



Федеральное государственное образовательное бюджетное учреждение высшего образования
«ФИНАНСОВЫЙ УНИВЕРСИТЕТ ПРИ ПРАВИТЕЛЬСТВЕ РОССИЙСКОЙ ФЕДЕРАЦИИ»
(Финансовый университет)

Кафедра информационных технологий
Факультет информационных технологий и анализа больших данных

Документ подписан усиленной неквалифицированной электронной подписью.

Организация: «ФИНАНСОВЫЙ УНИВЕРСИТЕТ ПРИ ПРАВИТЕЛЬСТВЕ
РОССИЙСКОЙ ФЕДЕРАЦИИ»

Сертификат: DbIPbR1CYH/R52SONNHMiOL9lnj9uG7

Утверждено: Проректор по учебной и методической работе, Е.А. Каменева

Дата: 09.02.2026 г.

Ю.С. Бузыкова

Кросс-платформенная разработка

Рабочая программа дисциплины

для студентов, обучающихся по направлению подготовки

09.03.04 - Программная инженерия,

Образовательная программа «Инженер-разработчик программного обеспечения»

Профиль:

«Инженер-разработчик программного обеспечения»

Рекомендовано

*Факультет информационных технологий и анализа больших данных
(протокол № 08 от 19.05.2026 г.)*

Одобрено

*Кафедра информационных технологий
(протокол № 03 от 05.05.2026 г.)*



Содержание

1. Наименование дисциплины	3
2. Перечень планируемых результатов освоения образовательной программы (перечень компетенций) с указанием индикаторов их достижения и планируемых результатов обучения по дисциплине	3
3. Место дисциплины в структуре образовательной программы	6
4. Объем дисциплины (модуля) в зачетных единицах и в академических часах с выделением объема аудиторной (лекции, семинары) и самостоятельной работы обучающихся	6
5. Содержание дисциплины, структурированное по темам (разделам) дисциплины с указанием их объемов (в академических часах) и видов учебных занятий	7
5.1. Содержание дисциплины	7
5.2. Учебно – тематический план	8
5.3. Содержание семинаров, практических занятий	9
6. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине	10
6.1. Перечень вопросов, отводимых на самостоятельное освоение дисциплины, формы внеаудиторной самостоятельной работы	10
6.2. Перечень вопросов, заданий, тем для подготовки к текущему контролю	11
7. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине	13
8. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины	6
9. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины	9
10. Методические указания для обучающихся по освоению дисциплины	24
11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень необходимого программного обеспечения и информационных справочных систем (при необходимости)	25
12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине	25



1. Наименование дисциплины

Кросс-платформенная разработка

2. Перечень планируемых результатов освоения образовательной программы (перечень компетенций) с указанием индикаторов их достижения и планируемых результатов обучения по дисциплине

Код компетенции	Наименование компетенции	Индикаторы достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции
ПКП-5	Способность проектировать и реализовывать интеллектуальные информационные системы	Демонстрирует знания основных методов машинного обучения и интеллектуального анализа данных, применяет готовые инструменты для создания интеллектуальных алгоритмов	Знать: основные стандарты описания архитектуры кросс-платформенных приложений, основные архитектурные шаблоны и модели Уметь: грамотно и понятно описывать используемую архитектуру приложения в виде технического документа
		Понимает особенности интеллектуальных информационных систем в части операций разработки, развертывания и сопровождения	Знать: особенности архитектуры и компонентов интеллектуальных информационных систем Уметь: анализировать и адаптировать существующие практики создания программных продуктов для интеллектуальных информационных систем
		Адаптирует практики создания программных продуктов, в том числе командные, для интеллектуальных информационных систем	Знать: принципы командной работы при создании программных продуктов для интеллектуальных информационных систем Уметь: организовывать командную работу при проектировании и реализации интеллектуальных информационных систем



Код компетенции	Наименование компетенции	Индикаторы достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции
		Организовывает сбор и подготовку данных для систем машинного обучения, в том числе потоковых, он-лайн обучения	Знать: методы сбора, хранения и обработки данных для систем машинного обучения, в том числе потоковых и он-лайн-обучения Уметь: собирать, обрабатывать и подготавливать данные для систем машинного обучения, в том числе потоковых и онлайн-обучения
ПКП-6	Способность вести разработку программных систем в команде, вести эффективную коммуникацию	Придерживается рекомендованного в команде стиля написания кода, проводит рефакторинг написанного кода, в том числе, другими членами команды, проводит код-ревью	Знать: основные понятия кросс-платформенной разработки, включая инструментальные средства программирования, языки программирования, сопутствующие библиотеки и фреймворки Уметь: применять данные средства на практике, иметь опыт их практического применения
		Документирует код, архитектуру, дизайн, требования к программной системе с использованием принятой в команде системы технических стандартов	Знать: принятые в командной разработке программного обеспечения системы технических стандартов для документирования кода, архитектуры, дизайна и требований к программной системе Уметь: документировать код, архитектуру, дизайн и требования к программной системе с использованием принятых в команде технических стандартов
		Использует инструментальные средства командной разработки программного обеспечения, контроля версий, отслеживания ошибок, планирования процессов разработки в	Знать: инструментальные средства командной разработки ПО, такие как системы контроля версий, отслеживания ошибок, планирования процессов разработки Уметь: эффективно взаимодействовать с членами ко-



Код компетенции	Наименование компетенции	Индикаторы достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции
		зависимости от принятой в команде методологии разработки	манды при разработке программных систем, в том числе при планировании процессов разработки
		Выстраивает эффективную двустороннюю коммуникацию с нетехническими специалистами по предмету разработки предметной области в целом	Знать: основные характеристики и возможности используемых инструментальных средств разработки кросс-платформенных приложений Уметь: выстраивать эффективную коммуникации внутри и вне команды разработки, включая обсуждение хода, целей и задач разработки
		Коммуницирует задачи разработки между членами команды, оценивает трудоемкость и сложность выполнения задач разработки, ставит задачи и контролирует их выполнение, в том числе в письменной формализованной форме	Знать: основные инструментальные средства организации командной работы, включая системы контроля версий, отслеживания ошибок Уметь: организовывать и участвовать в работе в команде по разработке кросс-платформенного программного продукта с использованием инструментальных средств автоматизации командной работы



3. Место дисциплины в структуре образовательной программы

Дисциплина «Кросс-платформенная разработка» относится к «Профилю "Инженер-разработчик программного обеспечения"».

4. Объем дисциплины (модуля) в зачетных единицах и в академических часах с выделением объема аудиторной (лекции, семинары) и самостоятельной работы обучающихся

очно-заочная форма обучения

Вид учебной работы по дисциплине	Всего (в з/е и часах)	Семестр 7 (в часах)
Общая трудоемкость дисциплины	5/180	180
Контактная работа – Аудиторные занятия	36	36
<i>Лекции</i>	<i>8</i>	<i>8</i>
<i>Семинары, практические занятия</i>	<i>28</i>	<i>28</i>
<i>Самостоятельная работа</i>	<i>144</i>	<i>144</i>
Вид текущего контроля	Контрольная работа	Контрольная работа
Вид промежуточной аттестации	Экзамен	Экзамен



5. Содержание дисциплины, структурированное по темам (разделам) дисциплины с указанием их объемов (в академических часах) и видов учебных занятий

5.1. Содержание дисциплины

Тема 1. Основные понятия кросс-платформенной разработки программного обеспечения

Разработка ПО для конкретной операционной системы. Совместимость операционных систем. Семейства и виды операционных систем. Совместимость на уровне машинного кода, байт-кода, исходного текста, API операционных систем. Инструментальные средства обеспечения кросс-платформенности программ.

Тема 2. Кросс-платформенная разработка десктопных программ

Основные инструментальные средства разработки кросс-платформенных программ для десктопных операционных систем. POSIX-совместимое программирование. Фреймворк Electron.

Тема 3. Использование веб-технологий для кросс-платформенной разработки

Веб-технологии как интегративная среда программирования. Условия доступности веб-приложений. Обзор современных средств программирования. Прогрессивные веб-приложения, Web-сокеты. Web-assembly. Назначение, применение, практикум работы в данных технологиях.

Тема 4. Кросс-платформенная разработка мобильных программ

Особенности создания мобильных приложений для iOS и Android. Специфика мобильных операционных систем. Виды мобильных приложений: нативные, веб, гибридные. Обзор современных фреймворков создания гибридных мобильных приложений. Практикум работы в фреймворках гибридных мобильных приложений.



5.2. Учебно – тематический план

Очно-заочная форма обучения

п/ п	Наименование тем (разделов) дисциплины	Трудоемкость в часах				
		Всего	Контактная работа - Аудиторная работа			Самостоятельная работа
			Общая, в т.ч.:	Лекции	Семинары, практические занятия	
1	Основные понятия кросс-платформенной разработки программного обеспечения	42	8	2	6	34
2	Кросс-платформенная разработка десктопных программ	43	10	2	8	33
3	Использование веб-технологий для кросс-платформенной разработки	41	8	2	6	33
4	Кросс-платформенная разработка мобильных программ	54	10	2	8	44
В целом по дисциплине		180	36	8	28	144



5.3. Содержание семинаров, практических занятий

Наименование тем (разделов) дисциплины	Перечень вопросов для обсуждения на семинарах, практических занятиях	Формы проведения занятий
Основные понятия кросс-платформенной разработки программного обеспечения	Знакомство с инструментальными средствами разработки кросс-платформенных программных продуктов.	Лабораторная работа, устный опрос, тестирование.
Кросс-платформенная разработка десктопных программ	Практикум работы с фреймворком Electron.	Лабораторная работа, устный опрос, тестирование.
Использование веб-технологий для кросс-платформенной разработки	Практикум разработки прогрессивных веб-приложений.	Лабораторная работа, устный опрос, тестирование.
Кросс-платформенная разработка мобильных программ	Практика работы с кросс-платформенными фреймворками создания гибридных мобильных приложений. React Native.	Лабораторная работа, устный опрос, тестирование.



6. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине

6.1. Перечень вопросов, отводимых на самостоятельное освоение дисциплины, формы внеаудиторной самостоятельной работы

Наименование тем (разделов) дисциплины	Перечень вопросов, отводимых на самостоятельное освоение	Формы внеаудиторной самостоятельной работы
Основные понятия кросс-платформенной разработки программного обеспечения	Создание POSIX-совместимых программных продуктов. Интерфейсы системных вызовов.	Изучение литературы, самостоятельное решение задач, подготовка к лабораторной работе.
Кросс-платформенная разработка десктопных программ	Технологии контейнеризации десктопных программных продуктов. Использование Docker для разработки переносимых программ.	Изучение литературы, самостоятельное решение задач, подготовка к лабораторной работе.
Использование веб-технологий для кросс-платформенной разработки	PWA, Web-assembly. Общая характеристика, назначение, основные приемы работы.	Изучение литературы, самостоятельное решение задач, подготовка к лабораторной работе.
Кросс-платформенная разработка мобильных программ	Технологии контейнеризации мобильных приложений. Apache Cordova, Flutter, Ionic.	Изучение литературы, самостоятельное решение задач, подготовка к лабораторной работе.



6.2. Перечень вопросов, заданий, тем для подготовки к текущему контролю

Примерные вопросы для подготовке к контрольной работе

1. Специфика архитектуры современных веб-приложений.
2. Современные библиотеки построения адаптивных интерфейсов веб-приложений.
3. Особенности механизма системных вызовов в операционных системах семейства Windows.
4. Особенности механизма системных вызовов в операционных системах семейства Linux.
5. Инструменты создания гибридных мобильных приложений.

Примерные задания контрольной работы

1. Спроектировать и реализовать кросс-платформенное приложения для решения задачи классификации заданного набора данных.
2. Спроектировать кросс-платформенное приложения на основе веб-технологий для доступа к базе данных с заданной структурой и описать информационную инфраструктуру, необходимую для его развертывания.
3. Реализовать в команде интеллектуальное кросс-платформенное приложение на основе веб-технологий для распознавания образов на фотоизображении.
4. Реализовать мобильное кросс-платформенное приложение обработки потоковых видеоданных.
5. Провести ревизию кода кросс-платформенного приложения, реализованного другим студентом.
6. Спроектировать и описать документально архитектуру кросс-платформенного мобильного приложения с использованием фреймворка NaMeR.
7. Использовать систему контроля версий для разработки кросс-платформенного приложения.
8. Реализовать в команде и задокументировать интеллектуальное кросс-платформенное приложение на основе веб-технологий для машинного обучения на основе методов регрессионного анализа.
9. Реализовать в команде интеллектуальное кросс-платформенное приложение на основе веб-технологий для распознавания речи.

Дополнительная информация



Критерии балльной оценки различных форм текущего контроля успеваемости содержатся в соответствующих методических рекомендациях Кафедры информационных технологий Факультета информационных технологий и анализа больших данных.



7. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине

Перечень компетенций с указанием индикаторов их достижения в процессе освоения образовательной программы содержится в разделе 2. «Перечень планируемых результатов освоения образовательной программы (перечень компетенций) с указанием индикаторов их достижения и планируемых результатов обучения по дисциплине».

Типовые контрольные задания или иные материалы, необходимые для оценки индикаторов достижения компетенций, умений и знаний

Наименование компетенции	Наименование индикаторов достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции	Типовые контрольные задания
ПКП-5. Способность проектировать и реализовывать интеллектуальные информационные системы	Демонстрирует знания основных методов машинного обучения и интеллектуального анализа данных, применяет готовые инструменты для создания интеллектуальных алгоритмов	Знать: основные стандарты описания архитектуры кросс-платформенных приложений, основные архитектурные шаблоны и модели Уметь: грамотно и понятно описывать используемую архитектуру приложения в виде технического документа	Спроектировать и реализовать кросс-платформенное приложения для решения задачи классификации заданного набора данных.



Наименование компетенции	Наименование индикаторов достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции	Типовые контрольные задания
	Понимает особенности интеллектуальных информационных систем в части операций разработки, развертывания и сопровождения	Знать: особенности архитектуры и компонентов интеллектуальных информационных систем Уметь: анализировать и адаптировать существующие практики создания программных продуктов для интеллектуальных информационных систем	Спроектировать кросс-платформенное приложения на основе веб-технологий для доступа к базе данных с заданной структурой и описать информационную инфраструктуру, необходимую для его развертывания.
	Адаптирует практики создания программных продуктов, в том числе командные, для интеллектуальных информационных систем	Знать: принципы командной работы при создании программных продуктов для интеллектуальных информационных систем Уметь: организовывать	Реализовать в команде интеллектуальное кросс-платформенное приложение на основе веб-технологий для распознавания речи.



Наименование компетенции	Наименование индикаторов достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции	Типовые контрольные задания
		вать командную работу при проектировании и реализации интеллектуальных информационных систем	
	Организовывает сбор и подготовку данных для систем машинного обучения, в том числе потоковых, онлайн обучения	Знать: методы сбора, хранения и обработки данных для систем машинного обучения, в том числе потоковых и онлайн-обучения Уметь: собирать, обрабатывать и подготавливать данные для систем машинного обучения, в том числе потоковых и онлайн-обучения	Реализовать в команде и задокументировать интеллектуальное кросс-платформенное приложение на основе веб-технологий для машинного обучения на основе методов регрессионного анализа.



Наименование компетенции	Наименование индикаторов достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции	Типовые контрольные задания
ПКП-6. Способность вести разработку программных систем в команде, вести эффективную коммуникацию	Придерживается рекомендаций в команде стиля написания кода, проводит рефакторинг написанного кода, в том числе, другими членами команды, проводит код-ревью	Знать: основные понятия кросс-платформенной разработки, включая инструменты программирования, языки программирования, сопутствующие библиотеки и фреймворки Уметь: применять данные средства на практике, иметь опыт их практического применения	Реализовать в команде и задокументировать интеллектуальное кросс-платформенное приложение на основе веб-технологий для машинного обучения на основе методов регрессионного анализа.
	Документирует код, архитектуру, дизайн, требования к программной системе с использованием принятой в команде си-	Знать: принятые в командной разработке программного обеспечения системы технических стандартов для документиро-	Особенности механизма системных вызовов в операционных системах семейства Linux.



Наименование компетенции	Наименование индикаторов достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции	Типовые контрольные задания
	стемы технических стандартов	вания кода, архитектуры, дизайна и требований к программной системе Уметь: документировать код, архитектуру, дизайн и требования к программной системе с использованием принятых в команде технических стандартов	
	Использует инструментальные средства командной разработки программного обеспечения, контроля версий, отслеживания ошибок, планирования процессов разработки в	Знать: инструментальные средства командной разработки ПО, такие как системы контроля версий, отслеживания ошибок, планирования процессов разработки	Инструментальные средства обеспечения кросс-платформенности.



Наименование компетенции	Наименование индикаторов достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции	Типовые контрольные задания
	зависимости от принятой в команде методологии разработки	Уметь: эффективно взаимодействовать с членами команды при разработке программных систем, в том числе при планировании процессов разработки	
	Выстраивает эффективную двустороннюю коммуникацию с нетехническими специалистами по предмету разработки предметной области в целом	Знать: основные характеристики и возможности используемых инструментальных средств разработки кросс-платформенных приложений Уметь: выстраивать эффективную коммуникации внутри и вне команды разработки, включая обсуждение	Реализовать в команде интеллектуальное кросс-платформенное приложение на основе веб-технологий для распознавания речи.



Наименование компетенции	Наименование индикаторов достижения компетенции	Результаты обучения (умения и знания), соотнесенные с индикаторами достижения компетенции	Типовые контрольные задания
		хода, целей и задач разработки	
	Коммуницирует задачи разработки между членами команды, оценивает трудоемкость и сложность выполнения задач разработки, ставит задачи и контролирует их выполнение, в том числе в письменной формализованной форме	Знать: основные инструментальные средства организации командной работы, включая системы контроля версий, отслеживания ошибок Уметь: организовывать и участвовать в работе в команде по разработке кросс-платформенного программного продукта с использованием инструментальных средств автоматизации командной работы	Реализовать в команде и задокументировать интеллектуальное кросс-платформенное приложение на основе веб-технологий для машинного обучения на основе методов регрессионного анализа.



Примеры практико-ориентированных заданий

Задание 1. Анализ совместимости ПО

Студенту предоставляется описание существующего приложения (например, десктопное приложение на C++ для Windows, использующее WinAPI). Необходимо определить уровень совместимости (машинный код, байт-код, исходный код, API) и предложить два варианта портирования на Linux с обоснованием трудозатрат и рисков. Результат оформляется в виде технической записки, включающей сравнительную таблицу инструментов для кросс-платформенной разработки (CMake, POSIX-обертки, контейнеризация).

Задание 2. Разработка POSIX-совместимого модуля

Требуется реализовать модуль на C/C++, который считывает содержимое директории с помощью POSIX-функций (`opendir`, `readdir`) и выводит список файлов с указанием их размера. Модуль должен быть скомпилирован и проверен в Ubuntu (gcc) и Windows (MinGW или WSL). К работе прилагаются исходный код, Makefile/CMakeLists.txt и скриншоты выполнения на обеих операционных системах.

Задание 3. Создание десктопного приложения на Electron

Разработать кросс-платформенное десктопное приложение для управления заметками (Markdown-редактор) с использованием Electron. Приложение должно позволять создавать, редактировать и удалять заметки, сохранять файлы в локальной файловой системе, поддерживать горячие клавиши (Ctrl+N, Ctrl+S). Результат: исходный код, установочные пакеты для Windows и Linux (или macOS), краткая инструкция по сборке.

Задание 4. Контейнеризация десктопного приложения

Взять простое GUI-приложение (например, на Python с Tkinter или Qt) и обеспечить его запуск в Docker с пробросом дисплея (X11 или Wayland). Необходимо написать Dockerfile, создать скрипт для автоматической сборки и запуска контейнера, протестировать работу на двух разных дистрибутивах Linux (например, Ubuntu и Fedora). Результат: Dockerfile, скрипт запуска, отчет о совместимости.

Задание 5. Разработка прогрессивного веб-приложения (PWA)

Создать PWA «Трекер задач», работающее офлайн (Service Worker, Cache API), с возможностью установки на рабочий стол (Web App Manifest). Требуется реализовать синхронизацию данных при восстановлении соединения (Background Sync) и адаптивный дизайн для мобильных и десктопных экранов. Результат: репозиторий с кодом, демонстрация работы в офлайн-режиме, установка приложения на Android и Windows.

Задание 6. Интеграция WebAssembly в веб-приложение



Реализовать веб-приложение для обработки изображений (например, фильтр «размытие»). Написать функцию фильтра на C++, скомпилировать в WebAssembly с помощью Emscripten, создать интерфейс на JavaScript/React для загрузки изображения, вызова WebAssembly-модуля и отображения результата. Результат: исходный код, демонстрация производительности по сравнению с чисто JavaScript-реализацией.

Задание 7. Гибридное мобильное приложение на React Native

Разработать мобильное приложение «Каталог фильмов» с использованием React Native (или Flutter). Приложение должно получать данные из открытого API (например, TMDb), отображать список фильмов с пагинацией, обеспечивать детальный просмотр и добавление в избранное с сохранением в локальном хранилище (AsyncStorage или SQLite). Требуется сборка и тестирование на эмуляторах iOS и Android или на реальных устройствах. Результат: исходный код, APK-файл для Android, скриншоты работы на iOS и Android.

Задание 8. Сравнительный анализ фреймворков гибридной разработки

Команда из 2–3 человек реализует MVP одного и того же приложения (например, чат-бот для заказа пиццы) на двух разных фреймворках: React Native и Flutter. По итогам выполняется сравнение производительности (время запуска, отзывчивость UI), сложности разработки, доступности библиотек, размера итогового приложения и удобства сборки под iOS/Android. Результат: два работающих приложения, отчет с анализом и рекомендацией по выбору технологии.

Задание 9. Командная разработка интеллектуального кросс-платформенного приложения

Команда (3–4 человека) разрабатывает кросс-платформенное приложение для распознавания речи (или классификации изображений) с использованием веб-технологий или гибридных фреймворков. Этапы: сбор данных и обучение модели (использование предобученной модели TensorFlow.js или Core ML), проектирование архитектуры с оформлением технического документа и диаграмм, реализация с использованием Git и проведением код-ревью, документирование (комментарии, README, инструкция по развертыванию), презентация с демонстрацией работы на разных платформах (веб, Android, Windows). Результат: репозиторий с историей коммитов, техническая документация, демонстрационный ролик.

Задание 10. Рефакторинг и кросс-ревью

Каждому студенту выдается проект кросс-платформенного приложения (предыдущего студента или открытый исходный код), содержащий стилистические ошибки, платформозависимые участки (жестко прописанные пути, использование ОС-специфичных API) и отсутствие документации. Задание: провести код-ревью с оформлением замечаний, выполнить рефакторинг для повышения кросс-платформенности, добавить документацию (JSDoc, Doxygen или аналоги), создать pull



request с пояснением внесенных изменений. Результат: оформленный pull request, список замечаний до и после рефакторинга, отчет о проделанной работе.

Примеры вопросов для подготовки к промежуточной аттестации

Примерные вопросы для подготовки к зачету

1. Семейства и виды операционных систем. Механизм системных вызовов.
2. Совместимость программ на уровне операционных систем.
3. Аппаратные и программные платформы.
4. Особенности создания десктопных приложений для операционных систем семейства Windows.
5. Особенности создания десктопных приложений для операционных систем семейства Linux.
6. Виртуальные машины Java. Байт код как средство обеспечения кросс-платформенности.
7. Виртуальная машина LLVM. Особенности работы, применимость, общая характеристика, совместимость с языками программирования.
8. Системные вызовы операционных систем. API операционных систем.
9. Понятие POSIX-совместимости.
10. Фреймворк Electron как средство создания кросс-платформенных десктопных приложений.
11. Веб-технологии как интегративная среда программирования.
12. Прогрессивные веб-приложения.
13. Современные инструменты создания веб-приложений.
14. Современные системы развертывания веб-приложений.
15. Инструментальные средства обеспечения кросс-платформенности.
16. Особенности мобильных операционных систем.
17. Современные средства создания гибридных мобильных приложений.
18. Основные проблемы кросс-платформенного программирования.
19. Кросс-платформенность в скриптовых и компилируемых языках программирования.
20. Среда выполнения NET как средство обеспечения кросс-платформенности.

** Промежуточная аттестация обучающихся проводится в формате тестирования*



8. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины

Основная литература:

1. Рочев, К.В. Разработка мобильных и кроссплатформенных приложений : Учебник / К.В. Рочев Электрон. дан. Москва : Русайнс, 2025 247 с. Режим доступа: book.ru Internet access <https://book.ru/book/960835> ISBN 978-5-466-10755-5. [БИК ID: RU\bookru\bibl\960835]
2. Операционные системы : Учебное пособие / Воронежский институт Федеральной службы исполнения наказаний России ; Воронежский государственный технический университет Воронеж : Издательско-полиграфический центр "Научная книга", 2022 92 с. ВО - Специалитет <https://znanium.com/catalog/document?id=426496>. [БИК ID: RU\infra-m\znanium\bibl\1999933]

Дополнительная литература:

1. Королькова, Т. В. Операционные системы. Основы работы в операционных системах семейства GNU/Linux. Практикум, направления подготовки: 02.03.02, 09.03.01, 09.03.02, 09.03.03, 09.03.04 [Электронный ресурс] / Королькова Т. В.,Алексян Д. А. Москва : МТУСИ, 2025 28 с. Книга из коллекции МТУСИ - Информатика <https://e.lanbook.com/book/501230>. [БИК ID: RU-LAN-BOOK-501230]

Нормативные правовые акты:

-

9. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

1. Электронная библиотека Финансового университета (ЭБ) <http://elib.fa.ru/> (<http://library.fa.ru/files/elibfa.pdf>)
2. Электронно-библиотечная система BOOK.RU <http://www.book.ru>
3. Электронно-библиотечная система "Университетская библиотека ОНЛАЙН" <http://biblioclub.ru/>
4. Электронно-библиотечная система Znanium <http://www.znanium.com>
5. Электронно-библиотечная система издательства "Лань" <https://e.lanbook.com/>
6. Образовательная платформа "ЮРАЙТ" <https://urait.ru/>
7. Научная электронная библиотека eLibrary.ru <http://elibrary.ru>
8. Информационно-образовательный портал Финуниверситета: <https://org.fa.ru>



10. Методические указания для обучающихся по освоению дисциплины

При изучении теоретического материала необходимо опираться на рабочую программу дисциплины, материалы лекций и литературу из основного списка. Кроме этого, необходимо активно работать с Интернет-источниками и пособиями других авторов, помогающими усвоить материал отдельных разделов программы.

Необходимо конспектировать лекции, помечая сложные и непонятные моменты с тем, чтобы задать вопросы лектору в конце лекции или же на консультации.

При подготовке к семинарским занятиям необходимо изучить вопросы, вынесенные на самостоятельное изучение, так как семинарские занятия предполагают их обсуждение и дискуссию по теме; кроме того, задания для самостоятельной работы необходимы для того, чтобы успешно выполнить самостоятельные задания на семинарах.

Индивидуальные задания для работы на компьютере, файлы с выполненными заданиями необходимо хранить в личной сетевой папке в компьютерной сети вуза.

При подготовке контрольной работы следует использовать нормативные документы Финансового университета, Методические рекомендации по планированию и организации внеаудиторной самостоятельной работы студентов по образовательным программам бакалавриата и магистратуры в Финансовом университете, утвержденные приказом Финуниверситета от 11.05.2021 г. № 1040/о (см. сайт Финансового Университета: на главной странице раздел "Наш университет" → "Единая правовая база Финуниверситета" → "Организация учебного процесса" → "Нормативные документы по самостоятельной работе").



11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень необходимого программного обеспечения и информационных справочных систем

Комплект лицензионного программного обеспечения:

1. Пакет офисных программ
2. Антивирус Kaspersky
3. Комплект свободно распространяемого программного обеспечения

Современные профессиональные базы данных и информационные справочные системы

1. Информационно-правовая система «Гарант»
2. Информационно-правовая система «Консультант Плюс»
3. Электронная энциклопедия: <http://ru.wikipedia.org/wiki/Wiki>
4. Система комплексного раскрытия информации «СКРИН» - <http://www.skrin.ru/>

Сертифицированные программные и аппаратные средства защиты информации

1. не предусмотрены

12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине

1. Учебная аудитория для проведения учебных занятий, предусмотренных программой, в том числе групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, оснащенная оборудованием и техническими средствами обучения: мебель аудиторная (столы, стулья, доска аудиторная), набор демонстрационного оборудования (проектор, экран)