

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ
федеральное государственное автономное образовательное учреждение высшего
образования
"САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ"

Кафедра № 41

УТВЕРЖДАЮ
Руководитель образовательной программы

доц., к.т.н.
(должность, уч. степень, звание)

С.А. Чернышев
(инициалы, фамилия)

(подпись)
«20» 02 2026 г

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

«Kotlin и Kotlin Compose Multiplatform»
(Наименование дисциплины)

Код направления подготовки/ специальности	09.04.03
Наименование направления подготовки/ специальности	Прикладная информатика
Наименование направленности/ специализации	Разработка кроссплатформенных систем с использованием искусственного интеллекта
Форма обучения	заочная
Год приема	2026

Лист согласования рабочей программы дисциплины

Программу составил (а)

доцент, к.т.н.
(должность, уч. степень, звание)

(подпись, дата)

С.А. Чернышев
(инициалы, фамилия)

Программа одобрена на заседании кафедры № 41

«16» 02 2026 г, протокол № 02-2026/р6

Зам. Заведующий кафедрой № 41

д.т.н., проф.
(уч. степень, звание)

(подпись, дата)

Г.А. Коржавин
(инициалы, фамилия)

Заместитель директора института №4 по методической работе

доц., к.т.н.
(должность, уч. степень, звание)

(подпись, дата)

А.А. Фоменкова
(инициалы, фамилия)

Аннотация

Дисциплина «Kotlin и Kotlin Compose Multiplatform» входит в образовательную программу высшего образования – программу магистратуры по направлению подготовки/специальности 09.04.03 «Прикладная информатика » направленности/специализации «Разработка кроссплатформенных систем с использованием искусственного интеллекта». Дисциплина реализуется кафедрой «№41».

Дисциплина нацелена на формирование у выпускника следующих компетенций:

ПК-1 «Способность применять современные методы и инструментальные средства прикладной информатики для автоматизации и информатизации решения прикладных задач различных классов и создания (модификации) кроссплатформенных информационных систем»

ПК-7 «Способность использовать технологии и инструменты кроссплатформенной разработки для создания программных компонентов информационных систем»

Содержание дисциплины охватывает круг вопросов, связанных с использованием языка Kotlin и технологии Compose Multiplatform для проектирования, разработки, тестирования, отладки и сопровождения кроссплатформенных информационных систем, включая организацию Kotlin Multiplatform-проектов, построение общего пользовательского интерфейса, управление состоянием, работу с данными, интеграцию с платформенными API, сборку и публикацию приложений.

Преподавание дисциплины предусматривает следующие формы организации учебного процесса: лекции, практические занятия, самостоятельная работа обучающегося.

Программой дисциплины предусмотрены следующие виды контроля: текущий контроль успеваемости, промежуточная аттестация в форме экзамена (3 семестр).

Общая трудоемкость освоения дисциплины составляет 4 зачетных единицы, 144 часа.

Язык обучения по дисциплине «русский»

1. Перечень планируемых результатов обучения по дисциплине

1.1. Цели преподавания дисциплины

Целью преподавания дисциплины является формирование у обучающихся знаний, умений и практических навыков в области разработки кроссплатформенных приложений и программных компонентов информационных систем на Kotlin и Compose Multiplatform, необходимых для выбора архитектурных решений, реализации общего и платформенно-специфичного кода, тестирования, отладки и сопровождения программного обеспечения. Дисциплина создает условия для применения современных инструментальных средств разработки, включая ИИ-ассистентов, при решении прикладных задач в области кроссплатформенных систем.

1.2. Дисциплина входит в состав части, формируемой участниками образовательных отношений, образовательной программы высшего образования (далее – ОП ВО).

1.3. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения ОП ВО.

В результате изучения дисциплины обучающийся должен обладать следующими компетенциями или их частями. Компетенции и индикаторы их достижения приведены в таблице 1.

Таблица 1 – Перечень компетенций и индикаторов их достижения

Категория (группа) компетенции	Код и наименование компетенции	Код и наименование индикатора достижения компетенции
Профессиональные компетенции	ПК-1 Способность применять современные методы и инструментальные средства прикладной информатики для автоматизации и информатизации решения прикладных задач различных классов и создания (модификации) кроссплатформенных информационных систем	ПК-1.3.1 знает основы применения современных методов и инструментальных средств прикладной информатики для автоматизации и информатизации решения прикладных задач различных классов, а также создания (модификации) кроссплатформенных информационных систем ПК-1.У.1 умеет оперировать современными методами и инструментальными средствами прикладной информатики с использованием искусственного интеллекта при создании (модификации) и сопровождении кроссплатформенных информационных систем ПК-1.В.1 владеет навыками и опытом работы с инструментальными средствами прикладной информатики (ИИ-агентами), предназначенными для автоматизации и информатизации решения прикладных задач и создания (модификации) кроссплатформенных информационных систем
Профессиональные компетенции	ПК-7 Способность использовать технологии и инструменты кроссплатформенной разработки для создания программных	ПК-7.3.1 знает принципы, архитектурные подходы и ключевые характеристики современных технологий и инструментов кроссплатформенной разработки ПК-7.У.1 умеет применять технологии и инструменты кроссплатформенной разработки для реализации, тестирования и отладки программных компонентов

	компонентов информационных систем	ПК-7.В.1 владеет навыками создания программных компонентов информационных систем с использованием фреймворков кроссплатформенной разработки
--	-----------------------------------	---

2. Место дисциплины в структуре ОП

Дисциплина может базироваться на знаниях, ранее приобретенных обучающимися при изучении следующих дисциплин:

- «Совр техн. разраб. ПО»,

3. Объем и трудоемкость дисциплины

Данные об общем объеме дисциплины, трудоемкости отдельных видов учебной работы по дисциплине (и распределение этой трудоемкости по семестрам) представлены в таблице 2.

Таблица 2 – Объем и трудоемкость дисциплины

Вид учебной работы	Всего	Трудоемкость по семестрам
		№3
1	2	3
Общая трудоемкость дисциплины, ЗЕ/ (час)	4/ 144	4/ 144
Из них часов практической подготовки	8	8
Аудиторные занятия, всего час.	16	16
в том числе:		
лекции (Л), (час)	8	8
практические/семинарские занятия (ПЗ), (час)	8	8
лабораторные работы (ЛР), (час)		
курсовой проект (работа) (КП, КР), (час)		
экзамен, (час)	9	9
Самостоятельная работа, всего (час)	119	119
Вид промежуточной аттестации: зачет, дифф. зачет, экзамен (Зачет, Дифф. зач, Экз.)	Экз.,	Экз.,

4. Содержание дисциплины

4.1. Распределение трудоемкости дисциплины по разделам и видам занятий.

Разделы, темы дисциплины и их трудоемкость приведены в таблице 3.

Таблица 3 – Разделы, темы дисциплины, их трудоемкость

Разделы, темы дисциплины	Лекции (час)	ПЗ (СЗ) (час)	ЛР (час)	КП/КР (час)	СР (час)
Семестр 3					
Раздел 1. Язык Kotlin как основа кроссплатформенной разработки Тема 1.1. Синтаксис Kotlin, null-safety, функции высшего порядка, data/sealed/enum классы Тема 1.2. Коллекции, extension-функции, DSL-элементы языка, coroutines и Flow	1	1			20

Раздел 2. Архитектура Kotlin Multiplatform-проекта Тема 2.1. Gradle Kotlin DSL, source sets, commonMain/platformMain, expect/actual Тема 2.2. Организация модулей, зависимости, ресурсы и межплатформенное API	1	1			30
Раздел 3. Compose Multiplatform: построение интерфейсов и управление состоянием Тема 3.1. @Composable, layout, Material 3, темы и ресурсы Тема 3.2. Состояние, remember, StateFlow, навигация, preview и hot reload	2	2			20
Раздел 4. Работа с данными, платформенными API и тестирование Тема 4.1. Ktor, kotlinx.serialization, локальное хранение данных, DI Тема 4.2. Интеграция с нативными API, отладка, unit- и UI-тестирование	2	2			20
Раздел 5. Сборка, публикация и сопровождение кроссплатформенных приложений Тема 5.1. Build variants, packaging, iOS integration, web preview Тема 5.2. CI/CD, code review, ИИ-ассистенты, обеспечение качества и сопровождение	2	2			29
Итого в семестре:	8	8			119
Итого	8	8	0	0	119

Практическая подготовка заключается в непосредственном выполнении обучающимися определенных трудовых функций, связанных с будущей профессиональной деятельностью.

4.2. Содержание разделов и тем лекционных занятий.

Содержание разделов и тем лекционных занятий приведено в таблице 4.

Таблица 4 – Содержание разделов и тем лекционного цикла

Номер раздела	Название и содержание разделов и тем лекционных занятий
1	Раздел 1. Язык Kotlin как основа кроссплатформенной разработки Тема 1.1. Базовые и составные типы данных Kotlin, null-safety, функции высшего порядка, коллекции и extension-функции Тема 1.2. Объектная модель языка, data/sealed/enum классы, coroutines, Flow и их роль при создании кроссплатформенных систем
2	Раздел 2. Архитектура Kotlin Multiplatform-проекта Тема 2.1. Структура KMP-проекта, Gradle Kotlin DSL, source sets, commonMain, androidMain, iosMain и desktopMain

	Тема 2.2. Механизм expect/actual, организация зависимостей, ресурсов и платформенно-специфичного кода
3	<p>Раздел 3. Compose Multiplatform: построение интерфейсов и управление состоянием</p> <p>Тема 3.1. Принципы декларативного UI, composable-функции, layout-модели, Material 3, темы и ресурсы</p> <p>Тема 3.2. Управление состоянием, remember/rememberSaveable, StateFlow, навигация и повторное использование UI-компонентов</p>
4	<p>Раздел 4. Работа с данными, платформенными API и тестирование</p> <p>Тема 4.1. Сетевое взаимодействие с Ktor, сериализация с kotlinx.serialization, локальное хранение и DI</p> <p>Тема 4.2. Интеграция с платформенными API, профилирование, отладка, unit-, UI- и интеграционные тесты</p>
5	<p>Раздел 5. Сборка, публикация и сопровождение кроссплатформенных приложений</p> <p>Тема 5.1. Подготовка проектов к сборке и выпуску, особенности Android, Desktop, iOS и Web</p> <p>Тема 5.2. CI/CD, код-ревью, использование ИИ-ассистентов, стандарты качества и сопровождение приложений</p>

4.3. Практические (семинарские) занятия

Темы практических занятий и их трудоемкость приведены в таблице 5.

Таблица 5 – Практические занятия и их трудоемкость

№ п/п	Темы практических занятий	Формы практических занятий	Трудоемкость, (час)	Из них практической подготовки, (час)	№ раздела дисциплины
Семестр 3					
1	Создание Kotlin Multiplatform-проекта в IDE, настройка Gradle Kotlin DSL, source sets и запуск Android/Desktop-целей	Индивидуальное практическое задание	1	1	1, 2
2	Реализация общего shared-модуля: null-safety, data/sealed-классы, coroutines/Flow, сериализация данных и сетевой клиент на Ktor	Практикум по программированию	1	1	1, 2, 4
3	Построение	Практикум по	2	2	3

	интерфейса на Compose Multiplatform: composable-функции, layout, Material 3, ресурсы и переиспользуемые компоненты	разработке UI			
4	Управление состоянием, навигация между экранами, локальное хранение данных и интеграция общего кода с платформенными API	Кейс и программирование	2	2	2, 3, 4
5	Мини-проект: отладка, рефакторинг, тестирование и подготовка Compose Multiplatform-приложения к сборке и защите	Проектная работа	2	2	1, 2, 3, 4, 5
Всего			8		

4.4. Лабораторные занятия

Темы лабораторных занятий и их трудоемкость приведены в таблице 6.

Таблица 6 – Лабораторные занятия и их трудоемкость

№ п/п	Наименование лабораторных работ	Трудоемкость, (час)	Из них практической подготовки, (час)	№ раздела дисциплины
Учебным планом не предусмотрено				
Всего				

4.5. Выполнение курсового проекта/ курсовой работы

Учебным планом не предусмотрено

4.6. Самостоятельная работа обучающихся

Виды самостоятельной работы и ее трудоемкость приведены в таблице 7.

Таблица 7 – Виды самостоятельной работы и ее трудоемкость

Вид самостоятельной работы	Всего, час	Семестр 3, час
1	2	3
Изучение теоретического материала дисциплины (ТО)	80	80
Курсовое проектирование (КП, КР)		
Расчетно-графические задания (РГЗ)		
Выполнение реферата (Р)		
Подготовка к текущему контролю успеваемости (ТКУ)	10	10
Домашнее задание (ДЗ)		
Контрольные работы заочников (КРЗ)	10	10
Подготовка к промежуточной аттестации (ПА)	19	19
Всего:	119	119

5. Перечень учебно-методического обеспечения

для самостоятельной работы обучающихся по дисциплине (модулю)

Учебно-методические материалы для самостоятельной работы обучающихся указаны в п.п. разделов 6-11.

6. Перечень печатных и электронных учебных изданий

Перечень печатных и электронных учебных изданий приведен в таблице 8.

Таблица 8– Перечень печатных и электронных учебных изданий

Шифр/ URL адрес	Библиографическая ссылка	Количество экземпляров в библиотеке (кроме электронных экземпляров)
https://www.piter.com/collection/top-2/product/kotlin-multiplatform-na-praktike	Жаркова, А. А. Kotlin Multiplatform на практике. — Санкт-Петербург: Питер, 2026. — 384 с. — ISBN 978-5-4461-4406-8.	-
https://www.piter.com/collection/A36832/product/kotlin-v-deystvii-2-e-izd	Айгнер, С., Елизаров, Р., Исакова, С., Жемеров, Д. Kotlin в действии, 2-е изд. — Санкт-Петербург: Питер, 2025. — 560 с. — ISBN 978-5-4461-4198-2	-
https://www.piter.com/collection/bestsellery-packt/product/kotlin-patterny-proektirovaniya-i-luchshie	Сошин, А. Kotlin. Паттерны	-

praktiki-3-e-izd	проектирования и лучшие практики, 3-е изд. — Санкт-Петербург: Питер, 2025. — 416 с. — ISBN 978-601-09-9694-6	
https://www.piter.com/collection/yazyki-programmirovaniya/product/kotlin-programmirovaniye-dlya-professionalov-2-e-izd	Скин, Д., Гринхол, Д., Бэйли, Э. Kotlin. Программирование для профессионалов. 2-е изд. — Санкт-Петербург: Питер, 2026. — 560 с. — ISBN 978-5-4461-2319-3	-

7. Перечень электронных образовательных ресурсов информационно-телекоммуникационной сети «Интернет»

Перечень электронных образовательных ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины приведен в таблице 9.

Таблица 9 – Перечень электронных образовательных ресурсов информационно-телекоммуникационной сети «Интернет»

URL адрес	Наименование
https://urait.ru/	Образовательная платформа Юрайт
http://lib.guap.ru/	Библиотека ГУАП
https://znanium.com/	Электронно-библиотечная система Znanium
https://book.ru	Электронно-библиотечная система book.ru

8. Перечень информационных технологий

8.1. Перечень программного обеспечения, используемого при осуществлении образовательного процесса по дисциплине.

Перечень используемого программного обеспечения представлен в таблице 10.

Таблица 10– Перечень программного обеспечения

№ п/п	Наименование
1	Android Studio с плагином Kotlin Multiplatform
2	JDK 17+
3	Git

8.2. Перечень информационно-справочных систем,используемых при осуществлении образовательного процесса по дисциплине

Перечень используемых информационно-справочных систем представлен в таблице 11.

Таблица 11– Перечень информационно-справочных систем

№ п/п	Наименование
-------	--------------

	Не предусмотрено
--	------------------

9. Материально-техническая база

Состав материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине, представлен в таблице 12.

Таблица 12 – Состав материально-технической базы

№ п/п	Наименование составной части материально-технической базы	Номер аудитории (при необходимости)
1	Мультимедийная лекционная аудитория	
2	Специализированная лаборатория	52-19, 52-17

10. Оценочные средства для проведения промежуточной аттестации

10.1. Состав оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине приведен в таблице 13.

Таблица 13 – Состав оценочных средств для проведения промежуточной аттестации

Вид промежуточной аттестации	Перечень оценочных средств
Экзамен	Список вопросов к экзамену;

10.2. В качестве критериев оценки уровня сформированности (освоения) компетенций обучающимися применяется 5-балльная шкала оценки сформированности компетенций, которая приведена в таблице 14. В течение семестра может использоваться 100-балльная шкала модульно-рейтинговой системы Университета, правила использования которой, установлены соответствующим локальным нормативным актом ГУАП.

Таблица 14 – Критерии оценки уровня сформированности компетенций

Оценка компетенции 5-балльная шкала	Характеристика сформированных компетенций
«отлично» «зачтено»	Обучающийся: – глубоко и всесторонне усвоил программный материал; – уверенно, логично, последовательно и грамотно его излагает; – опираясь на знания основной и дополнительной литературы, тесно связывает усвоенные научные положения с практической деятельностью направления; – умело обосновывает и аргументирует выдвигаемые им идеи; – делает выводы и обобщения; – свободно владеет системой специализированных понятий. – правильно выполнил от 90% до 100% тестовых заданий**.
«хорошо» «зачтено»	Обучающийся: – твердо усвоил программный материал, грамотно и по существу излагает его, опираясь на знания основной литературы; – не допускает существенных неточностей; – увязывает усвоенные знания с практической деятельностью направления; – аргументирует научные положения; – делает выводы и обобщения; – владеет системой специализированных понятий. – правильно выполнил от 70% до 89% тестовых заданий**.

Оценка компетенции	Характеристика сформированных компетенций
5-балльная шкала	
«удовлетворительно» «зачтено»	<ul style="list-style-type: none"> – обучающийся усвоил только основной программный материал, по существу излагает его, опираясь на знания только основной литературы; – допускает несущественные ошибки и неточности; – испытывает затруднения в практическом применении знаний направления; – слабо аргументирует научные положения; – затрудняется в формулировании выводов и обобщений; – частично владеет системой специализированных понятий. – правильно выполнил от 51% до 69% тестовых заданий**.
«неудовлетворительно» «не зачтено»	<ul style="list-style-type: none"> – обучающийся не усвоил значительной части программного материала; – допускает существенные ошибки и неточности при рассмотрении проблем в конкретном направлении; – испытывает трудности в практическом применении знаний; – не может аргументировать научные положения; – не формулирует выводов и обобщений. – правильно выполнил менее 51% тестовых заданий**.

10.3. Типовые контрольные задания или иные материалы.

Вопросы (задачи) для экзамена представлены в таблице 15.

Таблица 15 – Вопросы (задачи) для экзамена

№ п/п	Перечень вопросов (задач) для экзамена	Код индикатора
1.	Охарактеризуйте язык Kotlin как основу современной кроссплатформенной разработки. Раскройте значение null-safety, data/sealed-классов, функций высшего порядка и coroutines при создании информационных систем.	ПК-1.3.1
2.	Опишите структуру Kotlin Multiplatform-проекта: Gradle Kotlin DSL, source sets, commonMain, platformMain, механизм expect/actual и принципы управления зависимостями.	ПК-1.3.1
3.	Охарактеризуйте современные инструменты и библиотеки Kotlin Multiplatform, применяемые при разработке кроссплатформенных информационных систем: Compose Multiplatform, Ktor, kotlinx.serialization, SQLDelight, средства отладки и тестирования.	ПК-1.3.1
4.	Проанализируйте постановку задачи на создание кроссплатформенного приложения и определите, какие части решения целесообразно вынести в общий код, а какие оставить платформенно-специфичными. Обоснуйте выбор.	ПК-1.У.1
5.	Предложите архитектуру Kotlin Multiplatform/Compose Multiplatform-приложения с сетевым взаимодействием и локальным хранением данных. Обоснуйте выбор модулей, слоев и инструментальных средств.	ПК-1.У.1
6.	Опишите порядок применения IDE, средств статического анализа, профилирования и ИИ-ассистентов при разработке, модификации и сопровождении кроссплатформенной информационной системы.	ПК-1.У.1
7.	Покажите, как ИИ-ассистент может использоваться при	ПК-1.В.1

	генерации или рефакторинге shared-модуля на Kotlin. Укажите ограничения, риски и способы верификации полученного результата.	
8.	Опишите практику применения инструментальных средств прикладной информатики для повышения качества проекта: code review, линтеры, форматирование кода, CI, автоматическая генерация документации и тестов.	ПК-1.В.1
9.	Поясните, как использовать ИИ-ассистентов и средства IDE для миграции общего кода, подготовки шаблонов компонентов и ускорения сопровождения кроссплатформенного проекта.	ПК-1.В.1
10.	Раскройте принципы и архитектурные особенности Compose Multiplatform. Сравните Compose Multiplatform и Jetpack Compose, укажите поддерживаемые платформы и ограничения.	ПК-7.3.1
11.	Охарактеризуйте способы управления состоянием и навигацией в Compose Multiplatform: remember, rememberSaveable, StateFlow, ViewModel-подход, маршрутизация и повторное использование компонентов.	ПК-7.3.1
12.	Опишите подходы к организации ресурсов, тем, локализации и интеграции платформенных API в кроссплатформенном приложении на Kotlin и Compose Multiplatform.	ПК-7.3.1
13.	Спроектируйте экран приложения на Compose Multiplatform, содержащий список объектов, форму фильтрации и экран карточки элемента. Укажите состав компонентов, способ хранения состояния и организацию навигации.	ПК-7.У.1
14.	Предложите стратегию тестирования и отладки Compose Multiplatform-приложения: выделите unit-, UI- и интеграционные тесты, способы поиска ошибок и критерии готовности к выпуску.	ПК-7.У.1
15.	Спроектируйте решение для обмена данными и локального хранения в КМР-проекте: укажите способы работы с HTTP, сериализацией, кэшированием и синхронизацией данных.	ПК-7.У.1
16.	Поясните или реализуйте фрагмент кода общего state-holder/ViewModel-компонента на Kotlin с использованием StateFlow и Compose-состояния для отображения списка данных.	ПК-7.В.1
17.	Опишите процесс создания набора переиспользуемых UI-компонентов и темы оформления для Compose Multiplatform-приложения, пригодного для Android, iOS и Desktop.	ПК-7.В.1
18.	Опишите порядок подготовки кроссплатформенного приложения к сборке и выпуску: настройка целевых платформ, конфигураций сборки, CI/CD и контроль качества.	ПК-7.В.1

Вопросы (задачи) для зачета / дифф. зачета представлены в таблице 16.
Таблица 16 – Вопросы (задачи) для зачета / дифф. зачета

№ п/п	Перечень вопросов (задач) для зачета / дифф. зачета	Код индикатора
	Учебным планом не предусмотрено	

Перечень тем для выполнения курсового проекта/ курсовой работы представлены в таблице 17.

Таблица 17 – Перечень тем для выполнения курсового проекта / курсовой работы

№ п/п	Примерный перечень тем для выполнения курсового проекта/ курсовой работы
	Учебным планом не предусмотрено

Вопросы для проведения промежуточной аттестации в виде тестирования представлены в таблице 18.

Таблица 18 – Примерный перечень вопросов для тестов

№ п/п	Примерный перечень вопросов для тестов	Код индикатора
1.	Прочитайте текст и выберите один правильный ответ. Какой механизм Kotlin Multiplatform используется для объявления общего контракта и платформенно-специфичной реализации? <ul style="list-style-type: none"> • expect/actual • interface delegation • reflection bridge • synthetic override 	ПК-1.3.1
2.	Прочитайте текст и выберите два правильных ответа. Какие инструменты часто применяются в КМР-проекте для сетевого взаимодействия и сериализации данных? <ul style="list-style-type: none"> • Ktor • kotlinx.serialization • Storyboard • XML Pull Parser как основной межплатформенный инструмент 	ПК-1.3.1
3.	Прочитайте текст и выберите один правильный ответ. Для проекта требуется переиспользовать бизнес-логику на Android, iOS и Desktop, но часть платформенных API должна остаться нативной. Какой вариант архитектуры наиболее рационален? <ul style="list-style-type: none"> • Полностью отдельные приложения без общего кода • Общий shared-модуль и платформенные адаптеры для нативных возможностей • Только web-версия приложения • Исключительно общий UI без разделения ответственности 	ПК-1.У.1
4.	Прочитайте текст и выберите два правильных ответа. Какие факторы следует учитывать при выборе библиотек и инструментов для Kotlin Multiplatform-проекта? <ul style="list-style-type: none"> • Поддержку целевых платформ и совместимость с Kotlin-версией • Сопровождаемость решения и активность развития библиотеки • Только личные предпочтения разработчика • Только размер иконки библиотеки на сайте 	ПК-1.У.1

5.	<p>Прочитайте текст и выберите один правильный ответ.</p> <p>Какой подход к использованию ИИ-ассистента при разработке Kotlin-проекта является наиболее корректным?</p> <ul style="list-style-type: none"> • Полностью доверять сгенерированному коду без проверки • Использовать ИИ для черновой генерации, а затем проводить ручную проверку, тестирование и code review • Применять ИИ только для выбора цветовой темы приложения • Исключить запуск тестов после работы ИИ 	ПК-1.В.1
6.	<p>Прочитайте текст и выберите два правильных ответа.</p> <p>Какие задачи может выполнять ИИ-ассистент в процессе разработки Compose Multiplatform-приложения?</p> <ul style="list-style-type: none"> • Подготовка шаблонного кода и рефакторинг типовых фрагментов • Генерация тестовых сценариев и пояснений к коду • Публикация релиза без участия разработчика • Замена итоговой проверки качества продукта 	ПК-1.В.1
7.	<p>Прочитайте текст и выберите один правильный ответ.</p> <p>Что является базовой единицей построения пользовательского интерфейса в Compose Multiplatform?</p> <ul style="list-style-type: none"> • Activity • Fragment • @Composable-функция • Servlet 	ПК-7.3.1
8.	<p>Прочитайте текст и выберите два правильных ответа.</p> <p>Какие подходы и средства являются типичными для управления состоянием в Compose Multiplatform?</p> <ul style="list-style-type: none"> • remember / rememberSaveable • StateFlow / MutableStateFlow • BroadcastReceiver как основной UI-механизм • XML-макеты как обязательный способ описания состояния 	ПК-7.3.1
9.	<p>Прочитайте текст и выберите один правильный ответ.</p> <p>После реализации composable-экрана необходимо проверить, что он корректно отображает данные и реагирует на нажатие кнопки. Какой вид тестирования подходит лучше всего?</p> <ul style="list-style-type: none"> • Нагрузочное тестирование базы данных • UI-тест (Compose UI test) • Пентест • Тестирование серверной инфраструктуры 	ПК-7.У.1
10.	<p>Прочитайте текст и выберите два правильных ответа.</p> <p>Что относится к проектированию и разработке программных компонентов в Compose Multiplatform?</p> <ul style="list-style-type: none"> • Выделение слоев приложения и выбор подхода к управлению состоянием • Реализация экранов, навигации и автоматизированных тестов • Игнорирование обработки ошибок и валидации данных • Случайный выбор библиотек без анализа требований 	ПК-7.У.1
11.	<p>Прочитайте текст и выберите один правильный ответ.</p>	ПК-7.В.1

	<p>В каком source set обычно размещается общий UI-код Compose Multiplatform-приложения?</p> <ul style="list-style-type: none"> • androidMain • iosMain • commonMain • desktopTest 	
12.	<p>Прочитайте текст и выберите два правильных ответа.</p> <p>Какие действия характерны для подготовки кроссплатформенного приложения к выпуску?</p> <ul style="list-style-type: none"> • Настройка целевых платформ, конфигураций сборки и подписи • Подготовка CI/CD-сценариев и проверок качества • Удаление тестов перед релизом • Отказ от анализа зависимостей 	ПК-7.В.1
13.	<p>Прочитайте текст и выберите один правильный ответ.</p> <p>Какой механизм Kotlin чаще всего используется для представления ограниченного набора состояний экрана в общей бизнес-логике?</p> <ul style="list-style-type: none"> • data class • sealed class • package object • typealias 	ПК-1.3.1
14.	<p>Прочитайте текст и выберите два правильных ответа.</p> <p>Какие решения помогают корректно разделить общий и платформенно-специфичный код в Kotlin Multiplatform-проекте?</p> <ul style="list-style-type: none"> • Размещать бизнес-логику и модели данных в commonMain • Использовать expect/actual или адаптеры для платформенных API • Хранить весь код приложения только в androidMain • Дублировать сетевой слой отдельно для каждой платформы без необходимости 	ПК-1.У.1
15.	<p>Прочитайте текст и выберите один правильный ответ.</p> <p>Что следует сделать после того, как ИИ-ассистент предложил рефакторинг shared-модуля?</p> <ul style="list-style-type: none"> • Немедленно объединить изменения с основной веткой без проверки • Запустить тесты, проверить бизнес-логику и провести code review • Удалить комментарии из кода и не анализировать diff • Сразу заменить существующую архитектуру на предложенную ИИ 	ПК-1.В.1
16.	<p>Прочитайте текст и выберите два правильных ответа.</p> <p>Какие практики повышают надежность при использовании ИИ-ассистента в процессе разработки?</p> <ul style="list-style-type: none"> • Формулировать требования к результату и проверять код по чек-листу • Сопровождать изменения автотестами или ручной верификацией 	ПК-1.В.1

	<ul style="list-style-type: none"> • Публиковать релиз без локальной сборки и прогона тестов • Отключать статический анализ перед коммитом 	
17.	<p>Прочитайте текст и выберите один правильный ответ.</p> <p>Какая функция Compose запоминает значение между рекомпозициями внутри composable?</p> <ul style="list-style-type: none"> • remember • launch • suspend • apply 	ПК-7.3.1
18.	<p>Прочитайте текст и выберите два правильных ответа.</p> <p>Какие действия относятся к корректной организации навигации и состояния в Compose Multiplatform?</p> <ul style="list-style-type: none"> • Хранить UI-состояние в state-holder-компоненте, ViewModel-подходе или StateFlow • Передавать между экранами только необходимые аргументы и события • Использовать глобальные mutable-переменные как основной способ хранения состояния • Смешивать навигацию, сетевые запросы и всю бизнес-логику внутри одного composable 	ПК-7.3.1
19.	<p>Прочитайте текст и выберите один правильный ответ.</p> <p>Для хранения структурированных данных каталога, которые должны быть доступны офлайн, наиболее уместно использовать:</p> <ul style="list-style-type: none"> • SharedPreferences как основное хранилище сложных сущностей • SQLDelight или иное SQLite-ориентированное локальное хранилище • Только build.gradle.kts • Preview-аннотации Compose 	ПК-7.У.1
20.	<p>Прочитайте текст и выберите два правильных ответа.</p> <p>Какие действия обязательны перед выпуском Compose Multiplatform-приложения?</p> <ul style="list-style-type: none"> • Проверить сборку целевых платформ, конфигурации и зависимости • Выполнить тестирование ключевых сценариев и базовый контроль качества • Удалить обработку ошибок ради уменьшения размера приложения • Исключить CI/CD-проверки из процесса релиза 	ПК-7.В.1

Ключи правильных ответов на тесты размещены в Приложении 1 к РПД и находятся у специалистов по УМР кафедры 41, заместителя заведующего кафедрой и руководителя образовательной программы.

Система оценивания тестовых заданий показана в таблице 18.1

Таблица 18.1 – Система оценивания тестовых заданий

№	Указания по оцениванию	Результат оценивания (баллы, полученные за выполнение \ характеристика правильности ответа)
---	------------------------	---

1	Задание закрытого типа на установление соответствия считается верным, если установлены все соответствия (позиции из одного столбца верно сопоставлены с позициями другого столбца)	Полное совпадение с верным ответом оценивается 1 баллом, неверный ответ или его отсутствие – 0 баллов (либо указывается «верно» \ «неверно»)
2	Задание закрытого типа на установление последовательности считается верным, если правильно указана вся последовательность цифр	Полное совпадение с верным ответом оценивается 1 баллом, если допущены ошибки или ответ отсутствует – 0 баллов (либо указывается «верно» \ «неверно»)
3	Задание комбинированного типа с выбором одного верного ответа из четырех предложенных и обоснованием выбора считается верным, если правильно указана цифра и приведены конкретные аргументы, используемые при выборе ответа	Полное совпадение с верным ответом оценивается 1 баллом, неверный ответ или его отсутствие – 0 баллов (либо указывается «верно» \ «неверно»)
4	Задание комбинированного типа с выбором нескольких вариантов ответа из предложенных и развернутым обоснованием выбора считается верным, если правильно указаны цифры и приведены конкретные аргументы, используемые при выборе ответов	Полное совпадение с верным ответом оценивается 1 баллом, если допущены ошибки или ответ отсутствует – 0 баллов (либо указывается «верно» \ «неверно»)
5	Задание открытого типа с развернутым ответом считается верным, если ответ совпадает с эталонным по содержанию и полноте	Правильный ответ за задание оценивается в 3 балла, если допущена одна ошибка \ неточность \ ответ правильный, но не полный - 1 балл, если допущено более 1 ошибки \ ответ неправильный \ ответ отсутствует – 0 баллов (либо указывается «верно» \ «неверно»)

Инструкция по выполнению тестового задания находится в таблице 18.2.

Таблица 18.2 - Инструкция по выполнению тестового задания

№	Тип задания	Инструкция
1	Задание закрытого типа на установление соответствия	Прочитайте текст и установите соответствие. К каждой позиции, данной в левом столбце, подберите соответствующую позицию в правом столбце
2	Задание закрытого типа на установление последовательности	Прочитайте текст и установите последовательность Запишите соответствующую последовательность букв слева направо

3	Задание комбинированного типа с выбором одного верного ответа из четырех предложенных и обоснованием выбора	Прочитайте текст, выберите правильный ответ и запишите аргументы, обосновывающие выбор ответа
4	Задание комбинированного типа с выбором нескольких вариантов ответа из предложенных и развернутым обоснованием выбора	Прочитайте текст, выберите правильные варианты ответа и запишите аргументы, обосновывающие выбор ответов
5	Задание открытого типа с развернутым ответом	Прочитайте текст и запишите развернутый обоснованный ответ

Перечень тем контрольных работ по дисциплине обучающихся заочной формы обучения, представлены в таблице 19.

Таблица 19 – Перечень контрольных работ

№ п/п	Перечень контрольных работ
1	Разработка и обоснование архитектуры кроссплатформенного приложения на Kotlin Multiplatform с пользовательским интерфейсом на Compose Multiplatform

10.4. Методические материалы, определяющие процедуры оценивания индикаторов, характеризующих этапы формирования компетенций, содержатся в локальных нормативных актах ГУАП, регламентирующих порядок и процедуру проведения текущего контроля успеваемости и промежуточной аттестации обучающихся ГУАП.

11. Методические указания для обучающихся по освоению дисциплины

11.1. Методические указания для обучающихся по освоению лекционного материала

Основное назначение лекционного материала – логически стройное, системное, глубокое и ясное изложение учебного материала. Назначение современной лекции в рамках дисциплины не в том, чтобы получить всю информацию по теме, а в освоении фундаментальных проблем дисциплины, методов научного познания, новейших достижений научной мысли. В учебном процессе лекция выполняет методологическую, организационную и информационную функции. Лекция раскрывает понятийный аппарат конкретной области знания, её проблемы, дает цельное представление о дисциплине, показывает взаимосвязь с другими дисциплинами.

Планируемые результаты при освоении обучающимися лекционного материала:

- получение современных, целостных, взаимосвязанных знаний, уровень которых определяется целевой установкой к каждой конкретной теме;
- получение опыта творческой работы совместно с преподавателем;
- развитие профессионально-деловых качеств, любви к предмету и самостоятельного творческого мышления.
- появление необходимого интереса, необходимого для самостоятельной работы;
- получение знаний о современном уровне развития науки и техники и о прогнозе их развития на ближайшие годы;
- научиться методически обрабатывать материал (выделять главные мысли и положения, приходить к конкретным выводам, повторять их в различных формулировках);

- получение точного понимания всех необходимых терминов и понятий.

Лекционный материал может сопровождаться демонстрацией слайдов и использованием раздаточного материала при проведении коротких дискуссий об особенностях применения отдельных тематик по дисциплине.

Структура предоставления лекционного материала:

- изложение теоретических основ языка Kotlin, архитектуры Kotlin Multiplatform и Compose Multiplatform;
- разбор типовых архитектурных и технологических решений, используемых при создании кроссплатформенных приложений;
- обсуждение примеров проектирования, тестирования, отладки, сборки и публикации приложений.

11.2. Методические указания для обучающихся по участию в семинарах
Учебным планом не предусмотрено.

11.3. Методические указания для обучающихся по прохождению практических занятий

Практическое занятие является одной из основных форм организации учебного процесса, заключающаяся в выполнении обучающимися под руководством преподавателя комплекса учебных заданий с целью усвоения научно-теоретических основ учебной дисциплины, приобретения умений и навыков, опыта творческой деятельности.

Целью практического занятия для обучающегося является привитие обучающимся умений и навыков практической деятельности по изучаемой дисциплине.

Планируемые результаты при освоении обучающимися практических занятий:

- закрепление, углубление, расширение и детализация знаний при решении конкретных задач;
- развитие познавательных способностей, самостоятельности мышления, творческой активности;
- овладение новыми методами и методиками изучения конкретной учебной дисциплины;
- выработка способности логического осмысления полученных знаний для выполнения заданий;
- обеспечение рационального сочетания коллективной и индивидуальной форм обучения.

Требования к проведению практических занятий

Отчет о выполнении практической работы должен включать: титульный лист, цель и формулировку задания, описание использованных технологий и инструментов, листинг или ссылку на исходный код, полученные результаты в виде скриншотов и краткие выводы. Файлы с отчетом и исходным кодом загружаются в LMS. Защита практической работы предполагает демонстрацию работающего решения, пояснение архитектурных решений, ответ на вопросы преподавателя и анализ допущенных ошибок.

11.4. Методические указания для обучающихся по выполнению лабораторных работ

Учебным планом не предусмотрено.

11.5. Методические указания для обучающихся по выполнению курсового проекта/курсовой работы

Учебным планом не предусмотрено.

11.6. Методические указания для обучающихся по прохождению самостоятельной работы

В ходе выполнения самостоятельной работы, обучающийся выполняет работу по заданию и при методическом руководстве преподавателя, но без его непосредственного участия.

В процессе выполнения самостоятельной работы у обучающегося формируется целесообразное планирование рабочего времени, которое позволяет ему развивать умения и навыки в усвоении и систематизации приобретаемых знаний, обеспечивает высокий уровень успеваемости в период обучения, помогает получить навыки повышения профессионального уровня.

Методическими материалами, направляющими самостоятельную работу обучающихся являются:

- учебно-методический материал по дисциплине;

11.7. Методические указания для обучающихся по прохождению текущего контроля успеваемости.

Текущий контроль успеваемости предусматривает контроль качества знаний обучающихся, осуществляемого в течение семестра с целью оценивания хода освоения дисциплины.

Текущий контроль успеваемости рекомендуется осуществлять в форме тестирования, устного опроса по темам лекций и оценки результатов самостоятельной работы по анализу архитектурных решений backend-сервисов. Результаты текущего контроля учитываются при проведении промежуточной аттестации и характеризуют степень освоения заявленных компетенций.

11.8. Методические указания для обучающихся по прохождению промежуточной аттестации.

Промежуточная аттестация обучающихся предусматривает оценивание промежуточных и окончательных результатов обучения по дисциплине. Она включает в себя:

- экзамен – форма оценки знаний, полученных обучающимся в процессе изучения всей дисциплины или ее части, навыков самостоятельной работы, способности применять их для решения практических задач. Экзамен, как правило, проводится в период экзаменационной сессии и завершается аттестационной оценкой «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

Для успешного прохождения промежуточной аттестации в форме экзамена обучающийся должен продемонстрировать соответствие критериям оценки уровня сформированности компетенций (таблица 14), а также выполнить, выложить отчеты в личный кабинет и успешно защитить не менее 60% практических работ. На оценку отлично могут претендовать только те студенты, которые на протяжении семестра выполняли командный проект, сдали все лабораторные работы и получили в сумме минимум 85% от максимально возможного количества баллов и имеют 90% посещений лекций.

Система оценок при проведении промежуточной аттестации формируется в соответствии с требованиями «Положение о текущем контроле успеваемости и промежуточной аттестации студентов и аспирантов ГУАП, обучающихся по образовательным программам высшего образования».

Лист внесения изменений в рабочую программу дисциплины

Дата внесения изменений и дополнений. Подпись внесшего изменения	Содержание изменений и дополнений	Дата и № протокола заседания кафедры	Подпись зав. кафедрой