ЦЕНТРАЛЬНЫЙ УНИВЕРСИТЕТ

УТВЕРЖДЕНА

Решением Ученого совета
АНО ВО «Центральный университет»
«24» июня 2025 г.
Протокол №2

Рабочая программа дисциплины (модуля)
«Основы мобильной разработки»

Направление подготовки: *02.03.01 Математика и компьютерные науки*

Направленность (профиль) подготовки: *Математика и компьютерные науки*

Квалификация (степень) выпускника: *бакалавр*

Форма обучения: *очная*

Срок освоения программы: *4 года*

Год набора: *2025*

ДОКУМЕНТ ПОДПИСАН
ЭЛЕКТРОННОЙ ПОДПИСЬЮ

Сертификат: 0215B0C000D1B3FEB44A37EE631E0C27D4
Владелец: Ивашкевич Евгений Васильевич
Действителен: с 13.01.2026 до 13.04.2027

Электронный документ

Москва
2025

Содержание

1. **Краткая характеристика дисциплины (модуля)** 3
2. **Перечень планируемых результатов обучения** 4
3. **Тематический план** 6
4. **Содержание дисциплины (модуля)** 6
5. **Учебно-методическое обеспечение** 7
6. **Материально-техническое обеспечение** 7
7. **Методические и оценочные материалы** 9

1. Краткая характеристика дисциплины (модуля)

Рабочая программа дисциплины (модуля) «Основы мобильной разработки» составлена в соответствии с федеральным государственным образовательным стандартом высшего образования – бакалавриат по специальности 02.03.01 Математика и компьютерные науки, профиль Математика и компьютерные науки, утвержденный приказом Министерства науки и высшего образования Российской Федерации № 807 от 23.08.2017 года.

Изучение дисциплины (модуля) «Основы мобильной разработки» позволяет овладеть навыками создания современных мобильных приложений, отвечающих требованиям быстроразвивающегося рынка технологий. Это способствует развитию практических умений в программировании, дизайне интерфейсов и оптимизации работы приложений для различных платформ.

Место дисциплины (модуля) в структуре образовательной программы

Настоящая дисциплина (модуль) включена в учебный план по программе подготовки бакалавриата по направлению 02.03.01 Математика и компьютерные науки, профиль Математика и компьютерные науки и входит в вариативную часть Блока 1, формируемую участниками образовательных отношений.

Дисциплина (модуль) является выборной и доступна для изучения на 3 или 4 курсе в 5, 6 или 7 семестре на выбор.

Цель изучения дисциплины (модуля): формирование практических навыков проектирования, разработки и внедрения эффективных мобильных приложений для различных операционных систем и устройств.

Задачи изучения дисциплины (модуля):

— освоить базовые концепции архитектуры мобильных приложений, включая жизненный цикл, компоненты и паттерны проектирования;

— научиться разрабатывать пользовательские интерфейсы с помощью языков программирования и фреймворков, таких как Kotlin, Swift;

— получить навыки интеграции приложений с внешними сервисами, обеспечения безопасности и проведения тестирования для достижения качества и производительности.

В результате освоения дисциплины (модуля) обучающийся должен:

знать:

— на базовом уровне языки Swift и Kotlin;

— основные принципы и понятия мобильной разработки;

— базовые фреймворки для нативной разработки под iOS и Android;

— основные подходы в мобильной разработке и понимает их сильные и слабые стороны;

— существующие подходы к кросс- и мультиплатформенной мобильной разработке и понимает критерии их использования;

уметь:

— разрабатывать базовые приложения как для iOS, так и для Android с использованием нативных технологий;

— создавать пользовательские интерфейсы с использованием разных подходов;

— обосновывать выбор технологий для разработки;

владеть:

— разработать базовое приложение как для iOS, так и для Android с использованием нативных технологий;

— - выбрать подходящие технолоии для разработки.

2. Перечень планируемых результатов обучения

Компетенции, формируемые в результате освоения дисциплины (модуля) при проведении учебных занятий в форме контактной работы обучающихся с педагогическими работниками Университета и в форме самостоятельной работы обучающихся:

Компетенция	Содержание компетенции	Индикатор компетенции	Перечень планируемых результатов обучения по дисциплине (модулю)
УК-1.	Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1.1.	Знает методы поиска и анализа информации в области разработки, основные принципы критической оценки источников информации и их релевантности
		УК-1.2.	Умеет критически оценивать источники информации и синтезировать данные из различных источников для решения задач, применять системный подход к анализу и решению комплексных проблем
		УК-1.3.	Имеет практический опыт работы с современными инструментами и технологиями для обработки информации, формулировании и структурировании задач на основе полученной информации
ОПК-1.	Способен консультировать и использовать фундаментальные знания в области математического анализа, комплексного и функционального анализа алгебры, аналитической геометрии, дифференциальной геометрии и топологии, дифференциальных уравнений, дискретной математики и математической логики, теории вероятностей, математической статистики и случайных процессов, численных методов, теоретической механики в профессиональной деятельности	ОПК-1.1.	Знает основные концепции и теории в области математического анализа и смежных дисциплин; методы и подходы, используемые в различных областях математики
		ОПК-1.2.	Умеет применять математические методы для решения профессиональных задач
		ОПК-1.3.	Имеет практический опыт разработки и реализация математических моделей в профессиональной деятельности
ОПК-4.	Способен находить, анализировать, реализовывать программно и использовать на практике математические алгоритмы, в том числе с применением современных вычислительных систем	ОПК-4.1.	Знает базовые основы современного математического аппарата, связанного с проектированием, разработкой, реализацией и оценкой качества программных продуктов и программных комплексов в различных областях человеческой деятельности

		ОПК-4.2.	Умеет использовать этот математический аппарат в профессиональной деятельности
		ОПК-4.3.	Имеет практический опыт применения современного математического аппарата, связанного с проектированием, разработкой, реализацией и оценкой качества программных продуктов и программных комплексов в различных областях человеческой деятельности
ОПК-6.	Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения	ОПК-6.1.	Знает алгоритмы разработки, компьютерные программы, а также алгоритмы вычислительной математики в области профессиональной деятельности
		ОПК-6.2.	Умеет разрабатывать математические программные продукты и комплексы с использованием современных технологий программирования в области профессиональной деятельности
		ОПК-6.3.	Имеет практический опыт разработки интеллектуальных информационных систем для визуализации результатов исследований в области профессиональной деятельности
ПК-3.	Способен применять методы математического и алгоритмического моделирования для решения как теоретических, так и практических задач в рамках профессиональной деятельности	ПК-3.1.	Знает основные методы математического и алгоритмического моделирования, а также их применение для решения теоретических и прикладных задач
		ПК-3.2.	Умеет применять методы математического и алгоритмического моделирования для анализа и решения различных задач в области математики и компьютерных наук
		ПК-3.3.	Имеет опыт использования методов математического и алгоритмического моделирования при решении теоретических и прикладных задач в профессиональной деятельности

3. Тематический план

№п/п	Наименование раздела дисциплины (модуля)	Трудоемкость, академические часы				ТКУ (текущий контроль успеваемости)
		Очная форма				
		Контактная работа		Контроль	Самостоятельная работа	
		Лекции	Семинары (практические занятия)			
1	Платформы мобильной разработки и технологии	6	6		25	Домашние задания Аудиторная работа
2	Пользовательский интерфейс	6	6		25	Домашние задания Аудиторная работа
3	Инструменты работы с данными и процессами	6	6		26	Домашние задания Аудиторная работа
4	Архитектура и качество	6	6		26	Домашние задания Аудиторная работа
5	Продвинутые инструменты мобильной разработки	6	6		25	Аудиторная работа Проект
	Зачет с оценкой			3		
	Итого:	***30***	***30***	***3***	***127***	
Объем дисциплины (модуля) (в ак. ч.)		***190***				
Объем дисциплины (модуля) (в зач. ед.)		***5***				

4. Содержание дисциплины (модуля)

№п/п	Наименование раздела дисциплины (модуля)	Содержание дисциплины (модуля) по темам
1	Платформы мобильной разработки и технологии	Введение в основные понятия мобильной разработки. Основа базовых языков Swift и Kotlin. Создание проектов iOS/Android приложений. Менеджеры зависимостей. Инструменты сборки проектов
2	Пользовательский интерфейс	Разработка пользовательского интерфейса и работа с системными компонентами. Декларативный и императивный подход в пользовательском интерфейсе. Навигация
3	Инструменты работы с данными и процессами	Базовая многопоточность. Сравнение iOS и Android фреймворков для работы с событиями и параллельной логикой. Хранение данных на устройстве. Сетевое взаимодействие
4	Архитектура и качество	Введение в понятия качества разработки, архитектур и тестирования. Модульность и внедрение зависимостей
5	Продвинутые инструменты мобильной разработки	Фреймворки и платформенные инструменты. Фреймворки и платформенные инструменты. Кросс/мульти-платформенные инструменты

5. Учебно-методическое обеспечение

Университет располагает полным набором лицензионного и свободно распространяемого программного обеспечения, включая продукты отечественного производства.

Каждый студент в течение всего периода обучения получает индивидуальный неограниченный доступ к электронно-библиотечной системе и электронной информационно-образовательной среде университета. Эти системы предоставляют возможность доступа к ресурсам из любой точки, где есть подключение к сети Интернет, как на территории университета, так и за его пределами.

Студентам обеспечен удаленный доступ к современным профессиональным базам данных и информационным справочным системам.

Основная литература:

1. Усов, В. Swift. Основы разработки приложений под iOS, iPadOS и macOS : практическое руководство / В. Усов. - 6-е изд. доп. и перераб. - Санкт-Петербург : Питер, 2021. - 545 с. - Серия «Библиотека программиста»). - ISBN 978-5-4461-1796-3. - Текст : электронный. - URL: https://znanium.ru/catalog/product/2139472.

2. Гриффитс, Д. Head First. Kotlin : практическое руководство / Д. Гриффитс, Д. Гриффитс. - Санкт-Петербург : Питер, 2020. - 464 с. - (Серия «Head First O'Reilly»). - ISBN 978-5-4461-1335-4. - Текст : электронный. - URL: https://znanium.com/catalog/product/1733498.

3. Бурнет, Э. Привет, Android! Разработка мобильных приложений : практическое руководство / Э. Бурнет, А. Заика, А. Севостьянова. - Санкт-Петербург : Питер, 2012. - 256 с. - ISBN 978-5-459-01015-2. - Текст : электронный. - URL: https://znanium.com/catalog/product/1770490.

4. Аделекан, И. Kotlin: программирование на примерах : практическое руководство / И. Аделекан. - Санкт-Петербург : БХВ-Петербург, 2020. - 432 с. - ISBN 978-5-9775-6673-5. - Текст : электронный. - URL: https://znanium.ru/catalog/product/2139144.

Дополнительная литература:

1. Зараменских Е. П. Разработка информационных систем : учебник и практикум для вузов / Е. П. Зараменских. — 2-е изд. — Москва : Издательство Юрайт, 2025. — 78 с. — (Высшее образование). — ISBN 978-5-534-21420-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/571333.

2. Соколова В. В. Вычислительная техника и информационные технологии. Разработка мобильных приложений : учебник для вузов / В. В. Соколова. — Москва : Издательство Юрайт, 2025. — 160 с. — (Высшее образование). — ISBN 978-5-534-16302-5. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/561336.

6. Материально-техническое обеспечение

Университет располагает материально-технической базой, соответствующей действующим противопожарным правилам и нормам и обеспечивающей проведение всех видов дисциплинарной и междисциплинарной подготовки, практической и научно-исследовательской работ обучающихся, предусмотренных учебным планом.

Помещения, которые представляют собой учебные аудитории для проведения занятий лекционного типа, занятий семинарского (практического) типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, а также помещения для самостоятельной работы и помещения для хранения и профилактического обслуживания учебного оборудования. Помещения укомплектованы специализированной

мебелью и техническими средствами обучения, служащими для представления учебной информации большой аудитории.

Изучение дисциплины (модуля) обеспечивается в учебных аудиториях, оснащенных:

— столами и стульями;
— компьютерной техникой;
— механическими калькуляторами;
— специализированным оборудованием, включая демонстрационное оборудование.

Помещения для самостоятельной работы обучающихся, в том числе приспособленные для использования инвалидами и лицами с ограниченными возможностями здоровья, оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду Университета.

Обучающимся предоставляется доступ (в том числе удаленный) к ресурсам информационно-телекоммуникационной сети «Интернет», электронным ресурсам (в том числе электронным библиотечным системам, современным профессиональным базам данных и информационным справочным системам):

№	Наименование портала (издания, курса, документа)	Ссылка
1.	Научная электронная библиотека elibrary.ru библиотека	https://elibrary.ru/defaultx.asp
2.	База данных для IT-специалистов	https://habr.com
3.	База данных ScienceDirect	https://www.sciencedirect.com
4.	Официальный сайт Министерства науки и высшего образования Российской Федерации	https://minobrnauki.gov.ru/
5.	Федеральный портал «Российское образование"	https://www.edu.ru/
6.	Информационная система "Единое окно доступа к образовательным ресурсам"	http://window.edu.ru/
7.	Единая коллекция цифровых образовательных ресурсов	http://school-collection.edu.ru/
8.	Федеральный центр информационно - образовательных ресурсов	http://fcior.edu.ru/

Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), в том числе комплект лицензионного программного обеспечения, современные профессиональные базы данных и информационные справочные системы:

Наименование ПО	Производство	Лицензионное / свободно распространяемое
Операционные системы:		
Microsoft Imagine (Windows Client, Server)	зарубежное	лицензионное
Браузеры:		
Яндекс.Браузер	отечественное	свободно распространяемое
Google Chrome	зарубежное	свободно распространяемое
Офисные приложения:		
Microsoft Imagine (Visio, OneNote)	зарубежное	лицензионное
TeXstudio	зарубежное	свободно распространяемое
Adobe Acrobat Reader	зарубежное	свободно распространяемое
Программное обеспечение для планирования и учета времени:		
Toggle app	зарубежное	свободно распространяемое
Системы управления проектами:		
Microsoft Imagine (Project)	зарубежное	лицензионное

Системы управления базами данных:		
Microsoft Imagine (SQL Server)	зарубежное	лицензионное
Системы резервного копирования (backup):		
Acronis Backup Advanced for HyperV	зарубежное	лицензионное
Справочно-правовые системы:		
КонсультантПлюс: справочно-правовая система	отечественное	лицензионное
Средства антивирусной защиты:		
Kaspersky Endpoint Security для бизнеса Стандартный Russian Edition	отечественное	лицензионное
Среды разработки:		
Visual Studio Code	зарубежное	свободно распространяемое
Bash (Unix shell)	зарубежное	свободно распространяемое
Anaconda	зарубежное	свободно распространяемое
Robotic Operating System	зарубежное	свободно распространяемое
CopelliaSim	зарубежное	свободно распространяемое
Google Colaboratory	зарубежное	свободно распространяемое
Пакеты программных средств и библиотек:		
AutoPsy	зарубежное	свободно распространяемое
Interactive Disassembler (IDA)	зарубежное	свободно распространяемое
Системы управления библиографической информацией:		
Zotero	зарубежное	свободно распространяемое
Сервисы и службы:		
Bind	зарубежное	свободно распространяемое
Docker	зарубежное	свободно распространяемое

7. Методические и оценочные материалы

Методические указания для обучающихся по освоению дисциплины (модуля)

В процессе изучения дисциплины (модуля) «Основы мобильной разработки» в рамках текущего контроля успеваемости используются такие виды учебной работы, как лекции, семинары, аудиторная работа, домашние задания, проект, а также различные виды самостоятельной работы обучающихся по заданию преподавателя, направленные на развитие навыков профессиональной лексики, закрепление практических профессиональных компетенций, поощрение инициатив.

Лекция – систематическое, последовательное, монологическое изложение преподавателем учебного материала, как правило, теоретического характера.

В процессе лекций рекомендуется вести конспект лекций: кратко и схематично фиксировать основные идеи, выводы и обобщения лекции; выделять важные мысли, ключевые слова и термины. Необходимо отметить вопросы или материалы, которые вызывают затруднения, и попытаться найти ответы в рекомендованной литературе. Если разобраться в материале не удается, следует сформулировать вопрос и задать его преподавателю на консультации или во время семинарского (практического) занятия.

Семинар — это форма учебной деятельности, проводимая в учебном заведении под руководством преподавателя, где студенты активно участвуют в обсуждениях, практических заданиях и других формах взаимодействия.

Для успешной подготовки к семинару рекомендуется заранее ознакомиться с темой занятия и основными материалами, чтобы иметь возможность активно участвовать в обсуждении. Также полезно подготовить вопросы и идеи для обсуждения, что поможет глубже понять материал и продемонстрировать заинтересованность.

Домашнее задание – набор задач по темам недели.

При работе над домашними заданиями важно внимательно ознакомиться с требованиями и сроками выполнения. Рекомендуется разбивать задания на этапы, чтобы избежать перегрузки и лучше усвоить материал. Использовать различные источники информации, включая учебники и онлайн-ресурсы, для более глубокого понимания темы.

Аудиторная работа – активная работа студента на семинаре, решение небольших заданий во время семинарских занятий на пройденный лекционный материал.

Для успешного участия в семинаре студентам рекомендуется заранее ознакомиться с темой обсуждения, прочитать необходимые материалы и подготовить вопросы. Важно активно слушать и вовлекаться в дискуссию, высказывая свои мнения и аргументируя их. При ответах на вопросы преподавателя стоит быть уверенным, четким и логичным, опираясь на изученный материал. Также полезно поддерживать диалог с однокурсниками, чтобы обогатить обсуждение и расширить свои знания.

Проект – исследовательская работа по курсу и презентация результатов.

Для успешной подготовки к проекту: четко определите цели и задачи проекта, распределите роли и обязанности между участниками, а также установите сроки выполнения каждой части работы. Регулярно проводите встречи для обсуждения прогресса и решения возникающих вопросов.

Самостоятельная работа – работа студентов, направленная на углубленное изучение отдельных тем и вопросов учебной дисциплины (модуля).

В процессе самостоятельной работы студенты взаимодействуют с рекомендованными материалами при минимальном участии преподавателя. Задачи студента включают работу с конспектами лекций (обработка текста), повторное изучение учебных материалов, планов и тезисов ответов, изучение дополнительных тем, выполнение учебно-исследовательских заданий и другое.

Система оценивания результатов обучения по дисциплине (модулю)

Критерии получения уровня и оценивания сформированности компетенций по дисциплине (модулю) «Основы мобильной разработки»

Оценивание уровня учебных достижений, обучающихся по дисциплине (модулю), осуществляется в виде текущего контроля успеваемости и промежуточной аттестации.

Промежуточная аттестация по дисциплине (модулю) осуществляется в форме ***зачета с оценкой***, при этом проводится оценка компетенций, сформированных по дисциплине.

Для оценивания текущего контроля успеваемости и промежуточной аттестации используется десятибалльная шкала оценивания, которая соотносится с традиционной пятибалльной шкалой следующим образом:

Десятибалльная оценка	**Пятибалльная оценка**	**Оценка за зачет**	**Общая характеристика результата обучения по дисциплине (модулю)**
10	Отлично	Зачтено	Студент полностью владеет знаниями, изложенными в рабочей программе, и глубоко осмысляет дисциплину. Он самостоятельно и логически последовательно отвечает на все вопросы, акцентируя внимание на наиболее важном. Умеет анализировать,
9	Отлично	Зачтено	
8	Отлично	Зачтено	

Десятибалльная оценка	Пятибалльная оценка	Оценка за зачет	Общая характеристика результата обучения по дисциплине (модулю)
			сравнивать, классифицировать, обобщать, конкретизировать и систематизировать изученный материал, выделяя ключевые моменты и устанавливая причинно-следственные связи. Четко формулирует ответы, уверенно интерпретирует результаты анализов и других исследований, а также решает сложные задачи. Студент хорошо знаком с методами исследования, необходимыми для практической деятельности, и умеет связывать теоретические аспекты дисциплины (модуля) с практическими задачами.
7	Хорошо	Зачтено	Студент обладает знаниями предмета почти в полном объеме рабочей программы и самостоятельно, логически последовательно и всесторонне отвечает на все вопросы, акцентируя внимание на наиболее значимых моментах. Он умеет анализировать, сравнивать, классифицировать, обобщать, конкретизировать и систематизировать изученный материал, выделяя его ключевые аспекты и устанавливая причинно-следственные связи. Формулирует свои ответы, уверенно интерпретирует результаты анализов и других исследований, а также решает сложные ситуационные задачи. Студент хорошо знаком с методами исследования, необходимыми для практической деятельности, и умеет связывать теоретические аспекты предмета с практическими задачами.
6	Хорошо	Зачтено	
5	Удовлетворительно	Зачтено	Студент обладает базовыми знаниями по дисциплине (модулю), но испытывает трудности при самостоятельных ответах и использует неточные формулировки. В ходе ответов он допускает ошибки, касающиеся сути вопросов. Студент способен решать только самые простые задачи и владеет лишь минимальным набором методов исследования.
4	Удовлетворительно	Зачтено	
3	Не сдан	Не зачтено	Студент не овладел обязательным минимумом знаний по предмету и не может ответить на вопросы, даже если
2	Не сдан	Не зачтено	
1	Не сдан	Не зачтено	

Десятибалльная оценка	Пятибалльная оценка	Оценка за зачет	Общая характеристика результата обучения по дисциплине (модулю)
			преподаватель задает дополнительные наводящие вопросы.

Дисциплина (модуль) «Основы мобильной разработки» оценивается следующим образом:

Активность	Вес	Количество	Описание
Домашние задания	20%	7	Набор задач по темам недели
Аудиторная работа	20%	5	Решение небольших заданий во время семинарских занятий на пройденный лекционный материал
Проект	30%	1	Оценка за итоговое ревью всего проекта, реализованного в течение курса
Зачет с оценкой	30%	1	Устный ответ на вопросы из билетов

Формула расчёта итоговой оценки по дисциплине (модулю) «Основы мобильной разработки»: «0,2 × среднее за домашние задания + 0,2 × среднее за аудиторную работу + 0,3 ×за проект + 0,3 × зачет с оценкой».

Текущий контроль успеваемости обучающихся по дисциплине (модулю)

Примерные домашние задания

Домашнее задание № 1

1. **Теоретическое задание:** Изучите основные понятия мобильной разработки, такие как SDK, API и эмуляторы. Опишите, как Swift и Kotlin используются для создания приложений, и приведите примеры базовых конструкций языка (например, переменные, циклы и функции) для каждого.
2. **Практическое задание:** Создайте новый проект в Xcode (для iOS) и Android Studio (для Android). Настройте базовую структуру приложения, добавьте простой экран с текстом "Hello, World!" и запустите его на эмуляторе. Опишите процесс в отчете (не менее 300 слов).
3. **Сравнительное задание:** Сравните менеджеры зависимостей CocoaPods (iOS) и Gradle (Android). Объясните, как добавить библиотеку (например, для работы с изображениями) в проект, и приведите примеры кода для интеграции.
4. **Практическое задание:** Установите и настройте инструмент сборки (Xcode Build или Gradle Wrapper). Соберите простой проект с зависимостями, исправьте возможные ошибки и предоставьте скриншоты процесса сборки и результата.
5. **Обсуждающее задание:** Обсудите преимущества и недостатки разработки под iOS и Android. Какие инструменты вы бы выбрали для первого проекта новичка и почему? Подготовьте краткий отчет (200–300 слов) с аргументами.

Домашнее задание № 2

1. **Теоретическое задание:** Изучите основные компоненты UI в iOS (UIKit/SwiftUI) и Android (XML/Kotlin Compose). Опишите декларативный и императивный подходы, приведя примеры кода для создания кнопки в каждом стиле.
2. **Практическое задание:** Разработайте простой UI для приложения "Калькулятор" в iOS (используя SwiftUI) и Android (используя XML-разметку). Добавьте поля ввода, кнопки операций и отображение результата. Запустите на эмуляторе и предоставьте скриншоты.

3. **Сравнительное задание:** Сравните системы навигации: UINavigationController в iOS и Navigation Component в Android. Реализуйте переход между двумя экранами (главный и детальный) в обоих фреймворках и объясните различия в коде.

4. **Практическое задание:** Интегрируйте системный компонент, например, камеру (для iOS — UIImagePickerController, для Android — Camera API). Создайте экран, который открывает камеру, делает фото и отображает его. Опишите обработку разрешений в отчете.

5. **Обсуждающее задание:** Обсудите, как декларативный подход (SwiftUI или Compose) упрощает разработку UI по сравнению с императивным. Приведите примеры сценариев, где один подход предпочтительнее, и подготовьте анализ (300 слов) на основе личного опыта или примеров из приложений.

Домашнее задание № 3

1. **Теоретическое задание:** Изучите понятия качества разработки: модульность, тестируемость и архитектурные паттерны (MVP, MVVM, Clean Architecture). Опишите, как внедрение зависимостей (DI) помогает в поддержке кода, с примерами из iOS (Swinject) и Android (Dagger/Koin).

2. **Практическое задание:** Реализуйте простое приложение "Список задач" с использованием MVVM в iOS (Swift) и Android (Kotlin). Разделите код на слои (View, ViewModel, Model) и добавьте DI для управления зависимостями. Предоставьте код и диаграмму архитектуры.

3. **Сравнительное задание:** Сравните фреймворки для тестирования: XCTest в iOS и JUnit/Espresso в Android. Напишите юнит-тест для функции добавления задачи в список и UI-тест для кнопки добавления. Объясните различия в подходах.

4. **Практическое задание:** Улучшите качество кода в приложении из задания 2: добавьте модульность (разделите на модули/библиотеки), внедрите DI и напишите хотя бы 3 теста. Запустите тесты и предоставьте отчет с покрытием кода (используйте инструменты вроде Xcode Coverage или JaCoCo).

5. **Обсуждающее задание:** Обсудите влияние архитектуры на качество и поддерживаемость приложения. Приведите примеры из реальных проектов (например, Google или Apple), где плохая архитектура привела к проблемам, и предложите стратегии улучшения. Подготовьте эссе (400 слов) с рекомендациями.

Примерные задания для аудиторной работы

Платформы мобильной разработки и технологии

1. **Теоретическое задание:** Опишите основные различия между платформами iOS и Android, включая языки программирования (Swift и Kotlin) и их ключевые особенности. Приведите примеры сценариев, где каждый язык подходит лучше.

2. **Практическое задание:** Создайте новый проект в Xcode для iOS-приложения и в Android Studio для Android-приложения. Настройте базовую структуру проекта и объясните роль менеджеров зависимостей (например, CocoaPods для iOS и Gradle для Android).

3. **Сравнительное задание:** Сравните инструменты сборки проектов: Xcode для iOS и Gradle для Android. Опишите, как каждый инструмент управляет зависимостями и сборкой, и приведите примеры команд для сборки простого приложения.

4. **Практическое задание:** Установите менеджер зависимостей (CocoaPods или Gradle) и добавьте внешнюю библиотеку (например, Alamofire для iOS или Retrofit для Android) в проект. Продемонстрируйте, как интегрировать ее в код.

5. **Обсуждающее задание:** Обсудите преимущества и недостатки мобильной разработки для iOS и Android. Какие инструменты сборки вы бы выбрали для кросс-платформенного проекта и почему?

Пользовательский интерфейс

1. **Теоретическое задание:** Объясните различия между декларативным (например, SwiftUI) и императивным (UIKit) подходами к разработке UI в iOS. Приведите примеры компонентов, которые легче реализовать в каждом подходе.

2. **Практическое задание:** Создайте простой экран приложения для Android с использованием XML-разметки и для iOS с использованием Storyboard. Добавьте базовые компоненты, такие как кнопка и текстовое поле, и свяжите их с кодом.

3. **Сравнительное задание:** Сравните навигацию в iOS (с использованием UINavigationController) и Android (с использованием FragmentManager или Jetpack Navigation). Реализуйте простой переход между двумя экранами в обоих фреймворках.

4. **Практическое задание:** Разработайте UI с системными компонентами (например, камера или геолокация) для iOS и Android. Покажите, как запросить разрешения и обработать данные от этих компонентов.

5. **Обсуждающее задание:** Обсудите, как декларативный подход влияет на производительность и удобство разработки UI. Приведите примеры приложений, где один подход предпочтительнее другого.

Инструменты работы с данными и процессами

1. **Теоретическое задание:** Объясните концепцию многопоточности в мобильной разработке. Опишите основные проблемы, такие как deadlock, и как их избежать в iOS (GCD) и Android (Kotlin Coroutines).

2. **Сравнительное задание:** Сравните фреймворки для работы с событиями и параллельной логикой: Grand Central Dispatch (GCD) в iOS и RxJava/Kotlin Flow в Android. Приведите примеры использования для обработки асинхронных задач.

3. **Практическое задание:** Реализуйте хранение данных на устройстве: используйте Core Data для iOS и Room для Android, чтобы сохранить и загрузить список объектов (например, список задач).

4. **Практическое задание:** Создайте простое сетевое взаимодействие с использованием URLSession в iOS и OkHttp в Android. Реализуйте запрос к API (например, получение данных о погоде) и обработку ответа.

5. **Обсуждающее задание:** Обсудите лучшие практики для сетевого взаимодействия и многопоточности в мобильных приложениях. Как вы бы выбрали подход для приложения с высокой нагрузкой на сеть?

Архитектура и качество

1. **Теоретическое задание:** Опишите основные принципы качества разработки, включая модульность и внедрение зависимостей (DI). Приведите примеры архитектурных паттернов, таких как MVP, MVVM или Clean Architecture.

2. **Практическое задание:** Реализуйте простой DI-контейнер в iOS (с использованием Swinject) и Android (с использованием Dagger или Koin). Покажите, как внедрить зависимость в класс и протестировать его.

3. **Сравнительное задание:** Сравните подходы к тестированию в iOS (XCTest) и Android (JUnit и Espresso). Напишите юнит-тест для функции, проверяющей логику приложения.

4. **Практическое задание:** Разработайте модульную архитектуру для простого приложения (например, TODO-лист). Разделите код на слои (presentation, domain, data) и объясните, как DI помогает в этом.

5. **Обсуждающее задание:** Обсудите, как архитектурные паттерны влияют на качество кода и поддерживаемость. Приведите примеры из реальных приложений, где плохая архитектура привела к проблемам.

Продвинутые инструменты мобильной разработки

1. **Теоретическое задание:** Опишите популярные фреймворки и платформенные инструменты для iOS (например, Combine, SwiftUI) и Android (например, Jetpack Compose, WorkManager). Приведите примеры их использования.

2. **Сравнительное задание:** Сравните кросс-платформенные инструменты, такие как React Native, Flutter и Kotlin Multiplatform. Обсудите преимущества и недостатки каждого для разработки приложений под iOS и Android.

3. **Практическое задание:** Создайте простой проект с использованием Flutter или React Native. Реализуйте базовый UI и логику, демонстрируя кросс-платформенность.

4. **Практическое задание:** Интегрируйте платформенный инструмент (например, Firebase для аналитики или push-уведомлений) в проект iOS и Android. Покажите, как настроить и использовать его.

5. **Обсуждающее задание:** Обсудите будущее кросс-платформенной разработки. Какие инструменты вы бы рекомендовали для стартапа, разрабатывающего приложение под несколько платформ, и почему?

Примерное описание и критерии оценивания к итоговому проекту

Описание проекта:

Необходимо создать мобильное приложение, которое показывает навыки из дисциплины (модуля). В приложении должен быть удобный и адаптивный интерфейс с навигацией и анимацией, правильное управление состояниями приложения, работа с локальным хранением данных и взаимодействие с сервером через REST API с асинхронными запросами. Важно также обработать ввод пользователя, сделать валидацию данных и обеспечить стабильную работу приложения при ошибках сети.

Проект помогает проверить умение создавать качественные мобильные приложения с использованием современных инструментов и технологий.

Критерии оценивания

- Логичное и понятное построение кода, правильное разделение логики и интерфейса.
- Эффективное использование инструментов разработки и отладки.
- Удобный и адаптивный дизайн, работающий на разных устройствах.
- Корректное управление жизненным циклом приложения.
- Использование продвинутых элементов интерфейса (списки, анимации, мультимедиа).
- Работа с локальным хранением данных.
- Проверка и обработка пользовательского ввода.
- Асинхронное сетевое взаимодействие с обработкой ошибок.
- Кэширование данных и оптимизация работы приложения.
- Стабильность и отсутствие критических ошибок.

Задания для промежуточной аттестации по дисциплине (модулю)

№ п/п	Задание	Ответ	Компетенция
1	Укажите основной язык программирования для разработки приложений под iOS.	Swift	УК-1
2	Укажите название инструмента для управления зависимостями в Android-проектах.	Gradle	УК-1
4	Укажите название подхода к UI, где интерфейс описывается декларативно в SwiftUI.	Декларативный	УК-1
5	Укажите основной компонент для навигации между экранами в Android.	Navigation Component	УК-1
6	Укажите количество основных операций в математическом анализе, используемых для расчётов в алгоритмах приложений (например, производная, интеграл).	2	ОПК-1
7	Укажите название метода из дискретной математики, применяемого для оптимизации поиска данных в мобильных приложениях.	Граф	ОПК-1
8	Укажите вероятность события в теории вероятностей, используемой для моделирования многопоточности (например, вероятность блокировки потока).	0.5	ОПК-1
9	Укажите название численного метода, применяемого для решения дифференциальных уравнений в симуляциях мобильных приложений.	Метод Эйлера	ОПК-1
10	Укажите количество базовых шагов в алгоритме сортировки массива данных в приложении (например, пузырьковая сортировка).	3	ОПК-4
11	Укажите название алгоритма, используемого для разработки программ с параллельной логикой в мобильных приложениях.	Grand Central Dispatch	ОПК-4
12	Укажите основной инструмент для тестирования UI в Android.	Espresso	ОПК-4
13	Укажите название паттерна архитектуры, разделяющего код на слои Model-View-ViewModel.	MVVM	ОПК-4
14	Укажите количество основных фреймворков для кросс-платформенной разработки (например, Flutter, React Native).	2	ОПК-6
15	Укажите название метода внедрения зависимостей в iOS.	Swinject	ОПК-6
16	Укажите основной язык для формулировки задач в области компьютерных наук.	Алгоритмический	ОПК-6
17	Укажите название структуры данных, используемой для хранения списка задач в мобильном приложении.	Массив	ОПК-6
18	Укажите количество шагов в стандартном алгоритме решения задачи сетевого взаимодействия (например, HTTP-запрос).	4	ПК-3
19	Укажите название метода анализа результатов в математической статистике для оценки производительности приложения.	Регрессия	ПК-3
20	Укажите основной инструмент для модульного тестирования в iOS.	XCTest	ПК-3