

**КАЗАНСКИЙ КООПЕРАТИВНЫЙ ИНСТИТУТ (ФИЛИАЛ)
АВТОНОМНОЙ НЕКОММЕРЧЕСКОЙ ОБРАЗОВАТЕЛЬНОЙ
ОРГАНИЗАЦИИ ВЫСШЕГО ОБРАЗОВАНИЯ
ЦЕНТРОСОЮЗА РОССИЙСКОЙ ФЕДЕРАЦИИ
«РОССИЙСКИЙ УНИВЕРСИТЕТ КООПЕРАЦИИ»**

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

РАЗРАБОТКА ПОЛЬЗОВАТЕЛЬСКОГО WEB ИНТЕРФЕЙСА

Специальность: 10.05.01 Компьютерная безопасность

Специализация: «Разработка защищенного программного обеспечения»

Формы обучения: очная

Объем дисциплины:

в зачетных единицах: 3 з.е.

в академических часах: 108 ак.ч.

Форма промежуточной аттестации: зачет

Рабочая программа по дисциплине «Разработка пользовательского web интерфейса» по специальности 10.05.01 Компьютерная безопасность, специализации «Разработка защищенного программного обеспечения», составлена в соответствии с требованиями Федерального государственного образовательного стандарта высшего образования – специалитет по специальности 10.05.01 Компьютерная безопасность, утвержденного приказом Министерства науки и высшего образования Российской Федерации от «26» ноября 2020 г. № 1459, профессионального стандарта 06.032 «Специалист по безопасности компьютерных систем и сетей», утвержденного приказом Министерства труда и социальной защиты Российской Федерации от 14.09.2022 г. № 533н, профессионального стандарта 06.033 «Специалист по защите информации в автоматизированных системах», утвержденного приказом Министерства труда и социальной защиты Российской Федерации от 14.09.2022 г. №525н.

Рабочая программа:

обсуждена и рекомендована к утверждению решением кафедры естественных дисциплин, сервиса и туризма от 25 марта 2025 г., протокол № 6

одобрена Научно-методическим советом Казанского кооперативного института (филиала) от 2 апреля 2025 г., протокол № 3

СОДЕРЖАНИЕ

1. Цель и задачи освоения дисциплины	4
2. Место дисциплины в структуре образовательной программы	4
3. Перечень планируемых результатов обучения по дисциплине	4
4. Объем дисциплины и виды учебной работы.....	5
5. Содержание дисциплины, структурированное по темам (разделам) с указанием количества академических часов и видов учебных занятий.....	6
5.1. Содержание дисциплины	6
5.2. Разделы, темы дисциплины и виды занятий	6
6. Практические занятия	6
7. Лабораторные работы	6
8. Примерная тематика курсовых работ (проектов)	7
9. Самостоятельная работа студента	7
10. Оценивание результатов обучения и уровня сформированности компетенций	13
11. Учебно-методическое обеспечение дисциплины	13
12. Перечень программного обеспечения, профессиональных баз данных и информационных справочных систем, необходимых для освоения дисциплины.....	14
13. Материально–техническое обеспечение дисциплины	15
Обновление рабочей программы дисциплины (модуля)	16

1. Цель и задачи освоения дисциплины

Получение практических навыков и теоретической базы для проектирования и реализации пользовательских интерфейсов с применением современных веб-технологий.

Задачи .

1. Изучить основы веб-технологий и верстку веб-страниц при помощи HTML 5 и CSS 3.

2. Изучить фреймворки компонентов и методологии разработки, такие как Bootstrap, БЭМ.

3. Освоить программирование на JavaScript в функциональном и объектно-ориентированном стиле для разработки пользовательского веб-интерфейса и коммуникации с веб-сервисами.

4. Получить представление о возможностях и ограничениях современных браузеров.

5. Изучить основы компонентной разработки реактивных веб-интерфейсов на примере фреймворка React или Vue.JS.

2. Место дисциплины в структуре образовательной программы

Дисциплина «Разработка пользовательского web интерфейса» относится к части, формируемой участниками образовательных отношений Блока 1 «Дисциплины (модули)» основной образовательной программы – программы специалитета по специальности 10.05.01 Компьютерная безопасность специализация «Разработка защищенного программного обеспечения».

Код компетенции	Предшествующие дисциплины (модули), практики	Изучаемые в текущем семестре дисциплины (модули), практики	Последующие дисциплины (модули), практики
ОПК-2	-	Разработка пользовательского web интерфейса	Разработка приложений в интегрированных средах Автоматизация тестирования ПО Анализ и проектирование информационных систем

3. Перечень планируемых результатов обучения по дисциплине

Изучение дисциплины направлено на формирование у обучающихся компетенций.

Формируемые компетенции (код и наименование компетенции)	Индикаторы достижения компетенций	Планируемые результаты обучения
ОПК-2 Способен применять программные средства системного и прикладного назначений, в том числе отечественного производства, для решения задач профессиональной деятельности	ОПК-2.1 Демонстрирует умение использовать операционные системы общего назначения для решения практических задач	Знать: Основные понятия и термины, связанные с профессиональной компетентностью. Уметь: Анализировать и оценивать свои сильные и слабые стороны в контексте профессиональной деятельности Владеть: Навыками критического мышления и анализа информации..
	ОПК-2.3 Демонстрирует умение использовать специализированные программные средства для автоматизации процесса проектирования баз данных	Знать: Современные тенденции и вызовы, стоящие перед специалистами в данной области. Уметь: Использовать различные инструменты для управления временем и проектами. Владеть: Навыками использования цифровых технологий для повышения эффективности работы.

4. Объем дисциплины и виды учебной работы

Объем дисциплины и виды учебной работы в академических часах с выделением объема контактной работы обучающихся с преподавателем и самостоятельной работы обучающихся

очная форма обучения

Вид учебной деятельности	ак.часов	
	Всего	По семестрам
		3
1. Контактная работа обучающихся с преподавателем:	69	69
Аудиторные занятия, часов всего, в том числе:	68	68
• занятия лекционного типа	34	34
• занятия семинарского типа:	34	34
практические занятия	-	-
лабораторные занятия	34	34
в том числе занятия в форме практической подготовки	-	-
Консультации	0,5	0,5
Контактные часы на аттестацию в период экзаменационных сессий	0,5	0,5
2. Самостоятельная работа студентов всего, в том числе	39	39
- подготовка курсовой работы	-	-
- проработка учебного (теоретического) материала	15	15
- выполнение домашних заданий по практическим занятиям	24	24
- подготовка к зачету	-	-
Промежуточная аттестация: зачет	-	-
ИТОГО:	часов	108
общая трудоемкость	зач. ед.	108
		3
		3

5. Содержание дисциплины, структурированное по темам (разделам) с указанием количества академических часов и видов учебных занятий

5.1. Содержание дисциплины

Тема 1. Основы веб-технологий, HTML, CSS.

О профессии веб-разработчика. Введение в HTML, гиперссылки, формы, таблицы HTML, CSS, боксовая модель, float, Flexbox, адаптивная верстка, CSS Фреймворки, Bootstrap

Тема 2. Разработка пользовательских интерфейсов на JavaScript

Основы Javascript, синтаксис, работа с DOM, события, отладка. Библиотека jQuery JavaScript: массивы, объекты, словари, JSON, JavaScript: XHR, CORS, setTimeout/setInterval, исключения, область видимости, замыкания, анимации CSS, History API, расширенные возможности современных браузеров и HTML5. PWA

5.2. Разделы, темы дисциплины и виды занятий

очная форма обучения

№ п/п	Наименование раздела, темы дисциплины	Виды занятий, включая самостоятельную работу студентов (в часах)			Индикаторы достижения компетенций
		занятия лекционного типа	занятия семинарского типа / из них в форме практической подготовки (лабораторные работы)	самостоятельная работа	
1	Раздел 1 Основы веб-технологий, HTML, CSS	17	17/17	20	ОПК-2.1, ОПК -2.3
2	Разработка пользовательских интерфейсов на JavaScript	17	17/17	19	ОПК-2.1, ОПК -2.3
		34	34/34	39	

6. Практические занятия

Практические занятия не предусмотрены.

7. Лабораторные работы

№ п/п	Наименование раздела, темы дисциплины	Содержание лабораторных работ	Объем (час.)
			Очная форма обучения
1.	Раздел 1. Основы	Сверстать HTML-страницу с нумерованным списком	17

№ п/п	Наименование раздела, темы дисциплины	Содержание лабораторных работ	Объем (час.)
			Очная форма обучения
	веб-технологий, HTML, CSS	гиперссылок и формой. Сверстать шапку, футер, меню по схематичному макету. В контенте сверстать таблицу. Сверстать адаптивно с использованием Flexbox. Сверстать с помощью Bootstrap или другого фреймворка.	
2.	Раздел 2. Разработка пользовательских интерфейсов на JavaScript.	Форма с обработчиком событий JavaScript. Разработать калькулятор стоимости услуг на JavaScript. Галерея на JQuery Slick. Добавление контактной формы, отправка на сервер с помощью fetch/async. разработка проекта, работа в команде. Разработка проекта, работа в команде..	17
	Всего		34

8. Примерная тематика курсовых работ (проектов)

Курсовые работы (проекты) не предусмотрены.

9. Самостоятельная работа студента

Самостоятельная работа студента при изучении дисциплины «Разработка пользовательского web интерфейса» направлена на:

- освоение нормативных правовых актов и учебной литературы, необходимых для освоения дисциплины;
- самостоятельный поиск информации в Интернете и других источниках;
- выполнение домашних заданий по практическим занятиям;
- подготовку к зачету.

Краткие рекомендации по выполнению самостоятельной работы:

Успешное усвоение дисциплины предполагает большой, упорный, серьезный, систематический труд студентов. Важнейшая его составная часть – выполнение разных видов самостоятельной работы.

1. **Составление тематического конспекта** на основе изученной основной и дополнительной учебной литературы. В тематическом конспекте за основу берется содержание темы, вопросы для обсуждения.

Этапы работы.

1.1 Конспектирование делается только после того, как прочитан или усвоен материал для конспектирования.

1.2. Необходимо мысленно или письменно составить план конспекта. По этому плану и будет строиться конспект далее.

1.3. Составление самого конспекта. Можно сказать, что конспект – это расширенные тезисы, дополненные рассуждениями и доказательствами,

содержащимися в материалах для конспекта, а также собственными мыслями и положениями составителя конспекта.

Писать конспект рекомендуется четко и разборчиво. В конспекте можно выделять места текста в зависимости от их значимости. Для этого применяются различного размера буквы, подчеркивания, замечания на полях.

Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

Зачетно-экзаменационные материалы для промежуточной аттестации (зачет)

Полные версии заданий доступны в LMS Moodle.

ЗАДАНИЕ 1

Сверстать HTML 5 страницу с нумерованным списком гиперссылок и формой. Список

должен содержать следующие гиперссылки: 1) абсолютную гиперссылку на главную страницу сайта example.com; 2) абсолютную на главную сайта example.com в протоколе https; 3) ссылку на файл на сервере FTP без авторизации; 4) ссылку на файл на сервере FTP с авторизацией; 5) ссылку на фрагмент страницы некоторого сайта; 6) ссылку на фрагмент текущей страницы; 7) ссылку с двумя параметрами в URL; 8) список ссылок основной навигации сайта с подписями title; 9) ссылку без href; 10) ссылку с пустым href;

11) ссылку, по которой запрещен переход поисковикам; 12) запрещенную для индексации поисковиками; 13) контекстную в тексте абзаца; 14) ссылку-изображение; 15) ссылки из прямоугольных и круглых областей картинки (HTML-тег map); 16) относительную на страницу в текущем каталоге; 17) относительную на страницу в каталоге about; 18) относительную на страницу в каталоге уровнем выше текущего; 19) относительную на страницу в каталоге двумя уровнями выше; 20) сокращенную на главную; 21) сокращенную ссылку на внутреннюю. Форма должна содержать следующие поля: 1) имя (текстовое поле); 2) e-mail (поле email); 3) дата рождения (поле даты); 4) пол (радиокнопки); 5) количество конечностей (радиокнопки); 6) сверхспособности: бессмертие, прохождение сквозь стены, левитация (множественный выбор из списка); 7) биография (многострочное текстовое поле); 8) с контрактом ознакомлен (a) (чекбокс); 9) кнопка «Отправить».

ЗАДАНИЕ 2

Добавить к странице в задании 1 таблицу, шапку, логотип, название сайта, подвал и меню со ссылками по схематичному макету: Таблица содержит строку-заголовок с названиями колонок, не менее 3 колонок и 5 строк, использует объединенные ячейки. Фон четных и нечетных строк различается. Шапка и подвал тянутся на всю ширину окна. Максимальная ширина контента и меню 960px с выравниванием по центру. Логотип обтекает слева название сайта, по горизонтали слева выравнивается по границе контента.

CSS вынести в отдельный файл и подключить стили через id и классы. Использовать теги header, footer, nav.

ЗАДАНИЕ 3

Сверстать страницу из предыдущего задания адаптивно, Mobile First, с применением Flex Box и Media Query. 1. На смартфонах в портретном режиме ссылки в главном меню выводятся вертикальным списком. Шапка, навигация, контент и футер занимают 100% ширины экрана: На планшетах и десктопах страница выглядит как в предыдущем задании. Использовать Flex Box и Media Query. 2. Первыми грузятся правила для мобильных телефонов. Затем, с помощью Media Query, подключаются правила для остальных размеров экрана (Mobile First). 3. В потоке HTML таблица предшествует форме. На смартфоне необходимо отобразить форму визуально выше таблицы (использовать Flex Box order).

ЗАДАНИЕ 4

Сверстать страницу из задания 3 с помощью фреймворка Bootstrap или другого CSS фреймворка. Реализовать расположение элементов по сетке (допускается изменение размеров элементов в задании 3 для выравнивания по сетке). При реализации

адаптивности применить принцип Mobile First. Для управления отступами margin и padding использовать средства фреймворка.

ЗАДАНИЕ 5

Разработать на JavaScript калькулятор стоимости заказа. Калькулятор должен содержать: - текстовое поле для ввода цены товара; - текстовое поле для ввода количества товара; - кнопка подсчета стоимости. При нажатии на кнопку на странице выводится стоимость заказа, рассчитываемая как произведение цены и количества.

ЗАДАНИЕ 6

Разработать на JavaScript калькулятор стоимости товара или услуги и добавить на страницу из предыдущего задания. Калькулятор должен содержать: - поле для ввода количества; - выбор типа товара из списка (не менее 3-х вариантов); - выбор опции товара радиокнопкой (из двух и более вариантов); - выбор некоторого свойства товара чекбоксом. Опции и свойства товара зависят от выбранного типа товара и выводятся на форму динамически при смене типа товара: - первый тип товара не имеет дополнительных опций и свойств (радиокнопки и чекбокс не отображаются); - второй тип товара имеет только опции (радиокнопки отображаются, чекбокс – нет); - третий тип товара имеет только свойство (радиокнопки не отображаются, чекбокс отображается). В зависимости от количества, выбранного типа товара, выбранной опции или свойства динамически пересчитывается цена товара и выводится на странице без перезагрузки

страницы при изменении значений элементов формы калькулятора.

ЗАДАНИЕ 7

Реализовать галерею из 16 изображений одинакового размера в виде горизонтального слайдера. Всего на экране помещается четыре изображения, на смартфонах два. Галерея прокручивается влево и вправо без перезагрузки страницы кликом на иконки стрелок. Под галереей расположен «пейджер», отображающий количество страниц и текущую страницу.

ЗАДАНИЕ 8

Реализовать форму обратной связи с полями Имя, Email, сообщение, чекбокс согласия с политикой обработки персональных данных. Форма должна отображаться при клике на кнопку в попапе поверх страницы. На смартфоне форма должна полностью помещаться в экран без прокрутки. Внешний вид формы произвольный. При отображении попапа с формой необходимо менять URL страницы с помощью History API. Нажатие «Назад» в браузере должно закрывать форму. Данные из формы необходимо отправлять по нажатию кнопки Отправить без перезагрузки страницы с помощью XMLHttpRequest, jQuery AJAX или fetch и сохранять на сервер. Использовать сервер готового бекэнда для сохранения форм, например formcarry.com или slapform.com. После успешной отправки или ошибки отображать пользователю сообщение. Последние введенные в форму значения сохранять и восстанавливать при повторном открытии страницы с помощью LocalStorage API. При отправке формы очищать данные.

ПРОЕКТ

Обязательная часть проекта, выполняется самостоятельно или в команде 2 человека: Необходимо сверстать адаптивно (mobile-first) и реализовать на JavaScript веб-страницу, макеты которой находятся в архиве с заданием. Допускается использование других дизайнмакетов аналогичной сложности и с аналогичными функциями (выпадающее меню, форма, слайдер), в этом случае необходимо предоставить исходные дизайн-макеты на защите проекта. В файле page.png находится общий макет со всеми блоками страницы. В файле mobile.png находится макет страницы на смартфоне. В файле menu.png находится изображение выпадающего по наведению мышки меню. В файле mobile-menu.png находится изображение меню на мобильном устройстве. В файле fonts.txt находятся названия используемых на странице шрифтов. Шрифты необходимо найти и подключить самостоятельно. В шапке страницы отображается фоновое видео из файла video.mp4. Форма внизу страницы должна отправляться с помощью JavaScript без перезагрузки страницы на один из сервисов для сохранения форм из предыдущих заданий. Дополнительное задание повышенной сложности, выполняется обязательно в команде 2-3 человека (необходимо показать коммиты в git): Реализовать с помощью одного из популярных реактивных JS-фреймворков (React, Angular или Vue.js на выбор) отображение во всплывающем диалоговом окне контактной формы при нажатии на кнопку «связь с нами». Внешний вид и

поля формы аналогичны форме внизу страницы в дизайн-мекете. Реализовать анимацию отображения и сокрытия формы с помощью `RequestAnimationFrame`. Желательно учитывать положение нажатой кнопки на экране для иллюстрации плавного изменения состояния приложения с помощью анимации в ответ на действия пользователя. Использовать `LocalStorage` для сохранения введенных ранее значений и `History API` для работы с URL текущей страницы аналогично предыдущему заданию. Предполагается использование компонента роутинга для фреймворка. Для отправки запроса на сервер использовать `fetch`. Во время выполнения запроса на сервер блокировать кнопку отправки и отображать анимацию на кнопке или форме. В случае ошибки при отправке запроса отобразить сообщение об ошибке с просьбой отправить форму еще раз. Для меню сайта и меню на смартфонах реализовать анимацию плавного отображения и сокрытия с помощью CSS-переходов (`CSS-transition`), CSS-анимации или JavaScript. Допускается использование библиотек для реализации анимации. Желательно использование `Redux`, `MobX`, `VueX` или другой аналогичной библиотеки для управления состоянием. Возможно использование специальных библиотек и компонентов работы с формами, таких как, например, `Formik`, а также библиотек компонентов, таких как `Materialize`, `Vuetify` и т. д. Допускается использовать любые возможности и дополнительные модули выбранного фреймворка.

эффектам.

Зачетно-экзаменационные материалы для промежуточной аттестации (экзамен/зачет)

Вопросы для подготовки зачету

1. Структура HTML-документа. Теги `html`, `head`, `body`, `title`. Пример.
2. Теги HTML: абзац, заголовки, `div` и `span`. Пример.
3. Теги HTML: таблица, строки, ячейки, объединение ячеек. Пример.
4. URL и гиперссылки HTML. Пример.
5. Формы HTML: атрибуты `method` и `action`, элементы `submit` и простые текстовые поля `input`. Пример.
6. Формы HTML: радиокнопки и чекбоксы. Пример.
7. Формы HTML: многострочное текстовое поле, поле выбора даты. Пример.
8. Формы HTML: поле выбора из списка, множественный выбор. Пример.
9. CSS: правила каскада, источники и способы подключения стилей.
10. CSS: правила каскада, селекторы.
11. CSS: боксовая модель позиционирования.
12. CSS: `float`.
13. CSS: абсолютное и относительное позиционирование.
14. Основные свойства элементов CSS: цвет и размер шрифта, фон элементов.
15. CSS-переходы и CSS-анимация. Примеры.
16. Возможности отладки CSS в браузере.

17. Верстка макетов по сетке. CSS Flexbox.
18. Адаптивная верстка. CSS Media Queries.
19. CSS фреймворк Bootstrap.
20. БЭМ.
21. Подгружаемые шрифты CSS. Пример.
22. HTML5 Video. Пример.
23. Основы SEO: факторы ранжирования на поиске, поисковая оптимизация веб- страниц.
24. Семантическая верстка HTML5.
25. Структурированные данные schema.org.
26. Способы подключения JavaScript на страницу. Async и Defer.
27. Синтаксис основных операторов Javascript: создание переменных, условный оператор, циклы. Примеры.
28. Создание и использование функций в JavaScript. Анонимные функции. Примеры.
Встроенные математические функции.
29. Работа с DOM в JavaScript. Возможности встроенного объекта document. Примеры.
30. Обработка событий в JavaScript. Примеры.
31. Возможности отладки JavaScript в браузере.
32. Библиотека jQuery: подключение, примеры использования для поиска по DOM и изменения свойств элементов.
33. Массивы JavaScript. Создание и основные методы работы. Примеры.
34. Объекты JavaScript. Прототипы объектов. Примеры.
35. JSON: синтаксис, возможности, примеры использования.
36. Функция SetTimeout. Пример.
37. Функция setInterval. Пример.
38. XMLHttpRequest. Синхронная и асинхронная работа. Примеры.
39. Замыкания в JavaScript. Пример.
40. Область видимости переменных. Операторы var, let и const. Подъем (hoisting). Автоматическое управление памятью в Javascript.
41. Исключения в Javascript.
42. Немедленно выполняемое функциональное выражение (IIFE).
43. ООП в JavaScript: общие принципы ООП, литерал объекта, точечная и скобочная запись, пространства имен, использование функций в качестве классов, создание объектов, свойства и методы.
44. ООП в JavaScript: наследование с помощью Object.create, конструктор, Function.call. Использование this.
45. ООП в JavaScript: множественное наследование, подмешивания.
46. Встроенный объект location. History API.
47. LocalStorage и SessionStorage.
48. Workers.
49. WebSockets.

50. Обзор и сравнение возможностей Canvas, SVG и WebGL.
51. Анимация с помощью RequestAnimationFrame.
52. Стрелочные функции ES6. Функции map, reduce, filter. Примеры использования.
53. Классы в ES6. Наследование с помощью Extends. Методы прототипа и статические методы.
54. Promise. Примеры.
55. Fetch. Пример.
56. Async/await. Примеры.

10. Оценивание результатов обучения и уровня сформированности компетенций

Оценивание результатов обучения по дисциплине и уровня сформированности компетенций (части компетенции) осуществляется в рамках текущего контроля успеваемости и промежуточной аттестации в соответствии с оценочными материалами.

11. Учебно-методическое обеспечение дисциплины

Основная литература:

1. Малашкевич, В.Б. Интернет-программирование : лабораторный практикум /В.Б. Малашкевич ; Поволжский государственный технологический университет. -Йошкар-Ола : ПГТУ, 2017. - 96 с. : ил. - Библиогр.: с. 82. - ISBN 978-5-8158-1854-5 ; То же [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=476400>
2. Богданов, М.Р. Разработка клиентских приложений Web-сайтов : курс / М.Р. Богданов. - Москва : Интернет-Университет Информационных Технологий, 2010. -228 с. : ил. ; То же [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=233745>

Дополнительная литература:

1. Стандарты оформления исходного кода программ и современные интегрированные среды разработки программного обеспечения: учеб.-метод.пособие. Ю.В. Кольцов, А.В.Уварова, С.Г.Синица [и др.] – Краснодар: Кубанский гос.ун-т, 2017.
2. Громов Ю. Ю. , Иванова О. Г. , Данилкин С. В. Основы Web-инжиниринга : разработка клиентских приложений: учебное пособие Издательство ФГБОУ ВПО «ТГТУ», 2012. 240 стр.. URL: http://biblioclub.ru/index.php?page=book_red&id=277648&sr=1

Электронные библиотечные системы (ЭБС) и электронные образовательные ресурсы

1. Электронно-библиотечная система ZNANIUM <https://www.znanium.com>
2. Универсальная справочно-информационная полнотекстовая база данных периодических изданий «East View» (ИБИС) <https://dlib.eastview.com/>
3. Электронно-библиотечная система «BOOK.ru» www.book.ru
4. Электронно-библиотечная система «IPRbooks» <http://www.iprbookshop.ru/>
5. ЭБС «Национальный цифровой ресурс «Руконт» <https://lib.rucont.ru/search>
6. Образовательная платформа ЮРАЙТ <http://www.urait.ru>

Электронный каталог Университета – Автоматизированная информационная библиотечная система (АИБС «МегаПро»)

12. Перечень программного обеспечения, профессиональных баз данных и информационных справочных систем, необходимых для освоения дисциплины

Лицензионное программное обеспечение, в том числе отечественного производства:

- DsktpEdu ALNG LicSARk OLV F1 1Y Acdmc Ent (MS Windows) (подписка на ПО).

Свободно распространяемое программное обеспечение, в том числе отечественного производства:

- 7-Zip (свободный файловый архиватор с высокой степенью сжатия данных) (отечественное ПО).

- FastStone Image Viewer (программа для просмотра изображений в Microsoft Windows (лицензия бесплатного программного обеспечения)).

- Foxit Reader (Бесплатное прикладное программное обеспечение для просмотра электронных документов в стандарте PDF (лицензия бесплатного программного обеспечения)).

- Indigo (система программ для создания и проведения компьютерного тестирования знаний).

- Google Chrome, ЯндексБраузер (браузер).

- Adobe Acrobat Reader DC (• ПО для просмотра, печати и комментирования документов в формате PDF)

- Visual Studio Code (интегрированная среда разработки программного обеспечения)

Профессиональные базы данных не используются.

Информационные справочные системы:

- СПС КонсультантПлюс. Компьютерная справочная правовая система, широко используется учеными, студентами и преподавателями (подписка на ПО)

Каждый обучающийся в течение всего периода обучения обеспечивается индивидуальным неограниченным доступом к электронно-

13. Материально–техническое обеспечение дисциплины

Образовательный процесс обеспечен учебными аудиториями для проведения учебных занятий, а именно для проведения занятий лекционного типа, лабораторных занятий, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, помещениями для самостоятельной работы студентов.

Учебная аудитория для проведения занятий лекционного типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации.

Рабочее место преподавателя (стол, стул), рабочие места обучающихся, доска для написания мелом, кафедра.

Технические средства обучения: персональный компьютер-1, мультимедийное оборудование (проектор, проекционный экран) - 1

Учебная аудитория для проведения лабораторных занятий. Аудитория информационных технологий

Рабочее место преподавателя (стол, стул), рабочие места обучающихся:

Технические средства обучения:

персональный компьютер с подключением к сети «Интернет» - 15, сетевое оборудование.

Помещение для самостоятельной работы обучающихся, оснащенное компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду института.

Рабочие места обучающихся.

Технические средства обучения: персональный компьютер - 13.

Учебные аудитории соответствуют действующим противопожарным правилам и нормам, укомплектованы учебной мебелью.

Обновление рабочей программы дисциплины (модуля)

Наименование раздела рабочей программы, в который внесены изменения

(измененное содержание раздела)

обсуждено и рекомендовано к утверждению решением кафедры
наименование кафедры

от ____ ____ 20__ г., протокол № ____

одобрено Научно-методическим советом ____ от ____ ____ 20__ г.,
протокол № ____