дисциплины и виды занятий:

Тематический план дисциплины Блокчейн и его приложения представлен в таблице.

Разделы, темы дисциплины и виды занятий Очная форма обучения

№ п/п	Наименование темы (раздела) дисциплины	Контактная работа				Самостоятельная работа	Планируемые результаты обучения в соотношении с результатами обучения по образовательной программе
		Лекции	Занятия семинарского типа	ИКР	ГКР		
			Практич. занятия				
1.	Распределенные приложения DApps	18	9	0.75		20	ПК-1.1, ПК-1.2, ПК -1.3
2.	Специфика программного интерфейса	18	9	0.75		15.85	ПК-1.1, ПК-1.2, ПК -1.3
	Контроль	18					
	Итого	36	18	0.15		35.85	

4.2 Содержание разделов и тем

4.2.1 Контактная работа

Тематика занятий лекционного типа

№п/п	Наименование темы (раздела) дисциплины	Вид занятия лекционного типа*	Тематика занятия лекционного типа
1.	Распределенные приложения DApps	лекция	Обзор фреймворков для технологий блокчейн. Инструменты для работы с фреймворком.
		лекция	Фреймворки и смарт-контракты.
		лекция	Модульная Архитектура фреймворков для технологий блокчейн.
		лекция	Виртуализация и API.
		лекция	Создание DApp при помощи фреймворка.
		лекция	React JavaScript-библиотека для создания пользовательских интерфейсов.
		лекция	JavaScript-фреймворков для фронтенда.
		лекция	HTML структура web.
		лекция	CSS стилизирование web.
2.	Специфика программного интерфейса	лекция	Введение в объекты JavaScript.
		лекция	Блоки в JavaScript.
		лекция	Клиентский API web.
		лекция	Visual Studio Code. Среда разработки. Библиотеки и функции для работы с технологиями блокчейн.
		лекция	CSS HTML JavaScript JSON.

		лекция	Настройка сети блокчейн в VS Code.
		лекция	Docker виртуализация для размещения проекта.
		лекция	Разработка смарт-контракта с подключением кошелька.
		лекция	Разработка интерфейса приложения на фреймворков для блокчейн.

*лекции и иные учебные занятия, предусматривающие преимущественную передачу учебной информации педагогическими работниками организации и (или) лицами, привлекаемыми организацией к реализации образовательных программ на иных условиях, обучающимся

Тематика занятий семинарского типа

№п/п	Наименование темы (раздела) дисциплины	Вид занятия семинарского типа**	Тематика занятия семинарского типа
1.	Распределенные приложения DApps	практическое занятие	React JavaScript-библиотека для создания пользовательских интерфейсов.
		практическое занятие	JavaScript-фреймворков для фронтенда.
		практическое занятие	HTML структура web.
		практическое занятие	CSS стилизирование web.
2.	Специфика программного интерфейса	практическое занятие	CSS HTML JavaScript JSON.
		практическое занятие	Настройка сети блокчейн в VS Code.
		практическое занятие	Docker виртуализация для размещения проекта.
		практическое занятие	Разработка смарт-контракта с подключением кошелька.
		практическое занятие	Разработка интерфейса приложения на фреймворков для блокчейн.

** семинары, практические занятия, практикумы, лабораторные работы, коллоквиумы и иные аналогичные занятия

Иная контактная работа

При проведении учебных занятий СГЭУ обеспечивает развитие у обучающихся навыков командной работы, межличностной коммуникации, принятия решений, лидерских качеств (включая при необходимости проведение интерактивных лекций, групповых дискуссий, ролевых игр, тренингов, анализ ситуаций и имитационных моделей, преподавание дисциплин (модулей) в форме курсов, составленных на основе результатов научных исследований, проводимых организацией, в том числе с учетом региональных особенностей профессиональной деятельности выпускников и потребностей работодателей).

Формы и методы проведения иной контактной работы приведены в Методических указаниях по основной профессиональной образовательной программе.

4.2.2 Самостоятельная работа

№п/п	Наименование темы (раздела) дисциплины	Вид самостоятельной работы ***
1.	Распределенные приложения DApps	- подготовка доклада - подготовка электронной презентации - тестирование
2.	Специфика программного интерфейса	- подготовка доклада - подготовка электронной презентации - тестирование

*** самостоятельная работа в семестре, написание курсовых работ, докладов, выполнение контрольных работ

5. Материально-техническое и учебно-методическое обеспечение дисциплины

5.1 Литература:

Основная литература

1. Генкин, А. Блокчейн для всех: как работают криптовалюты, BaaS, NFT, DeFi и другие новые финансовые технологии / А. Генкин, А. Михеев ; под редакцией А. Новресли. — Москва : Альпина Паблишер, 2024. — 588 с. — ISBN 978-5-9614-8046-7. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/137904.html>

Дополнительная литература

1. Храмов, Ю. В. Основы технологии блокчейн и криптовалют для менеджеров : учебное пособие / Ю. В. Храмов. — Казань : Издательство КНИТУ, 2021. — 160 с. — ISBN 978-5-7882-3100-6. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/129149.html>

2. Шурыгин, В. А. Принципы и методы технологии блокчейн в приложении к криптовалютам : учебное пособие / В. А. Шурыгин, И. М. Ядыкин. — Москва : Национальный исследовательский ядерный университет «МИФИ», 2020. — 116 с. — ISBN 978-5-7262-2681-1. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/116419.html>

5.2. Перечень лицензионного программного обеспечения

1. Astra Linux Special Edition «Смоленск», «Орел»; РедОС
2. МойОфис Стандартный 2, МойОфис Образование, Р7-Офис Профессиональный

5.3 Современные профессиональные базы данных, к которым обеспечивается доступ обучающихся

1. Профессиональная база данных «Информационные системы Министерства экономического развития Российской Федерации в сети Интернет» (Портал «Официальная Россия» - <http://www.gov.ru/>)

2. Государственная система правовой информации «Официальный интернет-портал правовой информации» (<http://pravo.gov.ru/>)

3. Профессиональная база данных «Финансово-экономические показатели Российской Федерации» (Официальный сайт Министерства финансов РФ - <https://www.minfin.ru/ru/>)

4. Профессиональная база данных «Официальная статистика» (Официальный сайт Федеральной службы государственной статистики - <http://www.gks.ru/>)

5.4. Информационно-справочные системы, к которым обеспечивается доступ обучающихся

1. Справочно-правовая система «Консультант Плюс»
2. Справочно-правовая система «ГАРАНТ-Максимум»

5.5. Специальные помещения

Учебные аудитории для проведения занятий лекционного типа	Комплекты ученической мебели Мультимедийный проектор Доска Экран
Учебные аудитории для проведения практических занятий (занятий семинарского типа)	Комплекты ученической мебели Мультимедийный проектор Доска Экран Компьютеры с выходом в сеть «Интернет» и ЭИОС СГЭУ
Учебные аудитории для групповых и индивидуальных консультаций	Комплекты ученической мебели Мультимедийный проектор Доска Экран Компьютеры с выходом в сеть «Интернет» и ЭИОС СГЭУ

Учебные аудитории для текущего контроля и промежуточной аттестации	Комплекты ученической мебели Мультимедийный проектор Доска Экран Компьютеры с выходом в сеть «Интернет» и ЭИОС СГЭУ
Помещения для самостоятельной работы	Комплекты ученической мебели Мультимедийный проектор Доска Экран Компьютеры с выходом в сеть «Интернет» и ЭИОС СГЭУ
Помещения для хранения и профилактического обслуживания оборудования	Комплекты специализированной мебели для хранения оборудования

5.6 Лаборатории и лабораторное оборудование

Лаборатория информационных технологий в профессиональной деятельности	Комплекты ученической мебели Мультимедийный проектор Доска Экран Компьютеры с выходом в сеть «Интернет» и ЭИОС СГЭУ Лабораторное оборудование
---	--

6. Фонд оценочных средств по дисциплине Блокчейн и его приложения:

6.1. Контрольные мероприятия по дисциплине

Вид контроля	Форма контроля	Отметить нужное знаком « + »
Текущий контроль	Оценка докладов	+
	Устный/письменный опрос	+
	Тестирование	+
	Практические задачи	+
	Оценка контрольных работ (для заочной формы обучения)	+
Промежуточный контроль	Зачет	+

Порядок проведения мероприятий текущего и промежуточного контроля определяется Методическими указаниями по основной профессиональной образовательной программе высшего образования; Положением о балльно-рейтинговой системе оценки успеваемости обучающихся по основным образовательным программам высшего образования – программам бакалавриата, программам специалитета, программам магистратуры в федеральном государственном автономном образовательном учреждении высшего образования «Самарский государственный экономический университет».

6.2. Планируемые результаты обучения по дисциплине, обеспечивающие достижение планируемых результатов обучения по программе

Профессиональные компетенции (ПК):

ПК-1 - Способен осуществлять организационно-технологическое сопровождение работ по созданию интеллектуальных информационных систем и разрабатывать их прототипы

Планируемые	Планируемые результаты обучения по дисциплине
-------------	---

результаты обучения по программе			
	ПК-1.1: Знать:	ПК-1.2: Уметь:	ПК-1.3: Владеть (иметь навыки):
	инструменты и методы организационно-технического сопровождения работ на различных этапах цикла разработки интеллектуальных цифровых систем и сервисов	выбирать инструменты и методы организационно-технического сопровождения работ на различных этапах цикла разработки интеллектуальных цифровых систем и сервисов	навыками выполнения и управления работами по созданию интеллектуальных цифровых систем и сервисов на различных этапах цикла разработки с целью повышения эффективности деятельности организаций
Пороговый	Основные понятия организационно-технологического сопровождения работ в повседневных задачах с помощью технологий блокчейн	Реализовывать организационно-технологическое сопровождение работ в повседневных задачах с помощью технологий блокчейн	навыками реализации организационно-технологического сопровождения работ в повседневных задачах с помощью технологий блокчейн
Стандартный (в дополнение к пороговому)	Особенности эксплуатации в повседневных задачах в блокчейн	Идентифицировать инциденты в повседневных задачах в блокчейн	Навыками идентификации инцидентов в повседневных задачах в блокчейн
Повышенный (в дополнение к пороговому, стандартному)	Особенности администрирования в повседневных задачах в блокчейн	устранять инциденты в повседневных задачах в блокчейн	навыками устранения инцидентов в повседневных задачах в блокчейн

6.3. Паспорт оценочных материалов

№ п/п	Наименование темы (раздела) дисциплины	Контролируемые планируемые результаты обучения в соотношении с результатами обучения по программе	Вид контроля/используемые оценочные средства	
			Текущий	Промежуточный
1.	Распределенные приложения DApps	ПК-1.1, ПК-1.2, ПК-1.3	Практическое работы	зачет
2.	Специфика программного интерфейса	ПК-1.1, ПК-1.2, ПК-1.3	Практическое работы	зачет

6.4. Оценочные материалы для текущего контроля

Примерная тематика докладов

Раздел дисциплины	Темы
Распределенные приложения DApps	Перспективы распределенных реестров в финансовой сфере Hardhat (javascript)

	Truffle (javascript) Brownie (python) Безопасность и другие проблемы блокчейна Блокчейн и реальный бизнес: какова глубина пропасти? Распределенные процессы в страховании: первый проект в России
Специфика программного интерфейса	Блокчейн для существующих и новых моделей бизнеса Аккредитивы в блокчейне: в поисках бизнес-кейса Современная экосистема блокчейна Блокчейн за границами финансовых приложений Платформа «мастерчейн» и другие проекты Банка России Блокчейн в пост-трейдинговых сервисах Применение смарт-контрактов для интеграции торговых площадок

Вопросы для устного/письменного опроса

Раздел дисциплины	Вопросы
Распределенные приложения DApps	1. Какие методы HTTP-запросов вы знаете? 2. Какие версии HTTP-протокола вам известны? 3. Какие знаете коды ответа (состояния) HTTP? 4. Что такое Cross-Origin Resource Sharing? Как устранить проблемы с CORS? 5. Что такое cookie? 6. Какой максимальный размер cookie? 7. Что означает директива use strict? 8. Чем JS отличается при работе на front-end и back-end? 9. Что такое статическая и динамическая типизация? 10. Как клиент взаимодействует с сервером? 11. Что такое REST? 12. Объяснить понятие мутабельность/иммутабельность? Какие типы являются мутабельными и наоборот? 13. Как искать ошибки в коде? Используете ли вы дебаггер? 14. Каких известных людей из мира JS знаете?
Специфика программного интерфейса	15. Какие существуют типы данных в JS? 16. Как проверить, является ли объект массивом? 17. Как проверить, является ли число конечным? 18. Как проверить, что переменная равна NaN? 19. Чем отличается поведение isNaN() и Number.isNaN()? 20. Сравните ключевые слова var, let, const. 21. Что такое область видимости? 22. Что такое деструктуризация? 23. Для чего предназначены методы setTimeout и setInterval? 24. Сравните подходы работы с асинхронным кодом: callbacks vs promises vs async / await. 25. Можно ли записывать новые свойства / функции в прототипы стандартных классов (Array, Object и т. д.)? Почему нет? В каких случаях это делать можно? Как обезопасить себя, если нужно расширить прототип? 26. Назовите методы массивов, какие помните, и скажите, для чего они нужны. 27. Какие методы перебора массива знаете? В чем их отличие? 28. Как работают операторы присваивания / сравнения / строчные / арифметические / битовые и т. д.? 29. Опишите назначение и принципы работы с коллекциями Map и Set. 30. Что означает глубокая (deep) и поверхностная (shallow) копия объекта?

Задания для тестирования дисциплины для оценки сформированности компетенций (min 20, max 50 + ссылка на ЭИОС с тестами) <https://lms2.sseu.ru/course/index.php?categoryid=2028>

1. Какое ключевое слово позволяет создавать объекты общего вида:

- а) object
- б) this
- в) prototype

2. Какая функция позволяет очищать конечную анимацию:

- а) setInterval
- б) clearTimeout
- в) setTimeout

3. Что, из ниже перечисленного, относится к событию:

- а) начало загрузки web-страницы
- б) сохранение кодировки пользователем
- в) завершение загрузки web-страницы

4. Что, из ниже перечисленного, относится к событию:

- а) изменение кодировки пользователем
- б) начало загрузки web-страницы
- в) сохранение кодировки пользователем

6. Какой оператор служит для создания нового экземпляра из класса одностипных объектов:

- а) this
- б) new
- в) prototype

7. В какой строке создастся новый объект Array:

- а) var pattern = /s&/;
- б) var pattern = new Array[a,b,c];
- в) var pattern = [a,b,c];

8. Какой метод позволяет изменять порядок элементов массива на противоположный:

- а) reverse()
- б) join()
- в) sort()

10. Объекты, отвечающие за то, что содержится на Web-странице в окне браузера, называются:

- а) пользовательскими
- б) клиентскими
- в) встроенными

11. Какой AJAX-транспорт позволит отправить файл на сервер без перезагрузки страницы:

- а) только использование фреймов(Iframe)
- б) XMLHttpRequest справится!
- в) невозможно ввиду ограничений безопасности javascript

12. Что нельзя сделать с помощью XMLHttpRequest:

- а) передать запрос другого типа кроме GET и POST
- б) сделать запрос с http://yandex.ru на http://google.com

в) произвести запрос так, чтобы ответ был готов до следующей строки скрипта

13. Что такое ECMAScript:

- а) спецификация языка Javascript
- б) новый язык программирования
- в) переработанная реализация Javascript

14. Можно ли в скрипте перевести посетителя на другую страницу сайта:

- а) да, но только в рамках текущего сайта
- б) нет, нельзя
- в) да, куда угодно

15. Можно ли использовать один объект XMLHttpRequest для множества разных запросов:

- а) можно
- б) нет, только один запрос на один объект
- в) да, но перед каждым новым запросом надо вызывать abort()

16. Сколько параметров можно передать функции:

- а) сколько указано в определении функции или меньше
- б) любое количество
- в) ровно столько, сколько указано в определении функции

17. Какая арифметическая операция приводит к ошибке в javascript:

- а) корень из отрицательного числа
- б) деление на ноль
- в) нет верного ответа

18. Какие конструкции для циклов есть в javascript:

- а) только одна: for
- б) три: for, while и do...while.
- в) только две: for и while.

19. Hyperledger — это проект...

- а) с открытым исходным кодом
- б) для регистрации корпораций в блокчейне
- в) требующий приобретение лицензии
- г) для написания смарт-контрактов

20. Как называется смарт-контракт в Hyperledger Fabric?

- а) ISRC
- б) ISRC]
- в) код ID
- г) эфириум

Практические задачи (min 20, max 50 + ссылку на ЭИОС с электронным изданием, если имеется)

Раздел дисциплины	Задачи
Распределенные приложения DApps	Перечислите основные компоненты фреймворка (модуль, роут, директива и т.п.). В чем разница между компонентом и директивой? Расскажите о жизненном цикле компонента. Перечислите часто используемые хуки жизненного цикла компонента и расскажите, для чего они нужны? В чем разница между конструктором и ngOnInit-хуком? Как защитить роут от несанкционированного доступа? Какие механизмы предоставляет для этого фреймворк?

	Что такое Lazy loading, как и для чего используется? Какое назначение RouterOutlet? Как компоненты могут взаимодействовать друг с другом? Как создать two-way binding свойство для компонента? Какие типы форм у фреймворка? В каких случаях и что лучше использовать? Какие состояния у формы и как это можно применить? Зачем нужны сервисы? Как с ними работать? Что такое singleton-сервисы? Каково их назначение? Способ создания? Какие есть способы объявления сервисов? Для чего нужны модули? Сколько их должно быть в проекте? Зачем нужны общие модули (shared)? Какие преимущества типизации в TypeScript? Какие возможности TypeScript можно использовать для типизации (здесь имеются в виду интерфейсы, типы, enum и т. д.)? Какая разница между интерфейсом и классом? В чем разница между интерфейсом и абстрактным классом? Какая разница между интерфейсом и типом? Что такое RxJS? Как он используется во фреймворке? Какие компоненты фреймворка тесно связаны с ним? Чем отличаются Observable и Promise? Для чего нужны Subjects? Какие типы Subjects существуют? Как сделать несколько последовательных запросов к API с помощью HTTP-сервиса и RxJS? Какая разница между switchMap, concatMap, mergeMap? Как можно конфигурировать Angular-приложение? Зачем нужны environment-файлы? Когда их лучше не использовать? В чем разница между «умным» (smart) и «глупым» (dumb) компонентами? В каких случаях применяется каждый из них? В чем разница между NgForm, FormGroup и FormControl и как их применяют для построения форм? Зачем нужен и как работает async pipe? Как следить за развитием фреймворка? Каких известных людей, связанных с Angular, знаете / читаете?
Специфика программного интерфейса	Напишите простой запрос для вычисления трех авторов, у которых больше всего книг. Напишите запрос, который выбирает последние три комментария для конкретного пользователя для двух таблиц: комментарии и пользователи. Спроектируйте простую схему базы данных для библиотеки. Для чего используют SQL-оператор HAVING? Зачем используют SQL-оператор LEFT JOIN? Чем отличается embed- от reference-связи в MongoDB? В одном проекте программисты сохраняют данные в MongoDB-коллекции комментариев, используя такие типы данных (смотрите ниже). Что плохого в этом решении? <pre>id: ObjectId text: string author_id: string created_at: Date</pre> В проекте понадобилось внести изменения в структуру таблиц, добавить несколько полей и индексы. Как программисты будут делать это на продакшене?

Раздел дисциплины	Темы
Распределенные приложения DApps	Расскажите, какие есть способы копирования простого объекта типа <code>obj = {a 1, b 2, c 3}</code> Напишите <code>deep clone</code> для объекта. Назовите различные способы, как поменять местами значения двух переменных. Менеджер попросил в задаче поменять статусы из «active, inactive» на «active, removed», но в коде фигурируют только цифры и непонятно, какой статус соответствует какой цифре. Как помочь будущим программистам не лезть в документацию по коду? Вопрос ставят на конкретном примере с кодом.
Специфика программного интерфейса	Необходимо сделать мини проект - список пользователей с формой создания / редактирования пользователя: 1. Для хранения пользователей используйте Firebase (это бесплатно). 2. Для стилизации используйте Bootstrap. 3. Минимальный набор полей пользователя: • имя; • фамилия; • электронная почта; • телефон (в формате +380 (XX) XXX-XX-XX) • дата рождения; • будет плюсом - добавление аватара и возможность стор-картинки. 4. Пользователи должны иметь возможность фильтрации и пагинацию. 5. Проект должен содержать README-файл с шагами для запуска.

6.5. Оценочные материалы для промежуточной аттестации

Фонд вопросов для проведения промежуточного контроля в форме зачета

Раздел дисциплины	Вопросы
Распределенные приложения DApps	В чем принципиальная разница между событиями <code>mouseleave</code> и <code>mouseout</code>? В каком порядке обрабатываются пользовательские события в DOM (<code>click</code>, <code>mouseover</code> и т.д.)? FIFO или LIFO? Что такое Event bubbling и Event capturing? Сравните методы объекта <code>event stopPropagation</code> и <code>stopImmediatePropagation</code>. Какие есть подходы оптимизации производительности веб-страницы? Как реализован механизм <code>same-origin policy</code> в браузере? На какие браузерные API он распространяется? Назовите способы хранения данных в браузере. Сравните их. Web worker`ы. Опишите особенности передачи данных между worker`ами и основным потоком, между разделенными worker`ами. Что такое Transferable-объекты? Расскажите о способах оптимизации выполнения ресурсоемких операций JS для улучшения производительности рендеринга контента на странице. Почему <code>ResizeObserver</code> вызывает события изменения размера до воспроизведения элемента, а не после? Расскажите, как вы понимаете Web Accessibility? Опишите алгоритм создания функционала, который обеспечивает чтение содержимого .txt файла при перетаскивании его из файловой системы в окно браузера. Что такое Virtual DOM?
Специфика программного интерфейса	Объясните разницу между единицами измерения <code>px</code>, <code>em</code>, <code>rem</code>. Для чего нужны CSS-переменные? Приведите несколько примеров использования.

	Что произойдет при добавлении следующего <u>селектора</u>? <pre>* {Box-sizing: border-box; }</pre> Как адаптировать страницу для печати? Опишите особенности кастомизации стилей стандартных элементов форм. Что такое progressive рендеринг? Какие подходы используются? Назовите несколько способов реализации lazy-loading медиаресурсов на странице. Назовите популярные шаблонизаторы для фронтенд-разработки. Опишите особенности их использования. Назовите популярные CSS-методологии и их различия. Как работает CSS Grid? Какие форматы изображений поддерживают анимацию? Как отследить прогресс / окончание CSS @keyframes анимаций или плавных переходов, реализованных с помощью transition, в JS? Какие CSS-свойства могут быть обработаны непосредственно через GPU? Что такое композитные слои и почему большое их количество может привести к аварийному завершению работы браузера на мобильных устройствах? Как переиспользовать Инлайн SVG-элементы на странице? Опишите способы оптимизации SVG-файлов. Как реализовать иконочный шрифт из определенного набора SVG-файлов? Что такое ложное жирное или ложное курсивное (Faux) начертание шрифтов? Что такое #shadow-root в инспекторе HTML-страницы? Зачем нужны Custom Elements? Почему удаление лишних символов пробелов / символов переноса в HTML не отражается на конечной производительности загрузки страницы? Что такое контекст отображения canvas? Какие существуют типы контекста для рендеринга двумерной и трехмерной графики?
--	---

6.6. Шкалы и критерии оценивания по формам текущего контроля и промежуточной аттестации

Шкала и критерии оценивания

Оценка	Критерии оценивания для мероприятий контроля с применением 2-х балльной системы
«зачтено»	ПК-1
«не зачтено»	Результаты обучения не сформированы на пороговом уровне