

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

**федеральное государственное автономное
образовательное учреждение высшего образования_
«Национальный исследовательский Нижегородский государственный университет
им. Н.И. Лобачевского»**

Дзержинский филиал ННГУ

УТВЕРЖДЕНО
решением Ученого совета ННГУ
протокол № 10 от 02.12.2024 г.

Рабочая программа дисциплины

Технологии разработки приложений для мобильных устройств

Уровень высшего образования
Бакалавриат

Направление подготовки / специальность
09.03.03 - Прикладная информатика

Направленность образовательной программы
ИТ-сервисы и технологии обработки данных в экономике и финансах

Форма обучения
очно-заочная

г. Дзержинск

2025 год начала подготовки

1. Место дисциплины в структуре ОПОП

Дисциплина Б1.В.ДВ.07.02 Технологии разработки приложений для мобильных устройств относится к части, формируемой участниками образовательных отношений образовательной программы.

2. Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы (компетенциями и индикаторами достижения компетенций)

Формируемые компетенции (код, содержание компетенции)	Планируемые результаты обучения по дисциплине (модулю), в соответствии с индикатором достижения компетенции		Наименование оценочного средства	
	Индикатор достижения компетенции (код, содержание индикатора)	Результаты обучения по дисциплине	Для текущего контроля успеваемости	Для промежуточной аттестации
ПК-11: Способен осуществлять модульное и интеграционное тестирование ИС (ИИС), устранять (по мере возможности) обнаруженные несоответствия	ПК-11.1: Демонстрирует знание методологических основ модульного и интеграционного тестирования ИС (ИИС) ПК-11.2: Демонстрирует умение осуществлять модульное и интеграционное тестирование ИС (ИИС) и устранять (по мере возможности) обнаруженные несоответствия. ПК-11.3: Имеет практический опыт модульного и интеграционного тестирования конкретной ИС (ИИС)	ПК-11.1: Знать основные приемы разработки презентаций Web-сайтов Знать Этапы начального обучения пользователей Web-сайтов ПК-11.2: Уметь Разрабатывать презентации Web-сайтов с учетом преемственности Уметь дозировать информацию при презентации Web-сайтов ПК-11.3: Владеть Навыками разработки презентаций Web-сайтов Владеть Навыками публичного выступления при представлении презентации	Задания	Экзамен: Тест
ПК-6: Способен составлять технико-экономическое обоснование проектных решений и техническое задание на разработку ИС	ПК-6.1: Демонстрирует знание методик технико-экономического обоснования проектных решений, связанных с созданием ИС (ИИС) ПК-6.2: Демонстрирует умение выполнять технико-	ПК-6.1: Знать Структуру документации ИС ПК-6.2: Уметь Описывать проект ИС в соответствии с профилем	Задания	Экзамен: Тест

(ИИС)	экономические расчеты при обосновании проектных решений, составлять техническую документацию на разработку ИС (ИИС) ПК-6.3: Имеет практический опыт технико-экономического обоснования конкретного проектного решения и представления технической документации на разработку ИС (ИИС).	подготовки по видам обеспечения ПК-6.3: Владеть Прикладным программным обеспечением для проектирования экономических информационных систем по видам обеспечения		
-------	---	---	--	--

3. Структура и содержание дисциплины

3.1 Трудоемкость дисциплины

	очно-заочная
Общая трудоемкость, з.е.	4
Часов по учебному плану	144
в том числе	
аудиторные занятия (контактная работа):	
- занятия лекционного типа	10
- занятия семинарского типа (практические занятия / лабораторные работы)	14
- КСР	1
самостоятельная работа	83
Промежуточная аттестация	36 Экзамен

3.2. Содержание дисциплины

(структурированное по темам (разделам) с указанием отведенного на них количества академических часов и виды учебных занятий)

Наименование разделов и тем дисциплины	Всего (часы)	в том числе			
		Контактная работа (работа во взаимодействии с преподавателем), часы из них			Самостоятельная работа обучающегося, часы
		Занятия лекционного типа	Занятия семинарского типа (практические занятия/ лабора торные работы), часы	Всего	
	0 3 Ф 0	0 3 Ф 0	0 3 Ф 0	0 3 Ф 0	0 3 Ф 0
Введение в разработку мобильных приложений	12	1	1	2	10
Виды приложений и их структура	12	1	1	2	10
Основы разработки интерфейсов мобильных приложений	15	1	4	5	10
Основы разработки многооконных приложений	21	1	2	3	18

Использование возможностей смартфона в приложениях	14	2	2	4	10
Использование библиотек	14	2	2	4	10
Работа с базами данных, графикой и анимацией. Разработка игр	19	2	2	4	15
Аттестация	36				
КСР	1			1	
Итого	144	10	14	25	83

Содержание разделов и тем дисциплины

Тема 1. Введение в разработку мобильных приложений

Введение, история. Устройство платформ Android, iOS, Windows Phone. Обзор сред программирования. Эмуляторы. Стандартный эмулятор Android. Альтернативные эмуляторы. Возможности отладки на реальных устройствах

Тема 2. Виды приложений и их структура

Основные виды мобильных приложений. Безопасность. Архитектура приложения, основные компоненты. Активности (Activities). Сервисы (Services).

Тема 3 Основы разработки интерфейсов мобильных приложений

Визуальный дизайн интерфейсов. Графический дизайн и пользовательские интерфейсы. Визуальный информационный дизайн. Строительные блоки визуального дизайна интерфейсов.

Тема 4. Основы разработки многооконных приложений

Многооконные приложения. Работа с диалоговыми окнами. Диалоговые окна. Использование класса Dialog. Уведомления. Всплывающие подсказки. Особенности разработки приложения, содержащего несколько активностей. Перелистывание (Swipe).

Тема 5 Использование возможностей смартфона в приложениях

Отличительные особенности смартфонов. Сенсорное (touch) управление. Сбор данных. Сенсорных событиях. Распознавание жестов. Работа с мультимедиа. Использование встроенной камеры. Взаимодействие с системами позиционирования.

Тема 6. Использование библиотек.

Подключение библиотек. Обзор популярных библиотек о Android Support Library. Сторонние библиотеки. Библиотеки специального назначения. Прикладные библиотеки. Безопасность использования подключаемых библиотек.

Тема 7. . Работа с базами данных, графикой и анимацией. Разработка игр

Основы работы с базами данных. SQLite. Анимация. 2D и 3D графика. Основные принципы разработки игровых приложений для смартфонов.

4. Учебно-методическое обеспечение самостоятельной работы обучающихся

Самостоятельная работа обучающихся включает в себя подготовку к контрольным вопросам и заданиям для текущего контроля и промежуточной аттестации по итогам освоения дисциплины приведенным в п. 5.

Для обеспечения самостоятельной работы обучающихся используются:

Электронные курсы, созданные в системе электронного обучения ННГУ:

Технологии разработки приложений для мобильных устройств,

<https://e-learning.unn.ru/course/view.php?id=11733>.

5. Фонд оценочных средств для текущего контроля успеваемости и промежуточной аттестации по дисциплине (модулю)

5.1 Типовые задания, необходимые для оценки результатов обучения при проведении текущего контроля успеваемости с указанием критериев их оценивания:

5.1.1 Типовые задания (оценочное средство - Задания) для оценки сформированности компетенции ПК-11:

Подберите тематику для будущего приложения. Обоснуйте актуальность.

Примеры:

- **Анализ существующих решений:** Проведите исследование рынка и найдите 2-3 существующих мобильных приложения, которые решают схожую проблему или находятся в той же тематической области, что и ваше будущее приложение. Опишите их основные функции, пользовательский интерфейс (UI) и пользовательский опыт (UX). Выделите их сильные и слабые стороны.
- **Определение целевой аудитории:** Детально опишите целевую аудиторию вашего будущего приложения. Включите демографические характеристики (возраст, пол, местоположение), психографические характеристики (интересы, потребности, образ жизни), а также их потенциальные сценарии использования вашего приложения.
- **Разработка User Story Map:** Составьте карту пользовательских историй (User Story Map) для вашего будущего приложения. Определите основные этапы взаимодействия пользователя с приложением и разбейте каждый этап на конкретные задачи (пользовательские истории).
- **Создание скетчей интерфейса (Wireframes):** Нарисуйте от руки или с помощью простых инструментов (например, Figma, Miro) скетчи основных экранов вашего будущего приложения. Сосредоточьтесь на компоновке элементов интерфейса и навигации между экранами.
- **Проектирование архитектуры приложения:** Предложите высокоуровневую архитектуру вашего будущего приложения. Опишите основные компоненты (например, экраны, сервисы, хранилище данных) и их взаимодействие. Подумайте о том, какие Activities и Services могут потребоваться.
- **Планирование использования возможностей смартфона:** Определите, какие специфические возможности смартфонов (из Тема 5) могут быть полезны для вашего приложения. Опишите, как вы планируете использовать сенсорное управление, камеру, геолокацию, мультимедиа или другие функции для улучшения пользовательского опыта.
- **Предварительный выбор библиотек:** Исходя из предполагаемой функциональности вашего приложения, определите 2-3 потенциально полезные библиотеки (из Тема 6). Обоснуйте свой выбор и опишите, какие задачи эти библиотеки могут помочь решить.
- **Концепция хранения данных:** Опишите, как ваше приложение будет хранить данные. Если предполагается использование базы данных, подумайте о том, какие данные необходимо будет хранить и как они будут организованы (на уровне концепции, без детализации схемы SQLite).
- **Идеи по визуальному стилю:** Разработайте несколько ключевых идей для визуального стиля вашего приложения, опираясь на принципы визуального дизайна интерфейсов (из Тема 3). Опишите цветовую палитру, типографику и общее настроение, которое вы хотите создать.

- **Проработка многооконности и навигации:** Опишите, как будет организована навигация между различными экранами вашего приложения. Предусмотрите ли вы использование диалоговых окон, уведомлений или всплывающих подсказок (из Тема 4)? Обоснуйте свой выбор навигационной модели.

5.1.2 Типовые задания (оценочное средство - Задания) для оценки сформированности компетенции ПК-6:

Создайте мобильное приложение:

- **Настройка среды разработки:** Установите и настройте необходимую среду разработки для выбранной вами платформы (Android Studio для Android, Xcode для iOS или кроссплатформенные инструменты, такие как Flutter или React Native). Ознакомьтесь с основными компонентами IDE.
- **Создание нового проекта:** Создайте новый проект в выбранной среде разработки, используя подходящий шаблон. Дайте проекту осмысленное имя, соответствующее вашему приложению.
- **Разработка макета первого экрана:** Начните с создания макета пользовательского интерфейса для первого ключевого экрана вашего приложения (например, экран приветствия, главный экран или экран авторизации). Используйте инструменты визуального проектирования или XML/SwiftUI для определения расположения основных элементов (текстовых полей, кнопок, изображений и т.д.).
- **Реализация базовой навигации:** Если ваше приложение предполагает несколько экранов, настройте базовую навигацию между ними. Реализуйте переходы между первым экраном и одним-двумя последующими экранами с использованием соответствующих компонентов (например, Intent для Android, Segue для iOS, Navigator для Flutter/React Native).
- **Обработка пользовательского ввода:** Реализуйте обработку пользовательского ввода на одном из экранов. Например, если это экран авторизации, настройте получение текста из полей ввода логина и пароля и выведите их в консоль или временный текстовый элемент при нажатии кнопки "Войти".
- **Отображение статических данных:** Реализуйте отображение статических данных на одном из экранов вашего приложения. Это может быть заранее определенный текст, список элементов или изображение, встроенное в ресурсы приложения.
- **Использование одного из API смартфона:** Интегрируйте в ваше приложение использование одной из базовых возможностей смартфона (из Тема 5). Например:
 - Для Android: получите доступ к камере и отобразите превью, или получите текущие координаты местоположения (с запросом разрешений).
 - Для iOS: получите доступ к фотобиблиотеке и отобразите выбранное изображение, или запросите доступ к геолокации.
 - Для кроссплатформенных решений: реализуйте функцию, специфичную для выбранного фреймворка, которая демонстрирует взаимодействие с аппаратными возможностями.
- **Подключение и использование простой библиотеки:** Подключите и используйте одну простую стороннюю библиотеку (из Тема 6), которая может добавить полезную функциональность в ваше приложение (например, библиотеку для отображения красивых кнопок, карусели изображений или базовой работы с сетью).
- **Реализация сохранения и загрузки простых данных:** Реализуйте простую функцию сохранения и загрузки небольшого объема данных (например, настроек пользователя или временных заметок) с использованием локального хранилища (например, SharedPreferences для Android, UserDefaults для iOS, AsyncStorage для React Native).

- **Тестирование на эмуляторе/реальном устройстве:** Запустите ваше приложение на эмуляторе или физическом устройстве. Протестируйте реализованную функциональность, убедитесь в корректной работе навигации, обработки ввода и отображения данных. Зафиксируйте возникшие ошибки и попытайтесь их исправить.

Критерии оценивания (оценочное средство - Задания)

Оценка	Критерии оценивания
зачтено	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «превосходно» Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «отлично», при этом хотя бы одна компетенция сформирована на уровне «отлично» Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «очень хорошо», при этом хотя бы одна компетенция сформирована на уровне «очень хорошо» Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «хорошо», при этом хотя бы одна компетенция сформирована на уровне «хорошо» Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «удовлетворительно», при этом хотя бы одна компетенция сформирована на уровне «удовлетворительно»
не зачтено	Хотя бы одна компетенция сформирована на уровне «неудовлетворительно», ни одна из компетенций не сформирована на уровне «плохо» Хотя бы одна компетенция сформирована на уровне «плохо»

5.2. Описание шкал оценивания результатов обучения по дисциплине при промежуточной аттестации

Шкала оценивания сформированности компетенций

Уровень сформированности компетенций (индикатор достижения компетенций)	плохо	неудовлетворительно	удовлетворительно	хорошо	очень хорошо	отлично	превосходно
	не зачтено		зачтено				
<u>Знания</u>	Отсутствие знаний теоретического материала. Невозможность оценить полноту знаний вследствие отказа обучающегося от ответа	Уровень знаний ниже минимальных требований. Имели место грубые ошибки	Минимально допустимый уровень знаний. Допущено много негрубых ошибок	Уровень знаний в объеме, соответствующем программе подготовки. Допущено несколько негрубых ошибок	Уровень знаний в объеме, соответствующем программе подготовки. Допущено несколько несущественных ошибок	Уровень знаний в объеме, соответствующем программе подготовки. Ошибок нет.	Уровень знаний в объеме, превышающем программу подготовки.

<u>Умения</u>	Отсутствие минимальных умений. Невозможность оценить наличие умений вследствие отказа обучающегося от ответа	При решении стандартных задач не продемонстрированы основные умения. Имели место грубые ошибки	Продемонстрированы основные умения. Решены типовые задачи с негрубыми ошибками. Выполнены все задания, но не в полном объеме	Продемонстрированы все основные умения. Решены все основные задачи с негрубыми ошибками. Выполнены все задания в полном объеме, но некоторые с недочетами	Продемонстрированы все основные умения. Решены все основные задачи. Выполнены все задания в полном объеме, но некоторые с недочетами	Продемонстрированы все основные умения. Решены все основные задачи с отдельными и несущественными недочетами, выполнены все задания в полном объеме	Продемонстрированы все основные умения. Решены все основные задачи. Выполнены все задания, в полном объеме без недочетов
<u>Навыки</u>	Отсутствие базовых навыков. Невозможность оценить наличие навыков вследствие отказа обучающегося от ответа	При решении стандартных задач не продемонстрированы базовые навыки. Имели место грубые ошибки	Имеется минимальный набор навыков для решения стандартных задач с некоторыми недочетами	Продемонстрированы базовые навыки при решении стандартных задач с некоторыми недочетами	Продемонстрированы базовые навыки при решении стандартных задач без ошибок и недочетов	Продемонстрированы навыки при решении нестандартных задач без ошибок и недочетов	Продемонстрирован творческий подход к решению нестандартных задач

Шкала оценивания при промежуточной аттестации

Оценка		Уровень подготовки
зачтено	превосходно	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «превосходно», продемонстрированы знания, умения, владения по соответствующим компетенциям на уровне выше предусмотренного программой
	отлично	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «отлично».
	очень хорошо	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «очень хорошо»
	хорошо	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «хорошо».
	удовлетворительно	Все компетенции (части компетенций), на формирование которых направлена дисциплина, сформированы на уровне не ниже «удовлетворительно», при этом хотя бы одна компетенция сформирована на уровне «удовлетворительно»
не зачтено	неудовлетворительно	Хотя бы одна компетенция сформирована на уровне «неудовлетворительно».
	плохо	Хотя бы одна компетенция сформирована на уровне «плохо»

5.3 Типовые контрольные задания или иные материалы, необходимые для оценки результатов обучения на промежуточной аттестации с указанием критериев их оценивания:

5.3.1 Типовые задания (оценочное средство - Тест) для оценки сформированности компетенции ПК-6

1. Опишите, какие инструменты могут быть использованы для эмуляции различных конфигураций мобильных устройств (размер экрана, версия ОС) в процессе тестирования. Почему важно проводить тестирование на нескольких эмуляторах/реальных устройствах?
2. Предложите стратегию тестирования безопасности для мобильного приложения, работающего с конфиденциальными пользовательскими данными. Какие виды тестов (например, статический анализ кода, динамическое тестирование) вы бы включили?
3. Разработайте набор тестовых сценариев для проверки юзабилити ключевого экрана мобильного приложения (например, экран регистрации). Какие метрики вы будете использовать для оценки результатов тестирования?
4. Опишите процесс тестирования навигации между несколькими Activity в Android-приложении. Какие инструменты или методы вы будете использовать для обнаружения ошибок перехода и передачи данных между экранами?
5. Предложите план тестирования функции геолокации в мобильном приложении. Какие граничные случаи и условия следует проверить (например, отсутствие GPS-сигнала, имитация движения)?
6. При интеграции сторонней библиотеки в мобильное приложение, какие аспекты необходимо протестировать, чтобы убедиться в ее корректной работе и отсутствии негативного влияния на основное приложение?
7. Опишите стратегию тестирования локального хранения данных в мобильном приложении с использованием SQLite. Какие виды тестов (например, проверка CRUD-операций, миграция схемы) вы бы включили?
8. Предложите набор тестов для проверки производительности анимации в мобильном приложении на устройствах с различными характеристиками. Какие инструменты могут быть использованы для измерения FPS (кадров в секунду)?
9. Опишите процесс модульного тестирования отдельного компонента (например, функции обработки пользовательского ввода) в мобильном приложении с использованием фреймворков для тестирования (например, JUnit для Android).
10. Предложите сценарий интеграционного тестирования, включающий взаимодействие нескольких компонентов мобильного приложения (например, пользовательский интерфейс, бизнес-логика и локальная база данных). Как вы будете выявлять и устранять несоответствия на этом уровне?

5.3.2 Типовые задания (оценочное средство - Тест) для оценки сформированности компетенции ПК-11

1. Опишите, какие инструменты могут быть использованы для эмуляции различных конфигураций мобильных устройств (размер экрана, версия ОС) в процессе тестирования. Почему важно проводить тестирование на нескольких эмуляторах/реальных устройствах?

- **Ответ:** Для эмуляции можно использовать встроенные эмуляторы IDE (Android Emulator, iOS Simulator), а также сторонние эмуляторы (Genymotion, BlueStacks). Важность тестирования на нескольких эмуляторах/реальных устройствах заключается в выявлении проблем, **специфичных для определенных устройств, разрешений экрана, версий ОС и аппаратных особенностей**, которые могут не проявляться на одном тестовом окружении.
2. Предложите стратегию тестирования безопасности для мобильного приложения, работающего с конфиденциальными пользовательскими данными. Какие виды тестов (например, статический анализ кода, динамическое тестирование) вы бы включили?
- **Ответ:** Стратегия тестирования безопасности должна включать:
 - **Статический анализ кода:** Для выявления потенциальных уязвимостей в исходном коде.
 - **Динамическое тестирование:** Проведение тестов на работающем приложении для обнаружения уязвимостей во время выполнения (например, тестирование на проникновение, fuzzing).
 - Тестирование **аутентификации и авторизации, шифрования данных** (при передаче и хранении), **защиты от инъекций, обработки сессий, защиты от обратной инженерии.**
3. Разработайте набор тестовых сценариев для проверки юзабилити ключевого экрана мобильного приложения (например, экран регистрации). Какие метрики вы будете использовать для оценки результатов тестирования?
- **Ответ:** Набор тестовых сценариев может включать проверку:
 - Корректности работы всех элементов ввода и кнопок.
 - Понятности и информативности сообщений об ошибках.
 - Соответствия интерфейса гайдлайнам платформы.
 - Адаптивности к различным размерам экрана.
 - Доступности для пользователей с ограниченными возможностями (например, тестирование с TalkBack/VoiceOver).
 - Метрики для оценки: **время выполнения задачи, количество ошибок, уровень удовлетворенности пользователей** (через опросы), **эффективность выполнения задач.**
4. Опишите процесс тестирования навигации между несколькими Activity в Android-приложении. Какие инструменты или методы вы будете использовать для обнаружения ошибок перехода и передачи данных между экранами?
- **Ответ:** Процесс тестирования навигации включает:
 - Проверку корректности перехода между всеми предусмотренными экранами при различных действиях пользователя.
 - Тестирование передачи данных между Activity (корректность, целостность).
 - Проверку жизненного цикла Activity при переходах (например, сохранение и восстановление состояния).
 - Использование **инструментов мониторинга логов (Logcat в Android Studio)** для отслеживания ошибок и передачи Intent-ов, **инструментов UI-тестирования (Espresso, UI Automator)** для автоматизации сценариев навигации.

5. Предложите план тестирования функции геолокации в мобильном приложении. Какие граничные случаи и условия следует проверить (например, отсутствие GPS-сигнала, имитация движения)?

- **Ответ:** План тестирования геолокации должен включать:
 - Проверку получения корректных координат при наличии стабильного GPS-сигнала.
 - Тестирование обработки ситуаций **отсутствия GPS-сигнала** (например, отображение информационного сообщения).
 - Проверку работы приложения при **имитации движения** (с использованием эмуляторов или специальных инструментов).
 - Тестирование **различных режимов точности** определения местоположения.
 - Проверку **запросов разрешений** на использование геолокации.

6. При интеграции сторонней библиотеки в мобильное приложение, какие аспекты необходимо протестировать, чтобы убедиться в ее корректной работе и отсутствии негативного влияния на основное приложение?

- **Ответ:** Необходимо протестировать:
 - **Функциональность библиотеки:** Соответствие заявленным возможностям и корректность работы в контексте приложения.
 - **Совместимость:** Отсутствие конфликтов с другими библиотеками и кодом приложения.
 - **Производительность:** Влияние библиотеки на скорость работы и потребление ресурсов приложения.
 - **Безопасность:** Проверка библиотеки на известные уязвимости.
 - **Обработка ошибок:** Корректное поведение приложения при возникновении ошибок в библиотеке.

7. Опишите стратегию тестирования локального хранения данных в мобильном приложении с использованием SQLite. Какие виды тестов (например, проверка CRUD-операций, миграция схемы) вы бы включили?

- **Ответ:** Стратегия тестирования SQLite должна включать:
 - Проверку **CRUD-операций** (создание, чтение, обновление, удаление) для всех таблиц.
 - Тестирование **целостности данных и соответствия схеме**.
 - Проверку **производительности запросов** при больших объемах данных.
 - Тестирование **миграции схемы** при обновлении версии приложения.
 - Тестирование **обработки ошибок** при работе с базой данных.

8. Предложите набор тестов для проверки производительности анимации в мобильном приложении на устройствах с различными характеристиками. Какие инструменты могут быть использованы для измерения FPS (кадров в секунду)?

- **Ответ:** Набор тестов может включать проверку:
 - Плавности анимации при различных нагрузках на устройство.
 - Корректности работы анимации на устройствах с разной производительностью.
 - Отсутствия "лагов" и "зависаний".
 - Инструменты для измерения FPS: **Android Studio Profiler**, инструменты разработчика на iOS (**Instruments**), сторонние утилиты.

9. Опишите процесс модульного тестирования отдельного компонента (например, функции обработки пользовательского ввода) в мобильном приложении с использованием фреймворков для тестирования (например, JUnit для Android).

- **Ответ:**Процесс модульного тестирования включает:
 - Изоляцию тестируемого компонента от зависимостей (с использованием моков или стабов).
 - Написание тестовых случаев для различных сценариев (позитивные, негативные, граничные случаи).
 - Использование **фреймворков для модульного тестирования (например, JUnit для Android, XCTest для iOS)** для написания и запуска тестов.
 - Проверку ожидаемых результатов выполнения компонента для каждого тестового случая.

10.Предложите сценарий интеграционного тестирования, включающий взаимодействие нескольких компонентов мобильного приложения (например, пользовательский интерфейс, бизнес-логика и локальная база данных). Как вы будете выявлять и устранять несоответствия на этом уровне?

- **Ответ:**Сценарий интеграционного тестирования может включать, например, процесс регистрации пользователя: ввод данных в UI, их обработка бизнес-логикой, сохранение в локальной базе данных и отображение подтверждения в UI. Выявление несоответствий осуществляется через:
 - **Анализ логов** взаимодействия компонентов.
 - **Проверку состояния системы** после выполнения сценария (например, данные в базе данных).
 - **UI-тестирование** для проверки корректности отображения данных и реакции на действия пользователя.
 - Устранение несоответствий включает **анализ причин ошибки** в одном из взаимодействующих компонентов и **исправление кода** с последующим повторным тестированием.

Критерии оценивания (оценочное средство - Тест)

Оценка	Критерии оценивания
превосходно	Не оценивается
отлично	Все тестовые задания решены на 95% и более
очень хорошо	не оценивается
хорошо	Все тестовые задания решены на 85% и более
удовлетворительно	Все тестовые задания решены на 75% и более
неудовлетворительно	Все тестовые задания решены менее 75%
плохо	не оценивается

6. Учебно-методическое и информационное обеспечение дисциплины (модуля)

Основная литература:

1. Коржинский С.Н. Разработка мобильных приложений : Учебник / С.Н. Коржинский. - Москва : КноРус, 2024. - 421 с. - Режим доступа: book.ru. - ISBN 978-5-406-12577-9., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=878062&idb=0>.
2. Ретабоул С. Android NDK. Разработка приложений под Android на C/C++ : монография / Ретабоул С. - Москва : ДМК-пресс, 2023. - 497 с. - ISBN 978-5-89818-542-8., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=912784&idb=0>.

Дополнительная литература:

1. Волшебство Kotlin / Сомон П.-И. - Москва : ДМК-пресс, 2020., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=664875&idb=0>.
2. Разработка мобильных приложений в среде Android Studio / Пирская Л.В. - Москва : ЮФУ, 2019., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=789962&idb=0>.
3. Петросян Л. Э. Разработка мобильных приложений на языке Kotlin : учебное пособие для вузов / Петросян Л. Э., Гусев К. В.; Гусев К. В. - 2-е изд., стер. - Санкт-Петербург : Лань, 2025. - 104 с. - Книга из коллекции Лань - Информатика. - ISBN 978-5-507-52328-3., <https://e-lib.unn.ru/MegaPro/UserEntry?Action=FindDocs&ids=930414&idb=0>.

Программное обеспечение и Интернет-ресурсы (в соответствии с содержанием дисциплины):

Android Studio: (<https://developer.android.com/studio>) - Официальная интегрированная среда разработки (IDE) от Google для создания Android-приложений. Включает в себя эмулятор Android, инструменты для отладки, профилирования и визуального проектирования интерфейсов. Это основной инструмент для практической работы с платформой Android, упомянутой.

Официальная документация Android Developers: (<https://developer.android.com/docs>) - Обширный ресурс от Google, содержащий руководства, примеры кода, справочную информацию по API Android. Необходим для глубокого понимания всех аспектов разработки, от основ до использования возможностей устройства и работы с базами данных.

Руководство по Material Design: (<https://material.io/design>) - Официальное руководство от Google по созданию визуально привлекательных и интуитивно понятных пользовательских интерфейсов для Android и других платформ. Охватывает принципы визуального дизайна, компоненты интерфейса и лучшие практики.

Stack Overflow (раздел Android): (<https://stackoverflow.com/questions/tagged/android>) - Популярный форум вопросов и ответов для программистов. Здесь можно найти решения множества проблем, возникающих в процессе разработки Android-приложений, и обсуждения различных аспектов, включая работу с библиотеками и многооконными приложениями.

SQLite Documentation: (<https://www.sqlite.org/docs.html>) - Официальная документация по SQLite, встроенной легковесной реляционной базе данных, которая часто используется в мобильных приложениях. Содержит подробное описание синтаксиса SQL, API и принципов работы с базой данных.

Ray Wenderlich (Mobile Development Tutorials): (<https://www.raywenderlich.com/>) - Популярный веб-сайт с высококачественными туториалами и курсами по разработке мобильных приложений для iOS и Android (а также кроссплатформенной разработке). Может быть полезен для изучения различных тем курса на практических примерах.

7. Материально-техническое обеспечение дисциплины (модуля)

Учебные аудитории для проведения учебных занятий, предусмотренных образовательной программой, оснащены мультимедийным оборудованием (проектор, экран), техническими средствами обучения, компьютерами.

Помещения для самостоятельной работы обучающихся оснащены компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечены доступом в электронную информационно-образовательную среду.

Программа составлена в соответствии с требованиями ОС ННГУ по направлению подготовки/специальности 09.03.03 - Прикладная информатика.

Автор(ы): Беспалько Анна Андреевна, кандидат педагогических наук, доцент.

Заведующий кафедрой: Поляков Евгений Артурович, кандидат педагогических наук.

Программа одобрена на заседании методической комиссии от 28.12.2024, протокол № 21.