

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Соловьев Дмитрий Александрович

Должность: Декан факультета биотехнологий

Дата подписания: 03.09.2025 13:54:20

Уникальный программный ключ:

528882d78e671566ab07f3e1ba2172f735a12

Приложение 1

МИНИСТЕРСТВО СЕЛЬСКОГО ХОЗЯЙСТВА РОССИЙСКОЙ ФЕДЕРАЦИИ



Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Саратовский государственный университет
генетики, биотехнологии и инженерии имени Н.И.
Вавилова»

УТВЕРЖДАЮ

И.о. заведующего кафедрой

 /Ключиков А.В./

« 12 » апреля 2024 г.

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

Дисциплина	Разработка мобильных приложений
Направление подготовки	09.03.03 Прикладная информатика
Направленность (профиль)	Проектирование информационных систем
Квалификация выпускника	Бакалавр
Нормативный срок обучения	4 года
Форма обучения	Очная
Кафедра-разработчик	Цифровое управление процессами в АПК
Ведущий преподаватель	Леонтьев А.А., доцент

Разработчик(и): доцент, Леонтьев А.А.

ассистент, Гречечук Ю.Н.




Саратов 2024

Содержание

1	Перечень компетенций с указанием этапов их формирования в процессе освоения ОПОП	3
2	Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания	5
3	Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы.....	9
4	Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы их формирования	23

1. Перечень компетенций с указанием этапов их формирования в процессе освоения ОПОП

В результате изучения дисциплины «Разработка мобильных приложений» обучающиеся, в соответствии с ФГОС ВО по направлению подготовки 09.03.03 Прикладная информатика, утвержденного приказом Министерства образования и науки РФ от 19.09.2017 г. № 922, формируют следующие компетенции, указанные в таблице 1.

Таблица 1

Формирование компетенций в процессе изучения дисциплины

Компетенция		Индикаторы достижения компетенций	Этапы формирования компетенции в процессе освоения ОПОП (семестр)*	Виды занятий для формирования компетенции	Оценочные средства для оценки уровня сформированности компетенции
Код	Наименование				
1	2	3	4	5	6
ПК-6	Использует дизайнерские, компьютерные и общественные знания для создания и изменения программ и приложений объединяющих текстовые графические мультипликационные изобразительные и звуковые и видеоматериалы, а также другие интерактивные средства	ПК-6.2 Способен применять программные средства системного, прикладного и специального назначения, инструментальные средства, языки и системы программирования для разработки мобильных приложений	6	Лекция, лабораторное занятие.	Тестовые задания/ практическая работа /устный опрос/письменный опрос.

Примечание:*

Компетенция ПК-6 также формируется в ходе изучения следующих дисциплин и практик:

Б1.В.05	Проектирование геоинформационных систем
Б1.В.06	Технологии разработки веб-систем
Б1.В.12	Разработка компьютерных игр
Б1.В.21	Компьютерное зрение
Б1.В.ДВ.01.01	Технологии геопространственного анализа
Б1.В.ДВ.01.02	Визуализация геопространственных данных
Б2.В.02(П)	Технологическая (проектно-технологическая) практика

Б3.01(Д)	Выполнение и защита выпускной квалификационной работы
ФТД.01	Управление робототехническими комплексами

2. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

Таблица 2

Перечень оценочных материалов

№ п/п	Наименование оценочного материала	Краткая характеристика оценочного материала	Представление оценочного средства в ОМ
1	Устный опрос / письменный опрос	средство контроля, организованное как специальная беседа педагогического работника с обучающимся или письменный ответ на темы, связанные с изучаемой дисциплиной и рассчитанной на выяснение объема знаний обучающегося по определенному разделу, теме, проблеме и т.п.	вопросы по темам дисциплины: – перечень вопросов для устного опроса – перечень вопросов для письменного опроса
2	Практическая работа	средство, направленное на освоение методов практического использования современных компьютеров для обработки информации.	лабораторные работы
3	Тестирование	метод, который позволяет выявить уровень знаний, умений и навыков, способностей и других качеств личности, а также их соответствие определенным нормам путем анализа способов выполнения обучающимися ряда специальных заданий	банк тестовых заданий

Таблица 3

Программа оценивания по контролируемой дисциплине

№ п/п	Контролируемые разделы (темы дисциплины)	Код контролируемой компетенции (или ее части)	Наименование оценочного средства
1	2	3	4
1	Основы разработки мобильных приложений Основные понятия: платформы, архитектура, жизненный цикл приложения, основные инструменты разработчика.	ПК – 6	Письменный опрос (вопросы входного контроля) Устный опрос (вопросы для проведения устного опроса 10, вопросы ПК1)
2	Архитектура мобильных приложений Паттерны проектирования в мобильной разработке. MVVM, MVC и другие подходы к организации кода.	ПК – 6	Лабораторная работа №1 Устный опрос (вопросы для проведения устного опроса 20-24, вопросы ПК1) Письменный опрос (вопросы для проведения письменного опроса 11-16) Тест (Тест №1)
3	Языки программирования для мобильных платформ Особенности синтаксиса и базовые конструкции.	ПК – 6	Лабораторная работа №2
4	Инструменты разработки (IDE) Android Studio, Xcode. Настройка среды разработки, создание первого проекта.	ПК – 6	Устный опрос (вопросы для проведения устного опроса 25-29, вопросы ПК2) Письменный опрос (вопросы для проведения письменного опроса 17-25) Тест (Тест №2)
5	UI/UX дизайн мобильных приложений Принципы создания пользовательского интерфейса. Компоненты UI, адаптивность под разные экраны.	ПК – 6	Лабораторная работа №3 Устный опрос (вопросы для проведения устного опроса 30-34, вопросы ПК3) Письменный опрос (вопросы для проведения письменного опроса 26-32).
6	Работа с API в мобильных приложениях Взаимодействие с внешними сервисами. HTTP-запросы, RESTful API, JSON-парсинг.	ПК – 6	Лабораторная работа №4
7	Базы данных в мобильных приложениях SQLite, Room, Core Data. Хранение и управление данными на устройстве.	ПК – 6	Лабораторная работа №5
8	Межкомпонентное взаимодействие Навигация между экранами, передача данных между компонентами, использование Intent.	ПК – 6	Лабораторная работа №6
9	Работа с файловой системой Чтение и запись файлов, работа с внутренней и внешней памятью устройства.	ПК – 6	Лабораторная работа №7
10	Уведомления и фоновые задачи Push-уведомления, background	ПК – 6	Устный опрос (вопросы для проведения устного опроса 35-39,

№ п/п	Контролируемые разделы (темы дисциплины)	Код контролируемой компетенции (или ее части)	Наименование оценочного средства
1	2	3	4
	services, выполнение задач в фоновом режиме.		вопросы РК3) Письменный опрос (вопросы для проведения письменного опроса 32-40).
11	Кроссплатформенная разработка Flutter, React Native. Преимущества и недостатки кроссплатформенных решений.	ПК – 6	Устный опрос (вопросы для проведения устного опроса 40-44, вопросы РК3)
12	Сетевые запросы и безопасность HTTPS, авторизация, аутентификация, защита данных при передаче через сеть.	ПК – 6	Устный опрос (вопросы для проведения устного опроса 45-49, вопросы РК3)
13	Тестирование мобильных приложений Типы тестирования, инструменты для автоматизации тестирования, эмуляторы и реальные устройства.	ПК – 6	Устный опрос (вопросы для проведения устного опроса 50-54, вопросы РК3)
14	Оптимизация производительности Профилирование приложения, устранение узких мест, оптимизация использования ресурсов.	ПК – 6	Устный опрос (вопросы для проведения устного опроса 55-60, вопросы РК3)
15	Публикация мобильных приложений Подготовка приложения к выпуску, требования магазинов приложений (Google Play, App Store).	ПК – 6	Лабораторная работа №8
16	Разработка простого мобильного приложения Создание приложения для Android или iOS с использованием базовых функций.	ПК – 6	Лабораторная работа №9
17	Кроссплатформенное мобильное приложение Разработка приложения с использованием кроссплатформенных технологий (Flutter, React Native).	ПК – 6	Лабораторная работа №10

Таблица 4

Описание показателей и критериев оценивания компетенций по дисциплине на различных этапах их формирования, описание шкал оценивания

Код компетенции, этапы освоения компетенции	Индикаторы достижения компетенций	Показатели и критерии оценивания результатов обучения			
		ниже порогового уровня (неудовлетворительно)	пороговый уровень (удовлетворительно)	продвинутый уровень (хорошо)	высокий уровень (отлично)
1	2	3	4	5	6
ПК-6, 6 семестр	ПК-6.2 Способен применять программные средства системного, прикладного и специального назначения, инструментальные средства, языки и системы программирования для разработки мобильных приложений	Обучающийся не знает значительной части программного материала, включая основные понятия разработки мобильных приложений, архитектуры мобильных систем, языков программирования и инструментов разработки; Плохо ориентируется в материалах по разработке пользовательского интерфейса (UI/UX), работе с API, базами данных и сетевыми запросами; Не владеет методами разработки мобильных приложений для различных	обучающийся демонстрирует знание только основного материала, но не знает деталей или допускает неточности в формулировках; Знаком с основными понятиями разработки мобильных приложений, языками программирования (Kotlin, Swift) и инструментами разработки (Android Studio, Xcode); Может создать простое мобильное приложение, но допускает ошибки в реализации	обучающийся демонстрирует глубокое знание материала, включая архитектурные паттерны (MVVM, MVC), работу с API, базами данных и сетевыми запросами; Владеет навыками разработки мобильных приложений для конкретных платформ (Android, iOS) или кроссплатформенных решений (Flutter, React Native); Создает функциональные мобильные приложения с учетом требований к безопасности, производительности и пользователь	обучающийся демонстрирует исчерпывающее и последовательное знание всех аспектов разработки мобильных приложений, включая архитектуру, дизайн, интеграцию с API, работу с базами данных и обеспечение безопасности; Создает высококачественные мобильные приложения, соответствующие современным стандартам и требованиям рынка; Четко и логично излагает материал, хорошо ориентируется в теоретических и практических аспектах разработки; Успешно решает сложные задачи, связанные с оптимизацией производительности, адаптивностью

		платформ (Android, iOS) и кроссплатформенных технологий; Не демонстрирует навыки создания функциональных и безопасных мобильных приложений.	функционала, дизайна или взаимодействия с внешними сервисами; Нарушает логическую последовательность при объяснении процессов разработки или решении задач.	кому опыту; Успешно применяет современные инструменты и технологии, но может иметь отдельные пробелы в оптимизации кода или сложных аспектах разработки.	под разные устройства и версии операционных систем; Готов адаптироваться к изменяющимся условиям задач и уверенно работает с различными инструментами и технологиями.
--	--	---	---	--	---

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

3.1 Входной контроль

Цель проведения входного контроля: оценить уровень знаний и умений в области информатики, полученных на предыдущих уровнях обучения.

Критерии оценки входного контроля: оценка теоретических знаний и практических умений проводится в виде письменного опроса.

Вопросы входного контроля

1. Что такое мобильное приложение?
2. Какие основные операционные системы используются для мобильных устройств?
3. Какие языки программирования применяются для разработки под Android и iOS?
4. Что такое IDE?
5. Назовите популярные среды разработки для создания мобильных приложений.
6. Что такое архитектурный паттерн MVVM?
7. Какие инструменты используются для работы с API в мобильной разработке?
8. Что такое RESTful API и как оно взаимодействует с мобильным приложением?
9. Что такое база данных в контексте мобильных приложений?
10. Как организуется навигация между экранами в мобильном приложении?
11. Что такое Intent в Android или Segue в iOS?

12. Что такое UI/UX дизайн?
13. Какие ключевые принципы необходимо учитывать при создании пользовательского интерфейса для мобильных приложений?
14. Какие способы хранения данных существуют в мобильных приложениях помимо баз данных?
15. Что такое кроссплатформенная разработка?

3.2 Текущий контроль

Текущий контроль по дисциплине «Разработка мобильных приложений» позволяет оценить степень восприятия учебного материала и проводится для оценки результатов изучения разделов/тем дисциплины.

Текущий контроль проводится в виде:

- тематического контроля: по итогам изучения отдельных тем дисциплины;
- рубежного контроля: по итогам изучения раздела или нескольких разделов дисциплины.

3.2.1 Тестовые задания

По дисциплине «Разработка мобильных приложений» предусмотрено проведение следующих видов тестирования: письменное. Письменное тестирование рассматривается как: рубежный контроль по итогам изучения раздела или нескольких разделов дисциплины. Результаты тестирования учитываются при проведении промежуточной аттестации.

Пример тестовых заданий

Тест №1

Задание 1

Что такое мобильное приложение?

- А) Программа, работающая только на компьютерах
- Б) Программа, предназначенная для работы на мобильных устройствах
- В) Программа, работающая исключительно в браузере
- Г) Программа, используемая только для игр

Задание 2

Какие языки программирования используются для разработки под Android?

- А) Python, Ruby, PHP
- Б) Kotlin, Java, C#
- В) Swift, Objective-C, JavaScript
- Г) C++, Perl, Bash

Задание 3

Что такое RESTful API?

- А) Средство для создания пользовательского интерфейса
- Б) Архитектурный стиль для взаимодействия клиент-сервер
- В) Инструмент для оптимизации баз данных
- Г) Фреймворк для кроссплатформенной разработки

Задание 4

Какой инструмент используется для отладки HTTP-запросов в мобильной разработке?

- А) Git
- Б) Postman
- В) Gradle
- Г) Firebase

Задание 5

Что такое MVVM?

- А) Методология тестирования мобильных приложений
- Б) Архитектурный паттерн для организации кода
- В) Инструмент для управления версиями
- Г) Язык программирования

Задание 6

Какие типы баз данных используются в мобильных приложениях?

- А) Только облачные базы данных
- Б) SQLite, Room, Core Data
- В) MySQL, PostgreSQL
- Г) Excel таблицы

Тест №2

Задание 1

Что такое кроссплатформенная разработка?

- А) Разработка приложений только для одной платформы
- Б) Разработка приложений для нескольких платформ с использованием общего кода
- В) Разработка приложений без использования кода
- Г) Разработка приложений только для веба

Задание 2

Как называется файл конфигурации проекта в Android Studio?

- А) build.gradle
- Б) config.json
- В) settings.xml

Г) project.ini

Задание 3

Какой фреймворк используется для кроссплатформенной разработки?

А) Android Studio

Б) Xcode

В) Flutter

Г) Photoshop

Задание 4

Какой метод используется для обновления пользовательского интерфейса в Android после выполнения сетевого запроса?

А) onCreate()

Б) onResume()

В) runOnUiThread()

Г) onStart()

Задание 5

Какой из перечисленных фреймворков является кроссплатформенным для разработки мобильных приложений?

А) Android Studio

Б) Xcode

В) Flutter

Г) Photoshop

Задание 6

Какой тип данных используется в JSON для представления массива?

А) { }

Б) []

В) ()

Г) < >

3.2.3 Лабораторная работа

Тематика лабораторных работ устанавливается в соответствии с рабочей модульной программой по данной дисциплине. Перечень тем лабораторных работ приведен в разделе 2 в таблице 4 «Программа оценивания по контролируемой дисциплине».

Лабораторные работы выполняются в соответствии с Методическими указаниями по выполнению лабораторных работ по дисциплине «Разработка мобильных приложений».

3.2.4 Контрольные вопросы

Контрольные вопросы используются при проведении как письменных, так и устных опросов. Ответ на подобного рода вопрос подразумевает краткое

изложение теоретического материала.

Вопросы для проведения устного опроса:

1. Что такое мобильное приложение? Какие существуют типы мобильных приложений?
2. Назовите основные операционные системы для мобильных устройств.
3. Какие языки программирования используются для разработки под Android?
4. Какие языки программирования используются для разработки под iOS?
5. Что такое IDE? Приведите примеры популярных IDE для мобильной разработки.
6. Что такое архитектурный паттерн MVC? Опишите его компоненты.
7. Что такое архитектурный паттерн MVVM? В чем его преимущества перед MVC?
8. Как организуется навигация между экранами в Android? Что такое Intent?
9. Как организуется навигация между экранами в iOS? Что такое Segue?
10. Что такое API? Для чего оно используется в мобильной разработке?
11. Какие HTTP-методы вы знаете? Для чего они применяются?
12. Что такое RESTful API? Как оно работает?
13. Что такое JSON? Для чего он используется?
14. Что такое XML? В чем отличие JSON от XML?
15. Как работать с базами данных в мобильных приложениях? Приведите примеры локальных баз данных.
16. Что такое SQLite? Как создать таблицу в SQLite?
17. Что такое Room? Как оно упрощает работу с SQLite в Android?
18. Что такое Core Data? Как оно используется в iOS?
19. Как хранить данные в файловой системе мобильного устройства?
20. Что такое SharedPreferences в Android? Для чего они используются?
21. Что такое UserDefaults в iOS? В чем их отличие от SharedPreferences?
22. Как обеспечить безопасность данных в мобильном приложении?
23. Что такое HTTPS? Почему его использование важно для безопасности?
24. Что такое кроссплатформенная разработка? Приведите примеры фреймворков.
25. Что такое Flutter? Какие его преимущества и недостатки?
26. Что такое React Native? В чем его отличие от Flutter?
27. Как тестировать мобильные приложения? Какие виды тестирования существуют?
28. Что такое эмуляторы и симуляторы? Для чего они используются?
29. Как собирать логи с мобильного устройства? Приведите примеры инструментов.
30. Что такое UI/UX дизайн? Какие принципы необходимо учитывать при создании интерфейса?
31. Что такое адаптивный дизайн? Как его реализовать в мобильных приложениях?

32. Что такое сетка (Grid) в UI? Для чего она используется?
33. Как работать с изображениями в мобильных приложениях? Как оптимизировать их размер?
34. Что такое фоновые задачи в мобильных приложениях? Как их реализовать?
35. Что такое Push-уведомления? Как их настроить?
36. Как публиковать мобильное приложение в Google Play или App Store?
37. Что такое метаданные приложения? Какие они бывают?
38. Как оптимизировать производительность мобильного приложения?
39. Что такое Garbage Collector? Как он влияет на производительность?
40. Что такое Memory Leak? Как его избежать?
41. Что такое жизненный цикл Activity в Android? Опишите его этапы.
42. Что такое жизненный цикл ViewController в iOS? Опишите его этапы.
43. Как работать с камерой в мобильном приложении? Приведите примеры.
44. Как работать с геолокацией в мобильном приложении? Приведите примеры.
45. Что такое Firebase? Какие его функции можно использовать в мобильной разработке?
46. Что такое Gradle? Для чего он используется в Android Studio?
47. Что такое CocoaPods? Для чего он используется в Xcode?
48. Как работать с анимацией в мобильных приложениях? Приведите примеры.
49. Что такое RecyclerView в Android? Как его использовать?
50. Что такое UITableView в iOS? Как его использовать?
51. Как работать с картами в мобильных приложениях? Приведите примеры.
52. Что такое WebSocket? Как его использовать в мобильной разработке?
53. Как работать с Bluetooth в мобильных приложениях? Приведите примеры.
54. Что такое NFC? Как его использовать в мобильных приложениях?
55. Как работать с уведомлениями в Android? Приведите примеры.
56. Как работать с уведомлениями в iOS? Приведите примеры.
57. Что такое Dependency Injection? Как его использовать в мобильной разработке?
58. Что такое Dagger/Hilt? Как они упрощают Dependency Injection?
59. Что такое SwiftUI? Как оно отличается от UIKit?
60. Как работать с видео в мобильных приложениях? Приведите примеры.

Вопросы для проведения письменного опроса:

1. Опишите жизненный цикл мобильного приложения. Какие этапы он включает?
2. Что такое архитектурные паттерны? Сравните MVC, MVP и MVVM.

3. Опишите процесс создания простого мобильного приложения для Android. Какие шаги необходимо выполнить?
4. Какие инструменты использует разработчик при создании пользовательского интерфейса в Android Studio?
5. Опишите процесс работы с API в мобильном приложении. Какие инструменты можно использовать для этого?
6. Как работать с базами данных в Android? Опишите процесс создания и использования SQLite.
7. Какие способы хранения данных существуют в мобильных приложениях помимо баз данных? Сравните их.
8. Опишите процесс работы с файловой системой в Android. Как создать, записать и прочитать файл?
9. Как обеспечить безопасность данных в мобильном приложении? Приведите конкретные примеры.
10. Что такое кроссплатформенная разработка? Какие преимущества и недостатки она имеет по сравнению с нативной разработкой?
11. Опишите процесс создания простого мобильного приложения с помощью Flutter. Какие шаги необходимо выполнить?
12. Как тестировать мобильные приложения? Какие виды тестирования существуют и для чего они применяются?
13. Опишите процесс публикации мобильного приложения в Google Play. Какие требования предъявляются к приложению?
14. Как оптимизировать производительность мобильного приложения? Приведите конкретные примеры.
15. Что такое Memory Leak? Как его обнаружить и исправить в мобильном приложении?
16. Как работать с Push-уведомлениями в Android? Опишите процесс настройки.
17. Как создать адаптивный дизайн для мобильного приложения? Приведите примеры инструментов и подходов.
18. Опишите процесс работы с сетевыми запросами в мобильном приложении. Какие инструменты можно использовать?
19. Что такое Retrofit? Как его использовать для работы с API в Android?
20. Как работать с GraphQL вместо RESTful API? Какие преимущества это дает?
21. Что такое Dependency Injection? Почему его использование важно в мобильной разработке?
22. Как настроить CI/CD для мобильных приложений? Приведите примеры инструментов.
23. Как работать с асинхронными задачами в Android? Приведите примеры.
24. Как работать с асинхронными задачами в iOS? Приведите примеры.
25. Что такое State Management? Какие решения существуют для управления состоянием в Flutter?

26. Что такое Provider в Flutter? Как его использовать?
27. Что такое Bloc Pattern в Flutter? Как его использовать?
28. Как работать с изображениями в Flutter? Приведите примеры.
29. Как работать с видеоплеерами в мобильных приложениях? Приведите примеры.
30. Что такое Navigation Component в Android? Как его использовать?
31. Что такое Coordinating Layout в Android? Для чего он используется?
32. Как работать с Constraint Layout в Android? Приведите примеры.
33. Что такое Auto Layout в iOS? Как его использовать?
34. Как работать с стилями и темами в Android? Приведите примеры.
35. Как работать со стилями в iOS? Приведите примеры.
36. Что такое Unit Testing в мобильной разработке? Как его применять?
37. Что такое Integration Testing в мобильной разработке? Как его применять?
38. Что такое UI Testing в мобильной разработке? Как его применять?
39. Как работать с локализацией в мобильных приложениях? Приведите примеры.
40. Как оптимизировать размер APK/IPA файла? Приведите конкретные примеры.

3.3 Рубежный контроль

Рубежный контроль по дисциплине «Разработка мобильных приложений» позволяет оценить степень усвоения учебного материала и проводится для оценки результатов изучения разделов/тем дисциплины. Рубежной контроль проводится по итогам изучения раздела или нескольких разделов дисциплины.

Критерии оценки рубежного контроля: оценка теоретических знаний и практических умений проводится в виде устного опроса.

Вопросы рубежного контроля № 1

Вопросы, рассматриваемые на аудиторных занятиях

1. Что такое мобильное приложение? Какие существуют типы мобильных приложений?
2. Какие основные операционные системы используются для мобильных устройств?
3. Какие языки программирования применяются для разработки под Android?
4. Какие языки программирования применяются для разработки под iOS?
5. Что такое IDE? Приведите примеры популярных IDE для мобильной разработки.
6. Что такое архитектурный паттерн MVC? Опишите его компоненты.
7. Что такое архитектурный паттерн MVVM? В чем его преимущества перед MVC?

8. Как организуется навигация между экранами в Android? Что такое Intent?
9. Как организуется навигация между экранами в iOS? Что такое Segue?
10. Что такое API? Для чего оно используется в мобильной разработке?
11. Какие HTTP-методы вы знаете? Для чего они применяются?
12. Что такое RESTful API? Как оно работает?
13. Что такое JSON? Для чего он используется?
14. Что такое XML? В чем отличие JSON от XML?
15. Как работать с базами данных в мобильных приложениях? Приведите примеры локальных баз данных.
16. Что такое SQLite? Как создать таблицу в SQLite?
17. Что такое Room? Как оно упрощает работу с SQLite в Android?
18. Что такое Core Data? Как оно используется в iOS?
19. Как хранить данные в файловой системе мобильного устройства?
20. Что такое SharedPreferences в Android? Для чего они используются?

Вопросы для самостоятельного изучения

1. Какие способы хранения данных существуют в мобильных приложениях помимо баз данных? Сравните их.
2. Что такое кроссплатформенная разработка? Приведите примеры фреймворков.
3. Что такое Flutter? Какие его преимущества и недостатки?
4. Что такое React Native? В чем его отличие от Flutter?
5. Как тестировать мобильные приложения? Какие виды тестирования существуют и для чего они применяются?
6. Что такое эмуляторы и симуляторы? Для чего они используются?
7. Как собирать логи с мобильного устройства? Приведите примеры инструментов.
8. Что такое UI/UX дизайн? Какие принципы необходимо учитывать при создании интерфейса?
9. Что такое адаптивный дизайн? Как его реализовать в мобильных приложениях?
10. Как создать адаптивный дизайн для разных размеров экранов?

Вопросы рубежного контроля № 2

Вопросы, рассматриваемые на аудиторных занятиях

1. Что такое клиент-серверная архитектура? Приведите примеры ее использования.
2. Как работать с API в мобильном приложении? Приведите примеры инструментов.
3. Что такое Retrofit? Как его использовать для работы с API в Android?

4. Что такое GraphQL? Как его использовать вместо RESTful API?
5. Как обеспечить безопасность данных в мобильном приложении?
6. Что такое HTTPS? Почему его использование важно для безопасности?
7. Как работать с Push-уведомлениями в Android? Опишите процесс настройки.
8. Как работать с Push-уведомлениями в iOS? Опишите процесс настройки.
9. Что такое фоновые задачи в мобильных приложениях? Как их реализовать?
10. Как работать с геолокацией в мобильных приложениях? Приведите примеры.
11. Что такое Firebase? Какие его функции можно использовать в мобильной разработке?
12. Что такое Dependency Injection? Почему его использование важно в мобильной разработке?
13. Как настроить CI/CD для мобильных приложений? Приведите примеры инструментов.
14. Что такое State Management? Какие решения существуют для управления состоянием в Flutter?
15. Что такое Provider в Flutter? Как его использовать?
16. Что такое Bloc Pattern в Flutter? Как его использовать?
17. Как работать с видео в мобильных приложениях? Приведите примеры.
18. Как работать с картами в мобильных приложениях? Приведите примеры.
19. Как работать с Bluetooth в мобильных приложениях? Приведите примеры.
20. Что такое NFC? Как его использовать в мобильных приложениях?

Вопросы для самостоятельного изучения

1. Как работать с асинхронными задачами в Android? Приведите примеры.
2. Как работать с асинхронными задачами в iOS? Приведите примеры.
3. Что такое WebSocket? Как его использовать в мобильной разработке?
4. Как работать с камерой в мобильном приложении? Приведите примеры.
5. Что такое Memory Leak? Как его обнаружить и исправить в мобильном приложении?
6. Как оптимизировать производительность мобильного приложения? Приведите конкретные примеры.
7. Что такое Garbage Collector? Как он влияет на производительность?
8. Как создать адаптивный дизайн для мобильного приложения? Приведите примеры инструментов и подходов.
9. Как работать со стилями и темами в Android? Приведите примеры.
10. Как работать со стилями в iOS? Приведите примеры.

Вопросы рубежного контроля № 3

Вопросы, рассматриваемые на аудиторных занятиях

1. Что такое жизненный цикл Activity в Android? Опишите его этапы.
2. Что такое жизненный цикл ViewController в iOS? Опишите его этапы.
3. Как работать с RecyclerView в Android? Приведите примеры.
4. Как работать с UITableView в iOS? Приведите примеры.
5. Как работать с Constraint Layout в Android? Приведите примеры.
6. Что такое Auto Layout в iOS? Как его использовать?
7. Как работать с Navigation Component в Android? Приведите примеры.
8. Что такое Coordinating Layout в Android? Для чего он используется?
9. Как работать с локализацией в мобильных приложениях? Приведите примеры.
10. Как оптимизировать размер APK/IPA файла? Приведите конкретные примеры.
11. Что такое Unit Testing в мобильной разработке? Как его применять?
12. Что такое Integration Testing в мобильной разработке? Как его применять?
13. Что такое UI Testing в мобильной разработке? Как его применять?
14. Как работать с видеоплеерами в мобильных приложениях? Приведите примеры.
15. Как работать с изображениями в мобильных приложениях? Как оптимизировать их размер?
16. Что такое State Management? Какие решения существуют для управления состоянием в Flutter?
17. Что такое Provider в Flutter? Как его использовать?
18. Что такое Bloc Pattern в Flutter? Как его использовать?
19. Как работать с базами данных в Flutter? Приведите примеры.
20. Как работать с сетевыми запросами в Flutter? Приведите примеры.

Вопросы для самостоятельного изучения

1. Как работать с анимацией в мобильных приложениях? Приведите примеры.
2. Что такое Dependency Injection в Flutter? Как его использовать?
3. Как настроить CI/CD для Flutter-приложений? Приведите примеры инструментов.
4. Что такое Memory Leak в Flutter? Как его обнаружить и исправить?
5. Как оптимизировать производительность Flutter-приложения? Приведите конкретные примеры.
6. Как работать с платными подписками в мобильных приложениях? Приведите примеры.
7. Что такое In-App Purchases? Как их реализовать в Android и iOS?
8. Как работать с рекламой в мобильных приложениях? Приведите примеры.
9. Что такое A/B Testing в мобильной разработке? Как его применять?
10. Как анализировать пользовательское поведение в мобильных приложениях? Приведите примеры инструментов.

3.4 Промежуточная аттестация

Вид промежуточной аттестации в соответствии с учебным планом по направлению подготовки 09.03.03 Прикладная информатика – зачет.

Промежуточная аттестация по дисциплине «Разработка мобильных приложений» позволяет оценить степень усвоения учебного материала и проводится для оценки навыков и умений в области информационных технологий. проводится по итогам изучения всех разделов дисциплины.

Критерии оценки промежуточной аттестации: оценка теоретических знаний и практических умений проводится в виде устного опроса.

Тематика вопросов, выносимых на зачет

Раздел 1. Основы разработки мобильных приложений

1. Понятие и сущность мобильного приложения. Классификация типов мобильных приложений (nativo, hybrid, web).
2. Жизненный цикл мобильного приложения. Этапы разработки.
3. Архитектурные паттерны в мобильной разработке: MVC, MVP, MVVM.
4. Требования к качеству мобильных приложений. Ключевые метрики качества.
5. Процесс управления версиями мобильных приложений.

Раздел 2. Языки программирования и инструменты разработки

6. Языки программирования для Android: Kotlin, Java.
7. Языки программирования для iOS: Swift, Objective-C.
8. Интегрированные среды разработки (IDE): Android Studio, Xcode.
9. Конфигурация проекта в Android Studio: **build.gradle**, модули, зависимости.
10. Конфигурация проекта в Xcode: Storyboards, SwiftUI.
11. Git для контроля версий кода: основные команды и принципы работы.
12. CI/CD для мобильных приложений: примеры инструментов (Fastlane, Jenkins).

Раздел 3. Разработка пользовательского интерфейса (UI/UX)

13. Принципы создания эффективного пользовательского интерфейса (UI).
14. Взаимодействие с пользователем: элементы UI, события, обработка нажатий.
15. Адаптивный дизайн: поддержка разных размеров экранов и ориентаций.
16. Работа с макетами: Constraint Layout (Android), Auto Layout (iOS).
17. Стили и темы в мобильных приложениях: управление внешним видом компонентов.
18. Навигация между экранами: Navigation Component (Android), Storyboards и SwiftUI (iOS).

Раздел 4. Работа с данными

- 19.Хранение данных в мобильных приложениях: локальные и удаленные источники.
- 20.Базы данных SQLite: создание, чтение, обновление, удаление данных (CRUD).
- 21.Room для работы с базами данных в Android.
- 22.Core Data для работы с базами данных в iOS.
- 23.SharedPreferences в Android: хранение простых данных.
- 24.UserDefaults в iOS: аналог SharedPreferences.
- 25.Работа с файловой системой: создание, чтение, запись файлов.

Раздел 5. Работа с сетью и API

- 26.HTTP-протокол: основные методы (GET, POST, PUT, DELETE).
- 27.RESTful API: принципы взаимодействия клиент-сервер.
- 28.JSON: формат обмена данными. Парсинг JSON в мобильных приложениях.
- 29.Retrofit для работы с API в Android.
- 30.URLSession для работы с API в iOS.
- 31.GraphQL как альтернатива REST: преимущества и недостатки.
- 32.Postman: инструмент для тестирования API.
- 33.Swagger: документация API.

Раздел 6. Функциональные возможности мобильных приложений

- 34.Геолокация: работа с GPS и картами.
- 35.Камера: использование камеры устройства в мобильном приложении.
- 36.Push-уведомления: настройка и отправка уведомлений.
- 37.Фоновые задачи: выполнение задач в фоновом режиме.
- 38.NFC: использование технологии близкого бесконтактного соединения.
- 39.Bluetooth: взаимодействие с устройствами через Bluetooth.

Раздел 7. Безопасность и оптимизация

- 40.Безопасность мобильных приложений: шифрование данных, HTTPS.
- 41.Методы защиты от распространенных уязвимостей (XSS, SQL Injection).
- 42.Оптимизация производительности: минимизация использования ресурсов.
- 43.Memory Leak: причины возникновения и способы предотвращения.
- 44.Garbage Collector: принцип работы и влияние на производительность.

Раздел 8. Кроссплатформенная разработка

- 45.Что такое кроссплатформенная разработка? Преимущества и недостатки.
- 46.Flutter: основы языка Dart, создание первого приложения.
- 47.React Native: основы JavaScript, создание первого приложения.

- 48. Сравнение Flutter и React Native: особенности, сильные и слабые стороны.
- 49. Унификация кода: общие и специфические части для разных платформ.

Раздел 9. Тестирование мобильных приложений

- 50. Типы тестирования мобильных приложений: функциональное, нефункциональное, нагрузочное.
- 51. Эмуляторы и симуляторы: использование для тестирования.
- 52. Автоматизированное тестирование: Unit Testing, Integration Testing, UI Testing.
- 53. Инструменты для автоматизации тестирования: Espresso (Android), XCTest (iOS).
- 54. Логирование: сбор и анализ логов для отладки приложения.

Раздел 10. Публикация мобильных приложений

- 55. Подготовка приложения к выпуску: оптимизация размера APK/IPA.
- 56. Google Play Console: процесс загрузки и публикации приложения.
- 57. App Store Connect: процесс загрузки и публикации приложения.
- 58. Метаданные приложения: описание, скриншоты, ключевые слова.
- 59. Аналитика после выпуска: отслеживание показателей использования.
- 60. Обновления приложения: процесс подготовки и публикации новых версий.

4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

4.1 Процедуры оценивания знаний, умений, навыков и (или) опыта деятельности

Контроль результатов обучения, этапов и уровня формирования компетенций по дисциплине «Разработка мобильных приложений» осуществляется через проведение входного, текущего, рубежных, выходного контролей.

Формы текущего, промежуточного и итогового контроля и контрольные задания для текущего контроля разрабатываются кафедрой исходя из специфики дисциплины, и утверждаются на заседании кафедры.

4.2 Критерии оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

Описание шкалы оценивания достижения компетенций по дисциплине приведено в таблице 6.

Таблица 6

Уровень освоения компетенции	Отметка по пятибалльной системе (промежуточная аттестация)*			Описание
	«отлично»	«зачтено»	«зачтено (отлично)»	
высокий				Демонстрирует всестороннее, систематическое и глубокое знание всех аспектов разработки мобильных приложений, включая архитектуру, языки программирования, инструменты разработки, взаимодействие с API, работу с базами данных и обеспечение безопасности. Умеет свободно выполнять сложные практические задания, предусмотренные программой, такие как создание полнофункциональных мобильных приложений для различных платформ, реализация взаимодействия с внешними сервисами, оптимизация производительности и тестирование. Усвоил основную литературу и знаком с дополнительной литературой, рекомендованной программой. Проявляет творческие способности в понимании, изложении и применении

				материала, предлагая нестандартные решения задач и продемонстрировав способность к самостоятельному анализу и адаптации технологий под конкретные условия.
базовый	«хорошо»	«зачтено»	«зачтено (хорошо)»	Обнаружил полное знание учебного материала, необходимого для успешной разработки мобильных приложений. Успешно выполняет все практические задания, предусмотренные программой, включая создание простых мобильных приложений, работу с API, базами данных и пользовательским интерфейсом. Усвоил основную литературу, рекомендованную программой, и демонстрирует уверенное владение ключевыми концепциями мобильной разработки. Может работать самостоятельно, но может допускать незначительные ошибки или неточности в деталях, которые легко устраняются без помощи преподавателя.
пороговый	«удовлетворительно»	«зачтено»	«зачтено (удовлетворительно)»	Обнаружил знания основного учебного материала в объеме, достаточном для продолжения обучения и дальнейшей работы в области разработки мобильных приложений. Справляется с выполнением практических заданий, предусмотренных программой, но допускает некоторые погрешности в ответах или при реализации функционала приложения. Знаком с основной литературой, рекомендованной программой, но может испытывать затруднения при решении сложных задач или использовании продвинутых технологий. Требуется минимальная поддержки преподавателя для устранения ошибок или улучшения качества выполнения заданий.
—	«неудовлетворительно»	«не зачтено»	«не зачтено (неудовлетворительно)»	Обнаружил значительные пробелы в знаниях основного учебного материала, необходимого для разработки мобильных приложений. Допустил принципиальные ошибки в выполнении практических заданий, предусмотренных программой, что делает невозможным создание даже базовых мобильных приложений. Не способен продолжить обучение или

				приступить к профессиональной деятельности без дополнительных занятий и углубленного изучения материала. Требуется значительная поддержка преподавателя для освоения базовых концепций и выполнения элементарных задач.
--	--	--	--	---

* - форма промежуточной аттестации в семестре определяется в соответствии с таблицей 2 рабочей программы дисциплины (модуля)

4.2.1. Критерии оценки устных и письменных опросов

При устных и письменных опросах обучающийся демонстрирует:

знания: Основных понятий, принципов и современных стандартов в области разработки мобильных приложений для различных платформ (Android, iOS, кроссплатформенных решений).

Архитектурных паттернов (MVC, MVP, MVVM) и их применения в мобильной разработке.

Языков программирования (Kotlin, Swift, Dart) и инструментов разработки (Android Studio, Xcode, Flutter).

Методов работы с API, базами данных (SQLite, Room, Core Data) и файловой системой мобильных устройств.

Требований к безопасности мобильных приложений и методов защиты данных.

Типовых регламентов и подходов к тестированию мобильных приложений, включая функциональное, нефункциональное и нагрузочное тестирование.

умения: применять современные методологии и инструменты разработки мобильных приложений для создания качественного ПО.

Разрабатывать пользовательские интерфейсы (UI/UX), соответствующие требованиям адаптивности и удобства использования.

Интегрировать внешние API и работать с базами данных для хранения и обработки информации в мобильных приложениях.

Обеспечивать безопасность данных через использование протоколов HTTPS, шифрование и другие методы защиты.

Создавать и поддерживать мобильные приложения, соответствующие требованиям различных операционных систем и версий.

Выполнять тестирование мобильных приложений для выявления и устранения дефектов, а также проверки корректности принимаемых решений.

владение навыками: Навыками проектирования и реализации мобильных приложений для разных платформ с использованием современных технологий и инструментов.

Навыками работы с API, включая создание запросов, обработку ответов и интеграцию данных в мобильные приложения.

Навыками управления базами данных в мобильных приложениях для эффективного хранения и доступа к данным.

Навыками обеспечения безопасности мобильных приложений, включая защиту данных и предотвращение распространенных уязвимостей.

Навыками тестирования мобильных приложений для проверки их функциональности, производительности и совместимости.

Навыками анализа и решения проблем, возникающих на этапах разработки и тестирования мобильных приложений.

Критерии оценки

отлично	обучающийся демонстрирует: - Глубокое знание материала по основным понятиям разработки мобильных приложений, включая архитектурные паттерны, языки программирования, инструменты разработки и методологии. - Владение навыками создания высококачественных мобильных приложений для различных платформ (Android, iOS, кроссплатформенные решения) в соответствии с современными стандартами разработки. - Навыками реализации взаимодействия с API, работы с базами данных, обеспечения безопасности и оптимизации производительности. - Исчерпывающе и последовательно, четко и логично излагает материал, хорошо ориентируется в темах курса, не затрудняется с ответом даже при видоизменении заданий или сложных вопросах.
хорошо	обучающийся демонстрирует: - Знание материала без существенных неточностей, показывает хорошее понимание ключевых концепций разработки мобильных приложений. - В целом успешное выполнение практических задач, но могут быть отдельные пробелы в использовании современных технологий или методологий, таких как работа с API, базами данных или кроссплатформенной разработкой. - Уверенно применяет полученные знания на практике, хотя могут возникать затруднения при решении сложных или нетипичных задач.
удовлетворительно	обучающийся демонстрирует: - Знание только основного материала, но недостаточно глубокое понимание деталей. - Допускает неточности в формулировках и нарушает логическую последовательность при изложении теоретического материала. - Справляется с простыми практическими задачами, но испытывает трудности с более сложными аспектами разработки, такими как оптимизация производительности, безопасность или интеграция с внешними сервисами.
неудовлетворительно	обучающийся: - Не знает значительной части программного материала, плохо ориентируется в ключевых понятиях разработки мобильных приложений (архитектура, языки программирования, инструменты разработки, методологии). - Не владеет необходимыми методами и технологиями для создания функциональных мобильных приложений, такими как работа с API, базами данных, пользовательским

	интерфейсом или обеспечением безопасности. – Допускает принципиальные ошибки при выполнении практических заданий, что делает невозможным создание даже базовых мобильных приложений. –
--	--

4.2.2. Критерии оценки выполнения тестов

При выполнении письменных тестов обучающийся демонстрирует:

знания: Основных принципов и архитектурных подходов в разработке мобильных приложений (MVC, MVP, MVVM).

Языков программирования для различных платформ (Kotlin для Android, Swift для iOS, Dart для Flutter) и их особенностей.

Методов работы с API, базами данных (SQLite, Room, Core Data) и файловой системой мобильных устройств.

Правил создания пользовательских интерфейсов (UI/UX), включая адаптивность и кроссплатформенность.

Требований к безопасности мобильных приложений, методов защиты данных и использования протоколов (HTTPS, шифрование).

Стандартов и рекомендаций по оптимизации производительности мобильных приложений.

умения: разрабатывать мобильные приложения для различных платформ, используя современные инструменты и методологии.

Создавать функциональные компоненты приложений, такие как экраны, формы, навигацию и взаимодействие с внешними сервисами.

Интегрировать API и работать с базами данных для хранения и обработки информации.

Обеспечивать безопасность данных через использование протоколов HTTPS, шифрование и другие методы защиты.

Проводить тестирование мобильных приложений для выявления дефектов и проверки корректности принимаемых решений.

Подготавливать документацию для описания процессов разработки, тестирования и эксплуатации мобильных приложений.

владение навыками: Навыками проектирования и реализации мобильных приложений, соответствующих требованиям различных операционных систем и версий.

Навыками создания эффективных пользовательских интерфейсов (UI/UX), учитывающих особенности целевой аудитории и платформы.

Навыками интеграции внешних сервисов через API и работы с базами данных для обеспечения функциональности приложений.

Навыками оптимизации производительности мобильных приложений, включая работу с памятью, управление ресурсами и минимизацию загрузки процессора.

Навыками тестирования мобильных приложений для выявления ошибок, уязвимостей и несоответствий требованиям.

Навыками написания технической документации, включая описания архитектуры, алгоритмов и процессов тестирования.

Критерии оценки

Письменное тестирование рассматривается как: рубежный контроль по итогам изучения раздела или нескольких разделов дисциплины. Оценка «удовлетворительно» – от 50 до 70% верных ответов, «хорошо» – 71-85%, «отлично» – 89-100%. Результаты тестирования учитываются при проведении промежуточной аттестации.

Разработчик(и): доцент, Леонтьев А.А.

ассистент, Гречук Ю.Н.

Two handwritten signatures in blue ink are positioned above two horizontal lines. The top signature is a stylized 'AL' for A.A. Leontyev, and the bottom signature is 'Y.N.' for Y.N. Grechuk.