

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Кислова Наталья Николаевна

Должность: Проректор по УМР и качеству образования

Дата подписания: 01.09.2026 01:43:06

Уникальный программный ключ:

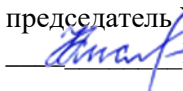
52802513f5b14a975b3e7b11008095d57260159016004f8b5ae65b96a966c055

МИНИСТЕРСТВО ПРОСВЕЩЕНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Самарский государственный социально-педагогический университет»

Информатики, прикладной математики и методики их преподавания

УТВЕРЖДАЮ

Проректор по УМР и КО,
председатель УМС СГСПУ

 Н.Н. Кислова

ПРЕДМЕТНО-МЕТОДИЧЕСКИЙ МОДУЛЬ ПО
ПРОФИЛЮ "ФИДЖИТАЛ-СПОРТ"
Разработка мобильных приложений для фиджитал-
спорта

рабочая программа дисциплины (модуля)

Закреплена за кафедрой

Информатики, прикладной математики и методики их преподавания

Учебный план

ФМФИ-626ИФo(5г)

Направление подготовки: 44.03.05 Педагогическое образование (с двумя п
Направленность (профиль): «Информатика» и «Фиджитал-спорт»

Квалификация

бакалавр

Форма обучения

очная

Общая трудоемкость **4 ЗЕТ**

Часов по учебному плану

144

в том числе:

аудиторные занятия

50

самостоятельная работа

94

Виды контроля в семестрах:

Зачет с оценкой 7

Распределение часов дисциплины по семестрам

Семестр (Курс. Номер семестра на курсе)	7(4.1)		Итого	
	УП	РПД	УП	РПД
Вид занятий				
Лекции	18	18	18	18
Лабораторные	32	32	32	32
В том числе инт.	10	10	10	10
Итого ауд.	50	50	50	50
Контактная работа	50	50	50	50
Сам. работа	94	94	94	94
Итого	144	144	144	144

Программу составил(и):

Пугач О.И.

При наличии обучающихся из числа лиц с ограниченными возможностями здоровья, которым необходим особый порядок освоения дисциплины (модуля), по их желанию разрабатывается адаптированная к ограничениям их здоровья рабочая программа дисциплины (модуля).

Рабочая программа дисциплины

Разработка мобильных приложений для фиджитал-спорта

разработана в соответствии с ФГОС ВО:

Федеральный государственный образовательный стандарт высшего образования - бакалавриат по направлению подготовки 44.03.05 Педагогическое образование (с двумя профилями подготовки) (приказ Минобрнауки России от 22.02.2018 г. № 125)

составлена на основании учебного плана:

Направление подготовки: 44.03.05 Педагогическое образование (с двумя профилями подготовки)
Направленность (профиль): «Информатика» и «Фиджитал-спорт»

утвержденного учёным советом СГСПУ от 30.04.2026г протокол № 9.

Рабочая программа одобрена на заседании кафедры

Информатики, прикладной математики и методики их преподавания

Протокол от 23.06.2026 № 11

Зав. кафедрой Т.В. Добудько

Начальник УОП



____ Н.А. Доманина

1. ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)

Цель изучения дисциплины: изучение методов и приемов разработки мобильных приложений, формирование навыков разработки прототипов и законченных приложений с использованием современных инструментальных средств

Задачи изучения дисциплины:

- формирование навыков и умений обучающихся в областях сбора и анализа требований к мобильным версиям и приложениям;
- дать обучающимся систематизированное представление о сфере мобильной разработки;
- способствовать формированию навыков создания и тестирования прототипов и несложных мобильных приложений для ОС Android

Область профессиональной деятельности: 01 Образование и наука

2. МЕСТО ДИСЦИПЛИНЫ (МОДУЛЯ) В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Цикл (раздел) ОП: Б1.О.08

2.1 Требования к предварительной подготовке обучающегося:

Содержание дисциплины базируется на материале:

Программирование, Веб-технологии

2.2 Дисциплины (модули) и практики, для которых освоение данной дисциплины (модуля) необходимо как предшествующее:

Подготовка к сдаче и сдача профессионального (демонстрационного) экзамена

3. КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)

УК-1. Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач

УК-1.1. Анализирует задачу, выделяя этапы ее решения, действия по решению задачи

Знает: этапы решения задач разработки мобильных приложений для фиджитал-спорта.

Умеет: анализировать задачу, выделяя её базовые составляющие; осуществлять декомпозицию задачи.

УК-1.2 Находит, критически анализирует и выбирает информацию, необходимую для решения поставленной задачи

Знает: средства версионного контроля, порядок обновления программных продуктов для мобильных устройств, публикации новых версий.

Умеет: осуществлять постановку задачи; анализировать условие и определять оптимальный метод решения поставленной задачи.

УК-1.3 Рассматривает различные варианты решения задачи, оценивает их преимущества и риски

Знает: особенности виртуального и физического тестирования мобильных приложений.

Умеет: оценивать приложение для мобильных устройств для фиджитал-спорта с позиций соответствия требованиям законодательства РФ и эргономичности, в том числе и для пользователей с ОВЗ.

ОПК-9 Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности

ОПК-9.1 Знает принципы функционирования, основные характеристики и возможности аппаратных и программных средств современных информационных технологий; тенденции развития сквозных цифровых технологий и профессионально значимые решения на их основе; этические и правовые нормы при работе с информацией

Знает: архитектуру мобильных операционных систем, жизненный цикл приложений, встроенные сенсоры и их технические параметры, а также возможности интегрированных сред разработки, кроссплатформенных фреймворков и специализированных SDK для создания фиджитал-активностей, совмещающих физическую активность и цифровое взаимодействие; этические и правовые нормы при работе с информацией в процессе разработки и эксплуатации мобильных фиджитал-приложений.

ОПК-9.2 Умеет применять современные информационно-коммуникационные технологии для решения профессиональных задач с учетом специфики предметной области; осуществлять выбор необходимых для осуществления профессиональной деятельности аппаратных и программных средств, мобильных приложений, средств сетевой коммуникации на основе стандартов и норм, принятых в профессиональной среде и с учетом

Умеет: применять современные информационные технологии для разработки мобильных приложений для фиджитал-спорта с учетом требований информационной безопасности.

4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

Код занятия	Наименование разделов и тем /вид занятия/	Семестр	Часов	Интеракт.
	Раздел 1. Разработка мобильных приложений для фиджитал-спорта			
1.1	Введение в фиджитал-спорт и архитектуру мобильных приложений. Обзор платформы Android и Kotlin /Лек/	7	2	0
1.2	Введение в фиджитал-спорт и архитектуру мобильных приложений. Обзор платформы Android и Kotlin /Лаб/	7	4	2
1.3	Введение в фиджитал-спорт и архитектуру мобильных приложений. Обзор платформы Android и Kotlin /Ср/	7	10	0
1.4	Основы Jetpack Compose. Композиция интерфейсов для отображения статистики и расписания турниров /Лек/	7	2	0
1.5	Основы Jetpack Compose. Композиция интерфейсов для отображения статистики и расписания турниров /Лаб/	7	4	2
1.6	Основы Jetpack Compose. Композиция интерфейсов для отображения статистики и расписания турниров /Ср/	7	10	0
1.7	Навигация и маршрутизация в приложении. Разработка экранов регистрации участников и судей /Лек/	7	2	0
1.8	Навигация и маршрутизация в приложении. Разработка экранов регистрации участников и судей /Лаб/	7	4	2
1.9	Навигация и маршрутизация в приложении. Разработка экранов регистрации участников и судей /Ср/	7	10	0
1.10	Работа с состоянием (State) и событиями (Event). Реализация интерактивных карточек матчей /Лек/	7	2	0
1.11	Работа с состоянием (State) и событиями (Event). Реализация интерактивных карточек матчей /Лаб/	7	4	2
1.12	Работа с состоянием (State) и событиями (Event). Реализация интерактивных карточек матчей /Ср/	7	10	0
1.13	Многопоточность и корутины. Загрузка данных о цифровых треках физической активности /Лек/	7	2	0
1.14	Многопоточность и корутины. Загрузка данных о цифровых треках физической активности /Лаб/	7	4	2
1.15	Многопоточность и корутины. Загрузка данных о цифровых треках физической активности /Ср/	7	10	0
1.16	Внедрение зависимостей (Dagger/Hilt). Подключение репозитория для обработки данных с датчиков /Лек/	7	2	0
1.17	Внедрение зависимостей (Dagger/Hilt). Подключение репозитория для обработки данных с датчиков /Лаб/	7	4	0
1.18	Внедрение зависимостей (Dagger/Hilt). Подключение репозитория для обработки данных с датчиков /Ср/	7	10	0
1.19	Работа с сетью (Retrofit). Получение актуальных результатов соревнований и рейтингов /Лек/	7	2	0
1.20	Работа с сетью (Retrofit). Получение актуальных результатов соревнований и рейтингов /Лаб/	7	4	0
1.21	Работа с сетью (Retrofit). Получение актуальных результатов соревнований и рейтингов /Ср/	7	10	0
1.22	Локальное хранилище (Room). Кэширование истории выступлений и офлайн-режим для арен /Лек/	7	2	0
1.23	Локальное хранилище (Room). Кэширование истории выступлений и офлайн-режим для арен /Лаб/	7	2	0

1.24	Локальное хранилище (Room). Кэширование истории выступлений и офлайн-режим для арен /Ср/	7	12	0
1.25	Тестирование и публикация. Написание UI-тестов для сценариев судейства и релиз приложения в Google Play /Лек/	7	2	0
1.26	Тестирование и публикация. Написание UI-тестов для сценариев судейства и релиз приложения в Google Play /Лаб/	7	2	0
1.27	Тестирование и публикация. Написание UI-тестов для сценариев судейства и релиз приложения в Google Play /Ср/	7	12	0

5. Оценочные и методические материалы по дисциплине (модулю)

5.1. Содержание аудиторной работы по дисциплине (модулю)

Лекция № 1 (2 часа)

Введение в фиджитал-спорт и архитектуру мобильных приложений. Обзор платформы Android и Kotlin

Вопросы.

- 1.Определение фиджитал-спорта. Роль мобильных приложений в гибридных соревнованиях (цифровой трекинг, дополненная реальность, онлайн-судейство)
- 2.Классификация мобильных приложений для фиджитал-спорта (сбор данных с датчиков, трансляции, управление турнирами, социальные взаимодействия)
- 3.Архитектура ОС Android. Компоненты приложения (Activity, Service, Broadcast Receiver, Content Provider). Жизненный цикл Android-приложения
- 4.Основы языка Kotlin: нулевая безопасность (null safety), функции расширения, лямбда-выражения, область видимости (scope functions)
- 5.Jetpack Compose как декларативный UI-фреймворк. Сравнение с XML-разметкой. Принцип реактивного программирования

Лекция № 2 (2 часа)

Основы Jetpack Compose. Композиция интерфейсов для отображения статистики и расписания турниров

Вопросы.

- 1.Концепция композиции в Jetpack Compose. Функции-композируемые (@Composable). Переиспользование UI-компонентов
- 2.Базовые элементы интерфейса: Text, Button, Image, TextField. Модификаторы (Modifier) и их цепочки
- 3.Основные компоновщики (Layouts): Column, Row, Box. Выравнивание, распределение веса и отступы
- 4.Работа со списками: LazyColumn и LazyRow. Оптимизация отрисовки больших наборов данных (расписание турниров, таблицы рейтингов)
- 5.Тематизация и стилизация. Material Design 3 в Compose. Создание кастомных цветовых схем для брендинга фиджитал-соревнований

Лекция № 3 (2 часа)

Навигация и маршрутизация в приложении. Разработка экранов регистрации участников и судей

Вопросы.

- 1.Библиотека Navigation Compose. Навигационный граф (NavHost). Роли стартового и конечного пунктов маршрута
- 2.Передача аргументов между экранами. Способы передачи данных через NavBackStackEntry и сохранение состояния при повороте экрана
- 3.Реализация глубоких ссылок (Deep Links) для быстрого доступа к конкретным турнирам и матчам
- 4.Проектирование форм регистрации. Валидация полей (имя, email, ID участника, контактные данные). Обработка ошибок ввода
- 5.BottomNavigationBar и Scaffold для организации нижней навигации по ключевым разделам приложения (турниры, профиль, рейтинги)

Лекция № 4 (2 часа)

Работа с состоянием (State) и событиями (Event). Реализация интерактивных карточек матчей

Вопросы.

- 1.Понятие состояния в Compose. Неизменяемость состояния (State<T>). Подъем состояния (State Hoisting) — принцип единственного источника истины
- 2.Классы ViewModel и их роль в управлении состоянием. Жизненный цикл ViewModel в контексте Compose-экранов
- 3.Обработка событий (Event) от пользователя. Колбэки и передача лямбд в композируемые функции. Разделение UI и бизнес-логики
- 4.Интерактивные элементы: нажатие на карточку матча для просмотра деталей, изменение статуса матча, смена счетов в реальном времени
- 5.Паттерн Unidirectional Data Flow (UDF) для фиджитал-приложений. Связка State → UI → Event → ViewModel → New State

Лекция № 5 (2 часа)

Многопоточность и корутины. Загрузка данных о цифровых треках физической активности

Вопросы.

1. Асинхронное программирование в Android. Проблемы блокировки UI-потока (ANR)
2. Корутины Kotlin: Scope, Context, Dispatchers (Main, IO, Default). Структурированная конкурентность (structured concurrency)
3. Запуск корутин в Compose (LaunchedEffect, rememberCoroutineScope). Отмена корутин при уничтожении экрана
4. Симуляция загрузки данных с датчиков физической активности (шаги, частота пульса, скорость, ускорение). Переключение состояний Loading / Success / Error
5. Обработка ошибок и повторные попытки (retry) при сбоях загрузки данных. Тайм-ауты и отмена долгих операций

Лекция № 6 (2 часа)

Внедрение зависимостей (Dagger/Hilt). Подключение репозитория для обработки данных с датчиков

Вопросы.

1. Принципы внедрения зависимостей (Dependency Injection). Проблема жесткой связанности кода и пути ее решения
2. Библиотека Dagger Hilt: модули, компоненты, скоупы. Интеграция Hilt с Android-компонентами (Application, Activity, ViewModel)
3. Создание модулей для датчиков устройства (акселерометр, гироскоп, шагомер). Получение системных сенсорных данных через SensorManager
4. Репозиторий как единый источник данных. Объединение данных с датчиков в поток (Flow) с использованием корутин
5. Тестирование слоя репозитория: создание фейковых (fake) реализаций датчиков для юнит-тестов без реального оборудования

Лекция № 7 (2 часа)

Работа с сетью (Retrofit). Получение актуальных результатов соревнований и рейтингов

Вопросы.

1. Сетевые протоколы в мобильных приложениях. RESTful API, принципы работы с сервером (GET, POST, PUT, DELETE)
2. Библиотека Retrofit: настройка клиента, создание интерфейсов API, аннотации (@GET, @POST, @Query, @Path)
3. Сериализация и десериализация данных. Библиотека Moshi или Gson для преобразования JSON в Kotlin-классы (data class)
4. Подключение интерцепторов (Interceptor) для логирования запросов и добавления заголовков (токен авторизации, версия API)
5. Обработка сетевых ошибок. Коды ответов HTTP (200, 404, 500). Отображение пользователю понятных сообщений об ошибках при загрузке рейтингов

Лекция № 8 (2 часа)

Локальное хранилище (Room). Кэширование истории выступлений и офлайн-режим для арен

Вопросы.

1. Локальные базы данных в Android. Библиотека Room: сущности (Entity), DAO (Data Access Object), База данных (Database)
2. Миграции базы данных. Добавление новых колонок и таблиц без потери существующих данных
3. Кэширование истории выступлений участников. Политики обновления кэша (устаревание данных, принудительное обновление)
4. Офлайн-режим для бронирования арен. Сохранение заявок на бронирование в локальной БД с синхронизацией при появлении сети
5. Работа с Flow и LiveData в Room. Автоматическое обновление UI при изменении данных в БД (реактивный подход)

Лекция № 9 (2 часа)

Тестирование и публикация. Написание UI-тестов для сценариев судейства и релиз приложения в Google Play

Вопросы.

1. Виды тестирования в Android: модульные (Unit), интеграционные, UI-тесты (Instrumentation). Инструменты: JUnit, Espresso, Compose UI Test
2. Написание UI-тестов для Jetpack Compose. Поиск компонентов (semantic nodes), имитация нажатий и ввода текста
3. Тестирование сценариев судейства: создание матча, добавление результата, изменение счета, проверка итогового статуса
4. Инструменты отладки и профилирования: Android Profiler, Layout Inspector, StrictMode для поиска проблем производительности
5. Подготовка релизной сборки. Подпись приложения (Keystore). Создание иконок, скриншотов и маркетингового описания. Публикация в Google Play Console (внутреннее, закрытое, открытое тестирование, продакшн)

Лабораторное занятие № 1 (2 часа)

Введение в фиджитал-спорт и архитектуру мобильных приложений. Обзор платформы Android и Kotlin

Вопросы и задания

Задание 1. Приведите примеры 3 мобильных приложений, которые можно использовать в фиджитал-футболе, волейболе, баскетболе

Задание 2. Для одного из этих приложений начертите жизненный цикл Android-приложения, используя любой онлайн-редактор (например, draw.io)

Задание 3. Напишите функцию на языке Kotlin для вычисления значения рейтинга команды по результатам матча (победа +3, ничья +1, поражение 0)

Лабораторное занятие № 2 (2 часа)

Введение в фиджитал-спорт и архитектуру мобильных приложений. Обзор платформы Android и Kotlin

Вопросы и задания

Задание 1. Установите Android Studio, создайте новый проект с пустым Compose-шаблоном. Настройте эмулятор с API 33+

Задание 2. Напишите Kotlin-программу, моделирующую регистрацию участника фиджитал-соревнования (класс Participant с полями id, name, sportType, rating)

Задание 3. Реализуйте функцию-расширение для List<Participant>, которая сортирует участников по рейтингу и возвращает топ-3

Лабораторное занятие № 3 (2 часа)

Основы Jetpack Compose. Композиция интерфейсов для отображения статистики и расписания турниров

Вопросы и задания

Задание 1. Создайте компонуюемую функцию TournamentCard, которая отображает карточку турнира (название, дата, место проведения, количество участников)

Задание 2. Реализуйте экран расписания турниров с использованием LazyColumn, в котором отображается не менее 10 карточек турниров с тестовыми данными

Задание 3. Примените модификаторы для настройки внешнего вида: закругленные углы, тень, отступы между карточками

Лабораторное занятие № 4 (2 часа)

Основы Jetpack Compose. Композиция интерфейсов для отображения статистики и расписания турниров

Вопросы и задания

Задание 1. Создайте экран статистики матча, содержащий Text для счета, Button для обновления счета, Column с информацией о командах

Задание 2. Реализуйте кастомную тему оформления в соответствии с брендингом фиджитал-турнира (задайте основные и акцентные цвета в Color.kt)

Задание 3. Разработайте компонент MatchProgressBar — индикатор прогресса матча в процентах (от 0 до 100) с использованием animateFloatAsState

Лабораторное занятие № 5 (2 часа)

Навигация и маршрутизация в приложении. Разработка экранов регистрации участников и судей

Вопросы и задания

Задание 1. Настройте навигационный граф с тремя экранами: экран регистрации, экран профиля, экран списка турниров

Задание 2. Реализуйте экран регистрации участника с полями ввода (имя, email, вид спорта) и кнопкой перехода на экран профиля с передачей введенных данных

Задание 3. Добавьте нижнюю навигацию (BottomNavigationBar) для переключения между экранами "Турниры", "Профиль", "Рейтинги"

Лабораторное занятие № 6 (2 часа)

Навигация и маршрутизация в приложении. Разработка экранов регистрации участников и судей

Вопросы и задания

Задание 1. Реализуйте экран регистрации судьи с валидацией полей (проверка email на корректность, обязательность заполнения всех полей)

Задание 2. Настройте глубокие ссылки (Deep Links) для перехода на конкретный матч по URL-схеме (например, app://match/123)

Задание 3. Создайте экран выбора роли (участник или судья) с последующей маршрутизацией на соответствующие формы регистрации

Лабораторное занятие № 7 (2 часа)

Работа с состоянием (State) и событиями (Event). Реализация интерактивных карточек матчей

Вопросы и задания

Задание 1. Создайте ViewModel для экрана матчей с полем state (список матчей) и методами обновления статуса матча (Start, Finish, Cancel)

Задание 2. Реализуйте интерактивную карточку матча, которая при нажатии разворачивается и показывает дополнительную информацию (состав команд, текущий счет)

Задание 3. Используйте подъем состояния (State Hoisting) для вынесения логики обработки нажатий из компоуемой функции в родительский компонент

Лабораторное занятие № 8 (2 часа)

Работа с состоянием (State) и событиями (Event). Реализация интерактивных карточек матчей

Вопросы и задания

Задание 1. Реализуйте экран Live-матча с отображением текущего счета и кнопками "+1" для каждой команды с использованием StateFlow во ViewModel

Задание 2. Настройте обработку события "завершить матч" с диалогом подтверждения (AlertDialog) и обновлением состояния карточки матча

Задание 3. Реализуйте таймер обратного отсчета до начала матча (CountdownTimer) с отображением в формате MM:SS, используя remember и LaunchedEffect

Лабораторное занятие № 9 (2 часа)

Многопоточность и корутины. Загрузка данных о цифровых треках физической активности

Вопросы и задания

Задание 1. Напишите корутину, которая эмулирует загрузку данных с датчика шагов (симуляция с задержкой 2 секунды) с использованием withContext(Dispatchers.IO)

Задание 2. Реализуйте экран с состоянием загрузки (ProgressBar), отображением успешных данных и обработкой ошибки с кнопкой повтора

Задание 3. Используйте LaunchedEffect для запуска корутины при входе на экран и отмены ее при покидании экрана

Лабораторное занятие № 10 (2 часа)

Многопоточность и корутины. Загрузка данных о цифровых треках физической активности

Вопросы и задания

Задание 1. Создайте симулятор пульса, который генерирует случайные значения от 60 до 180 в реальном времени с интервалом 1 секунда, используя Flow с оператором interval

Задание 2. Реализуйте сбор данных с нескольких датчиков параллельно (шаги + пульс) с использованием async/await и объедините результаты в один data class ActivityData

Задание 3. Обработайте ситуацию тайм-аута загрузки (withTimeoutOrNull) и отобразите пользователю уведомление "Превышено время ожидания данных"

Лабораторное занятие № 11 (2 часа)

Внедрение зависимостей (Dagger/Hilt). Подключение репозитория для обработки данных с датчиков

Вопросы и задания

Задание 1. Настройте Dagger Hilt в проекте (добавьте зависимости в build.gradle, создайте класс Application с аннотацией @HiltAndroidApp)

Задание 2. Создайте модуль SensorModule с предоставлением зависимостей для SensorManager и репозитория датчиков (@Provides, @Singleton)

Задание 3. Внедрите репозиторий датчиков во ViewModel с использованием @Inject и реализуйте метод для получения текущих показаний акселерометра

Лабораторное занятие № 12 (2 часа)

Внедрение зависимостей (Dagger/Hilt). Подключение репозитория для обработки данных с датчиков

Вопросы и задания

Задание 1. Создайте интерфейс SensorRepository и две его реализации: RealSensorRepository (работа с реальными датчиками) и FakeSensorRepository (генерация тестовых данных)

Задание 2. Настройте переключение между реализациями с помощью аннотаций @Qualifier для использования Fake-репозитория в режиме отладки

Задание 3. Напишите юнит-тест для ViewModel, используя FakeSensorRepository и проверяющий корректность обновления состояния при получении данных

Лабораторное занятие № 13 (2 часа)

Работа с сетью (Retrofit). Получение актуальных результатов соревнований и рейтингов

Вопросы и задания

Задание 1. Настройте Retrofit-клиент с базовым URL (например, <https://jsonplaceholder.typicode.com>) и создайте API-интерфейс для получения списка результатов соревнований

Задание 2. Создайте data-классы для десериализации JSON-ответа (MatchResult, Team, Score) и реализуйте запрос на получение данных с использованием корутин

Задание 3. Отобразите полученный список рейтингов на экране с помощью LazyColumn. Добавьте состояние ошибки и повторную загрузку по кнопке

Лабораторное занятие № 14 (2 часа)

Работа с сетью (Retrofit). Получение актуальных результатов соревнований и рейтингов

Вопросы и задания

Задание 1. Добавьте интерцептор (Interceptor) для логирования всех запросов и ответов в Logcat (используйте HttpLoggingInterceptor)

Задание 2. Реализуйте отправку данных о завершённом матче на сервер методом POST (модель FinishMatchRequest с полями matchId, scoreTeam1, scoreTeam2)

Задание 3. Настройте обработку HTTP-ошибок 404 (не найдено) и 500 (внутренняя ошибка сервера) с выводом соответствующих сообщений пользователю

Лабораторное занятие № 15 (2 часа)

Локальное хранилище (Room). Кэширование истории выступлений и офлайн-режим для арен

Вопросы и задания

Задание 1. Создайте сущность Room HistoryRecord с полями id, matchId, participantName, score, date. Напишите DAO с методами insert, getAll, deleteAll

Задание 2. Реализуйте сохранение истории выступлений в базу данных при завершении матча и отображение сохраненных данных на отдельном экране

Задание 3. Реализуйте офлайн-режим бронирования арен: сохраните заявку в локальную таблицу Booking с полями arenaId, date, time, status (PENDING, SYNCED)

Задание 4. Реализуйте экран "Мои бронирования", где отображаются все заявки из локальной БД с индикатором статуса синхронизации (иконка облака или галочка)

Лабораторное занятие № 16 (2 часа)

Тестирование и публикация. Написание UI-тестов для сценариев судейства и релиз приложения в Google Play

Вопросы и задания

Тестирование и защита итогового проекта

5.2. Содержание самостоятельной работы по дисциплине (модулю)

Содержание обязательной самостоятельной работы по дисциплине

№ п/п	Темы дисциплины	Содержание самостоятельной работы	Продукты деятельности
1	Введение в фиджитал-спорт и архитектуру мобильных приложений. Обзор платформы Android и Kotlin	Выполнение примера из документации Android Studio	Программный код
2	Основы Jetpack Compose. Композиция интерфейсов для отображения статистики и расписания турниров	Выполнение примера из документации Android Studio	Программный код
3	Навигация и маршрутизация в приложении. Разработка экранов регистрации участников и судей	Выполнение примера из документации Android Studio	Программный код
4	Работа с состоянием (State) и событиями (Event). Реализация интерактивных карточек матчей	Выполнение примера из документации Android Studio	Программный код
5	Многопоточность и корутины. Загрузка данных о цифровых треках физической активности	Выполнение примера из документации Android Studio	Программный код
6	Внедрение зависимостей (Dagger/Hilt). Подключение репозитория для обработки данных с датчиков	Выполнение примера из документации Android Studio	Программный код
7	Работа с сетью (Retrofit). Получение актуальных результатов соревнований и рейтингов	Выполнение примера из документации Android Studio	Программный код
8	Локальное хранилище (Room). Кэширование истории выступлений и офлайн-режим для арен	Выполнение примера из документации Android Studio	Программный код

Содержание самостоятельной работы по дисциплине на выбор студента

№ п/п	Темы дисциплины	Содержание самостоятельной работы	Продукты деятельности
1	Тестирование и публикация. Написание UI-тестов для сценариев судейства и релиз приложения в Google Play	Создание игрового приложения	.apk приложение для мобильного устройства

5.3.Образовательные технологии

При организации изучения дисциплины будут использованы следующие образовательные технологии: информационно-коммуникационные технологии, технология организации самостоятельной работы, технология

рефлексивного обучения, технология модульного обучения, технология игрового обучения, технологии групповой дискуссии, интерактивные технологии, технология проблемного обучения, технология организации учебно-исследовательской деятельности, технология проектного обучения, технология развития критического мышления.

5.4. Текущий контроль, промежуточный контроль и промежуточная аттестация

Балльно-рейтинговая карта дисциплины оформлена как приложение к рабочей программе дисциплины. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по дисциплине оформлен отдельным документом.

6. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ

Дата обращения к электронным учебным изданиям, электронным образовательным ресурсам, современным базам данных и информационным справочным системам совпадает с датой утверждения рабочей программы

6.1. Рекомендуемая литература

6.1.1. Основная литература

	Авторы, составители	Заглавие	Издательство, год
ЛП.1	Пирская, Л. В.	Разработка мобильных приложений в среде Android Studio: учебное пособие https://biblioclub.ru/index.php?page=book_red&id=598634	Таганрог: Южный федеральный университет, 2019.

6.1.2. Дополнительная литература

	Авторы, составители	Заглавие	Издательство, год
ЛП.2.1	Березовская, Ю.В., Юфрякова, О.А., Вологодина, В.Г. и др.	Введение в разработку приложений для ОС Android URL: http://biblioclub.ru/index.php?page=book&id=428937	Москва: Национальный Открытый Университет «ИНТУИТ», 2016

6.2 Перечень программного обеспечения

Acrobat reader DC;
 Dr.Web Desktop security, Dr.Web Server security;
 GIMP;
 Мой офис, сервисы Яндекс 360 для образования (почта, Яндекс Диск, Яндекс Телемост, Яндекс документы, Календарь);
 Многофункциональный сервис обмена информацией Мах;
 Архиватор 7-Zip;
 Браузер Яндекс для организаций.

6.3 Перечень информационных справочных систем, профессиональных баз данных

- ЭБС «Университетская библиотека онлайн»
 - Базы данных Springer eBooks
 - Единое содержание общего образования <https://edsoo.ru/>
 - Библиотека цифрового образовательного контента <https://моиуроки.рф/>

7. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

7.1	Наименование специального помещения: учебная аудитория для проведения практических занятий, групповых консультаций, индивидуальных консультаций, текущего контроля, промежуточной аттестации, Учебная аудитория. Оснащенность: Меловая доска-1шт., Комплект учебной мебели, ноутбук, проекционное оборудование (мультимедийный проектор и экран).
7.2	Наименование специального помещения: помещение для самостоятельной работы, Читальный зал. Оснащенность: ПК-4шт. с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду СГСПУ, Письменный стол-4 шт., Парта-2 шт.

8. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ (МОДУЛЯ)

Работа над теоретическим материалом происходит кратко, схематично, последовательно фиксировать основные положения, выводы, формулировки, обобщения; помечать важные мысли, выделять ключевые слова, термины. Проверка терминов, понятий с помощью энциклопедий, словарей, справочников с выписыванием толкований в тетрадь. Обозначить вопросы, термины, материал, который вызывает трудности, пометить и попытаться найти ответ в рекомендуемой литературе. Если самостоятельно не удастся разобраться в материале, необходимо сформулировать вопрос и задать преподавателю.

Проработка рабочей программы дисциплины, уделяя особое внимание целям и задачам, структуре и содержанию дисциплины. Конспектирование источников, подготовка ответов к контрольным вопросам, просмотр рекомендуемой литературы, работа с информационными источниками в разных форматах.

Также в процессе изучения дисциплины методические рекомендации могут быть изданы отдельным документом.

Балльно-рейтинговая карта дисциплины «Разработка мобильных приложений для фиджитал-спорта»

Курс 4 Семестр 7

Вид контроля		Минимальное количество баллов	Максимальное количество баллов
Наименование раздела			
Текущий контроль по разделу:			
1	Аудиторная работа	32	32
2	Самостоятельная работа (специальные обязательные формы)	24	24
3	Самостоятельная работа (специальные формы на выбор студента)	0	20
Контрольное мероприятие по разделу			
Промежуточный контроль		56	76
Промежуточная аттестация		0	24
Итого:		56	100

Виды контроля		Перечень или примеры заданий, критерии оценки и количество баллов	Темы для изучения и образовательные результаты
Текущий контроль по разделу «Разработка мобильных приложений для фиджитал-спорта»			
1	Аудиторная работа	Лабораторное занятие № 2 (2 часа) Введение в фиджитал-спорт и архитектуру мобильных приложений. Обзор платформы Android и Kotlin Вопросы и задания Задание 1. Установите Android Studio, создайте новый проект с пустым Compose-шаблоном. Настройте эмулятор с API 33+ Задание 2. Напишите Kotlin-программу, моделирующую регистрацию участника фиджитал-соревнования (класс Participant с полями id, name, sportType, rating) Задание 3. Реализуйте функцию-расширение для List<Participant>, которая сортирует участников по рейтингу и возвращает топ-3 Критерий оценивания: 1 балл – выполнена базовая часть лабораторной работы, 2 балла – выполнена базовая и дополнительная(индивидуальная) часть лабораторной работы. Итого – 16х2=32 баллов	Тема Введение в фиджитал-спорт и архитектуру мобильных приложений. Обзор платформы Android и Kotlin Основы Jetpack Compose. Композиция интерфейсов для отображения статистики и расписания турниров Навигация и маршрутизация в приложении. Разработка экранов регистрации участников и судей Работа с состоянием (State) и событиями (Event). Реализация интерактивных карточек матчей Многопоточность и корутины. Загрузка данных о цифровых треках физической активности Внедрение зависимостей (Dagger/Hilt). Подключение репозитория для обработки данных с датчиков

			Работа с сетью (Retrofit). Получение актуальных результатов соревнований и рейтингов Локальное хранилище (Room). Кэширование истории выступлений и офлайн-режим для арен Тестирование и публикация. Написание UI-тестов для сценариев судейства и релиз приложения в Google Play Образовательные результаты Знает: этапы решения задач разработки мобильных приложений для фиджитал-спорта. Умеет: анализировать задачу, выделяя её базовые составляющие; осуществлять декомпозицию задачи. Знает: средства версионного контроля, порядок обновления программных продуктов для мобильных устройств, публикации новых версий. Умеет: осуществлять постановку задачи; анализировать условие и определять оптимальный метод решения поставленной задачи. Знает: особенности виртуального и физического тестирования мобильных приложений. Умеет: оценивать приложение для мобильных устройств для фиджитал-спорта с позиций соответствия требованиям законодательства РФ и эргономичности, в том числе и для пользователей с ОВЗ. Знает: архитектуру мобильных операционных систем, жизненный цикл приложений, встроенные сенсоры и их технические параметры, а также возможности интегрированных сред разработки, кроссплатформенных фреймворков и специализированных SDK для создания фиджитал-активностей, совмещающих физическую активность и цифровое взаимодействие; этические и правовые нормы при работе с информацией в процессе
--	--	--	---

			разработки и эксплуатации мобильных фиджитал-приложений. Умеет: применять современные информационные технологии для разработки мобильных приложений для фиджитал-спорта с учетом требований информационной безопасности.
2	Самостоятельная работа (обязательные формы)	Выполнены учебные примеры по каждой теме. Пример запускается в виртуальной машине Интерфейс приложения соответствует образцу Структура проекта соответствует образцу Студент отвечает на вопросы по коду Студент может быстро изменить интерфейс мобильного приложения (шрифт-цвет-рисунок) Каждый критерий оценивается в 0-2 балла. Итого – 8х3=24 баллов	Тема Введение в фиджитал-спорт и архитектуру мобильных приложений. Обзор платформы Android и Kotlin Основы Jetpack Compose. Композиция интерфейсов для отображения статистики и расписания турниров Навигация и маршрутизация в приложении. Разработка экранов регистрации участников и судей Работа с состоянием (State) и событиями (Event). Реализация интерактивных карточек матчей Многопоточность и корутины. Загрузка данных о цифровых треках физической активности Внедрение зависимостей (Dagger/Hilt). Подключение репозитория для обработки данных с датчиков Работа с сетью (Retrofit). Получение актуальных результатов соревнований и рейтингов Локальное хранилище (Room). Кэширование истории выступлений и офлайн-режим для арен Образовательные результаты Знает: этапы решения задач разработки мобильных приложений для фиджитал-спорта.

			Умеет: анализировать задачу, выделяя её базовые составляющие; осуществлять декомпозицию задачи. Знает: средства версионного контроля, порядок обновления программных продуктов для мобильных устройств, публикации новых версий. Умеет: осуществлять постановку задачи; анализировать условие и определять оптимальный метод решения поставленной задачи. Знает: особенности виртуального и физического тестирования мобильных приложений. Умеет: оценивать приложение для мобильных устройств для фиджитал-спорта с позиций соответствия требованиям законодательства РФ и эргономичности, в том числе и для пользователей с ОВЗ. Знает: архитектуру мобильных операционных систем, жизненный цикл приложений, встроенные сенсоры и их технические параметры, а также возможности интегрированных сред разработки, кроссплатформенных фреймворков и специализированных SDK для создания фиджитал-активностей, совмещающих физическую активность и цифровое взаимодействие; этические и правовые нормы при работе с информацией в процессе разработки и эксплуатации мобильных фиджитал-приложений. Умеет: применять современные информационные технологии для разработки мобильных приложений для фиджитал-спорта с учетом требований информационной безопасности.
--	--	--	--

Направление подготовки: 44.03.05 Педагогическое образование (с двумя профилями подготовки)

Направленность (профиль): «Информатика» и «Фиджитал-спорт»

Рабочая программа дисциплины «Разработка мобильных приложений для фиджитал-спорта»

3	Самостоятельная работа (на выбор студента)	Прототип игрового мобильного приложения доведен до стадии готового программного продукта. Итого 20 баллов	Тема: Тестирование и публикация. Написание UI-тестов для сценариев судейства и релиз приложения в Google Play Образовательные результаты Знает: архитектуру мобильных операционных систем, жизненный цикл приложений, встроенные сенсоры и их технические параметры, а также возможности интегрированных сред разработки, кроссплатформенных фреймворков и специализированных SDK для создания фиджитал-активностей, совмещающих физическую активность и цифровое взаимодействие; этические и правовые нормы при работе с информацией в процессе разработки и эксплуатации мобильных фиджитал-приложений. Умеет: применять современные информационные технологии для разработки мобильных приложений для фиджитал-спорта с учетом требований информационной безопасности.
Контрольное мероприятие по разделу			
Промежуточный контроль (количество баллов)		0-24	