МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ГЕОДЕЗИИ И КАРТОГРАФИИ
(МИИГАиК)

Факультет геоинформатики и информационной безопасности
Кафедра информатики и геоинформационных технологий

**ДОКУМЕНТ ПОДПИСАН
ЭЛЕКТРОННОЙ ПОДПИСЬЮ**

Сертификат: 5BFEE1D286E9C95A0C0D2C611E434BD5
Владелец: Степанченко Алексей Леонидович
Действителен: с 23.09.2026 по 17.12.2027

«УТВЕРЖДАЮ»
Проректор
А. Л. Степанченко
15 мая 2026 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ) «Б1.В.06 ОСНОВЫ РАЗРАБОТКИ МОБИЛЬНЫХ ПРИЛОЖЕНИЙ»

Уровень высшего образования: бакалавриат

Направление подготовки: 09.03.03 Прикладная информатика

Направленность (профиль) подготовки: Разработка программного обеспечения для обработки и анализа пространственных данных

Квалификация (степень) выпускника: бакалавр

Форма обучения: очная

Год набора (приема на обучение): 2026

Срок получения образования: 4 года

Объем: в зачетных единицах: 4 з.е.
в академических часах: 144 ак.ч.

Москва, 2026

Разработчики:
Старший преподаватель кафедры Информатики и
геоинформационных технологий Стрельцов А. С.

Рабочая программа дисциплины (модуля) составлена в соответствии с требованиями ФГОС ВО по направлению подготовки 09.03.03 Прикладная информатика, утвержденного приказом Минобрнауки от 19.09.2017 № 922

Согласование и утверждение

№	Подразделение	Ответственное лицо	ФИО	Виза	Дата, протокол (при наличии)
1	факультет Геоинформатики и информационной безопасности	Председатель ученого совета факультета	Матерухин А. В.	Рассмотрено	05.11.2025, № 12-25

Содержание

1. Цель и задачи освоения дисциплины (модуля)

2. Планируемые результаты обучения по дисциплине (модулю), соотнесенные с планируемыми результатами освоения образовательной программы

3. Место дисциплины в структуре ОП

4. Объем дисциплины и виды учебной работы

5. Содержание дисциплины
5.1. Разделы, темы дисциплины и виды занятий
5.2. Содержание разделов, тем дисциплины и формы текущего контроля

6. Оценочные материалы текущего контроля

7. Оценочные материалы промежуточной аттестации

8. Порядок проведения промежуточной аттестации

9. Материально-техническое и учебно-методическое обеспечение дисциплины
9.1. Перечень основной и дополнительной учебной литературы
9.2. Профессиональные базы данных и ресурсы «Интернет», к которым обеспечивается доступ обучающихся
9.3. Программное обеспечение и информационно-справочные системы, используемые при осуществлении образовательного процесса по дисциплине
9.4. Оборудование и технические средства обучения

10. Методические указания по освоению дисциплины (модуля)

1. Цель и задачи освоения дисциплины (модуля)

Цель освоения дисциплины - формирование у обучающихся теоретических знаний и практических навыков в области проектирования, разработки и тестирования кроссплатформенных мобильных приложений с использованием фреймворка Flutter и языка программирования Dart, а также подготовка специалистов, способных разрабатывать требования к программному обеспечению и осуществлять его проектирование в соответствии с современными стандартами и методологиями.

Задачи изучения дисциплины:

- Изучение теоретических основ кроссплатформенной разработки мобильных приложений и особенностей фреймворка Flutter.;
- Формирование практических навыков проектирования архитектуры мобильных приложений с использованием паттернов MVC, MVVM, BLoC и Clean Architecture..;
- Освоение методов работы с пользовательским интерфейсом, навигацией, управлением состоянием и хранением данных в мобильных приложениях.;
- Приобретение навыков интеграции мобильных приложений с облачными сервисами (Supabase), работы с сетевыми запросами и внешними API.

2. Планируемые результаты обучения по дисциплине (модулю), соотнесенные с планируемыми результатами освоения образовательной программы

Компетенции, индикаторы и результаты обучения

ПК-2 Способен разрабатывать требования и осуществлять проектирование программного обеспечения

ПК-2.1 Формализует и анализирует высокоуровневые требования к программному обеспечению

Знать:

ПК-2.1/Зн4 Современные способы визуализации пространственных данных и методы их реализации

Уметь:

ПК-2.1/Ум1 Планировать процессы и этапы обработки(получения) данных фотографической, оптико-электронной и других видов аэрокосмических съемок для решения определенного круга производственных и научных задач

Владеть:

ПК-2.1/Нв2 Навыками проведения анализа требований к системам распознавания образов

ПК-2.2 Обосновывает выбор системных решений и управляет архитектурными компромиссами при проектировании программного обеспечения

Знать:

ПК-2.2/Зн3 Базовые понятия компьютерной графики

Уметь:

ПК-2.2/Ум2 Обосновывать выбор архитектурных решений при построении систем распознавания образов

Владеть:

ПК-2.2/Нв1 Методами сбора и оценки качества пространственной информации, имеющей значения для подготовки и реализации проектов в соответствующих сферах профессиональной деятельности

ПК-2.3 Разрабатывает детальные функциональные и логические спецификации программных компонентов, включая описание алгоритмов, структур данных и интерфейсов взаимодействия

Знать:

ПК-2.3/Зн2 Методы описания алгоритмов и основные принципы построения функциональных и логических спецификаций программного обеспечения

Уметь:

ПК-2.3/Ум2 Математически грамотно формулировать задачу распознавания, формализовать алгоритмы распознавания в виде пошаговых описаний

Владеть:

ПК-2.3/Нв3 Использовать графические инструменты для построения изображения

3. Место дисциплины в структуре ОП

Дисциплина (модуль) Б1.В.06 «Основы разработки мобильных приложений» относится к формируемой участниками образовательных отношений части образовательной программы и изучается в семестре(ах): 8.

В процессе изучения дисциплины студент готовится к решению типов задач профессиональной деятельности, предусмотренных ФГОС ВО и образовательной программой.

4. Объем дисциплины (модуля) и виды учебной работы

Период обучения	Общая трудоемкость (часы)	Общая трудоемкость (ЗЕТ)	Практические занятия (часы)	Самостоятельная работа (часы)	Промежуточная аттестация (часы)
8 семестр	144	4	32	112	Зачет
Всего	144	4	32	112	

5. Содержание дисциплины (модуля)

5.1. Разделы, темы дисциплины и виды занятий

(часы промежуточной аттестации не указываются)

Наименование раздела, темы	Всего	Практические занятия	Самостоятельная работа	Планируемые результаты обучения, соотнесенные с результатами освоения программы
Раздел 1. Введение в кроссплатформенную мобильную разработку	**15**	**3**	**12**	ПК-2.1 ПК-2.2 ПК-2.3

Тема 1.1. Обзор мобильных платформ и подходов к разработке	5	1	4	
Тема 1.2. Установка и настройка среды разработки	5	1	4	
Тема 1.3. Основы языка Dart	5	1	4	
Раздел 2. Построение пользовательского интерфейса	**20**	**4**	**16**	ПК-2.1 ПК-2.2 ПК-2.3
Тема 2.1. Виджеты Flutter	5	1	4	
Тема 2.2. Компоновка интерфейса	5	1	4	
Тема 2.3. Material Design и адаптивный интерфейс	5	1	4	
Тема 2.4. Работа с пользовательским вводом	5	1	4	
Раздел 3. Навигация и управление состоянием	**17**	**3**	**14**	ПК-2.1 ПК-2.2 ПК-2.3
Тема 3.1. Навигация между экранами	5	1	4	
Тема 3.2. Управление состоянием с Provider	6	1	5	
Тема 3.3. Архитектура BLoC	6	1	5	
Раздел 4. Работа с данными	**18**	**3**	**15**	ПК-2.1 ПК-2.2 ПК-2.3
Тема 4.1. Локальное хранение данных	6	1	5	
Тема 4.2. Сетевое взаимодействие	6	1	5	
Тема 4.3. Интеграция с Supabase	6	1	5	
Раздел 5. Архитектурные паттерны	**18**	**3**	**15**	ПК-2.1 ПК-2.2 ПК-2.3
Тема 5.1. Паттерн MVC	6	1	5	
Тема 5.2. Паттерн MVVM	6	1	5	
Тема 5.3. Clean Architecture	6	1	5	
Раздел 6. Работа с устройством	**21**	**6**	**15**	ПК-2.1 ПК-2.2 ПК-2.3
Тема 6.1. Геолокация и карты	7	2	5	
Тема 6.2. Камера и галерея	7	2	5	
Тема 6.3. Сенсоры и уведомления	7	2	5	
Раздел 7. Интеграция методов машинного обучения	**14**	**4**	**10**	ПК-2.1 ПК-2.2 ПК-2.3
Тема 7.1. TensorFlow Lite в Flutter	7	2	5	
Тема 7.2. Обработка изображений с ML	7	2	5	
Раздел 8. Тестирование и публикация	**21**	**6**	**15**	ПК-2.1 ПК-2.2 ПК-2.3
Тема 8.1. Тестирование приложений	7	2	5	
Тема 8.2. Отладка и профилирование	7	2	5	

Тема 8.3. Сборка и публикация	7	2	5	
Итого	**144**	**32**	**112**	

5.2. Содержание разделов, тем дисциплин и формы текущего контроля

Раздел 1. Введение в кроссплатформенную мобильную разработку

Тема 1.1. Обзор мобильных платформ и подходов к разработке

История развития мобильных операционных систем. Сравнение нативной и кроссплатформенной разработки. Обзор фреймворков: Flutter, React Native, Xamarin. Преимущества Flutter и архитектура фреймворка.

Текущий контроль

Вид (форма) контроля
Просмотр выполненных работ

Тема 1.2. Установка и настройка среды разработки

Установка Flutter SDK и Dart. Настройка Android Studio и VS Code. Создание эмуляторов Android. Структура Flutter-проекта. Запуск первого приложения.

Текущий контроль

Вид (форма) контроля
Просмотр выполненных работ

Тема 1.3. Основы языка Dart

Синтаксис Dart. Типы данных и переменные. Функции и классы. Null safety. Коллекции. Асинхронное программирование: Future, async/await, Stream.

Текущий контроль

Вид (форма) контроля
Просмотр выполненных работ

Раздел 2. Построение пользовательского интерфейса

Тема 2.1. Виджеты Flutter

Концепция виджетов. StatelessWidget и StatefulWidget. Жизненный цикл виджетов. Дерево виджетов и BuildContext. Ключи виджетов.

Текущий контроль

Вид (форма) контроля
Просмотр выполненных работ

Тема 2.2. Компоновка интерфейса

Базовые виджеты компоновки: Container, Row, Column, Stack. Expanded и Flexible. Padding и SizedBox. Scaffold и AppBar. SingleChildScrollView и ListView.

Текущий контроль

Вид (форма) контроля
Просмотр выполненных работ

Тема 2.3. Material Design и адаптивный интерфейс

Принципы Material Design. Темы и стили. MediaQuery и LayoutBuilder. Адаптация под разные размеры экранов. Ориентация устройства.

Текущий контроль

Вид (форма) контроля
Просмотр выполненных работ

Тема 2.4. Работа с пользовательским вводом

TextField и TextEditingController. Кнопки и обработка нажатий. Формы и валидация. GestureDetector и обработка жестов.

Текущий контроль

Вид (форма) контроля
Просмотр выполненных работ

Раздел 3. Навигация и управление состоянием

Тема 3.1. Навигация между экранами

Navigator и Route. Передача данных между экранами. Именованные маршруты. Возврат данных с экрана. Диалоговые окна и модальные листы.

Текущий контроль

Вид (форма) контроля
Просмотр выполненных работ

Тема 3.2. Управление состоянием с Provider

Проблема управления состоянием. Паттерн Provider. ChangeNotifier и Consumer. MultiProvider. Контекст и доступ к провайдерам.

Текущий контроль

Вид (форма) контроля
Просмотр выполненных работ

Тема 3.3. Архитектура BLoC

Концепция BLoC (Business Logic Component). Events и States. Пакеты bloc и flutter_bloc. BlocProvider и BlocBuilder. HydratedBloc для сохранения состояния.

Текущий контроль

Вид (форма) контроля
Просмотр выполненных работ

Раздел 4. Работа с данными

Тема 4.1. Локальное хранение данных

SharedPreferences для простых данных. Работа с файловой системой. Пакет path_provider. Сериализация JSON.

Текущий контроль

Вид (форма) контроля
Просмотр выполненных работ

Тема 4.2. Сетевое взаимодействие

HTTP-запросы с пакетом http. REST API. Обработка JSON-ответов. Обработка ошибок сети. Загрузка и кэширование изображений.

Текущий контроль

Вид (форма) контроля
Просмотр выполненных работ

Тема 4.3. Интеграция с Supabase

Обзор платформы Supabase. Настройка проекта Supabase. Аутентификация пользователей. Работа с базой данных PostgreSQL. Облачное хранилище файлов.

Текущий контроль

Вид (форма) контроля
Просмотр выполненных работ

Раздел 5. Архитектурные паттерны

Тема 5.1. Паттерн MVC

Принципы Model-View-Controller. Разделение ответственности. Реализация MVC в Flutter. Пакет mvc_pattern.

Текущий контроль

Вид (форма) контроля
Просмотр выполненных работ

Тема 5.2. Паттерн MVVM

Принципы Model-View-ViewModel. ViewModel и связывание данных. Реализация MVVM с Provider. Сравнение с MVC.

Текущий контроль

Вид (форма) контроля
Просмотр выполненных работ

Тема 5.3. Clean Architecture

Принципы SOLID. Слои Clean Architecture: Domain, Data, Presentation. Dependency Injection. Тестируемость архитектуры.

Текущий контроль

Вид (форма) контроля
Просмотр выполненных работ

Раздел 6. Работа с устройством

Тема 6.1. Геолокация и карты

Пакет geolocator. Получение координат. Обратное геокодирование. Разрешения на доступ к местоположению.

Текущий контроль

Вид (форма) контроля
Просмотр выполненных работ

Тема 6.2. Камера и галерея

Пакет image_picker. Выбор изображений из галереи. Съемка фото с камеры. Сохранение изображений.

Текущий контроль

Вид (форма) контроля
Просмотр выполненных работ

Тема 6.3. Сенсоры и уведомления

Работа с акселерометром и гироскопом. Локальные уведомления. Push-уведомления.

Текущий контроль

Вид (форма) контроля
Просмотр выполненных работ

Раздел 7. Интеграция методов машинного обучения

Тема 7.1. TensorFlow Lite в Flutter

Обзор TensorFlow Lite. Пакет tflite_flutter. Подготовка и конвертация моделей. Загрузка моделей в приложение.

Текущий контроль

Вид (форма) контроля
Просмотр выполненных работ

Тема 7.2. Обработка изображений с ML

Предобработка изображений для моделей. Классификация изображений. Стилизация с GAN. Постобработка результатов.

Текущий контроль

Вид (форма) контроля
Выполнение контрольных нормативов

Раздел 8. Тестирование и публикация

Тема 8.1. Тестирование приложений

Unit-тесты в Dart. Widget-тесты. Integration-тесты. Пакет flutter_test. Покрытие кода тестами.

Текущий контроль

Вид (форма) контроля

Просмотр выполненных работ

Тема 8.2. Отладка и профилирование

Flutter DevTools. Debug и Release режимы. Профилирование производительности. Анализ потребления памяти.

Текущий контроль

Вид (форма) контроля
Просмотр выполненных работ

Тема 8.3. Сборка и публикация

Настройка иконки и названия приложения. Сборка APK и App Bundle.

Текущий контроль

Вид (форма) контроля
Просмотр выполненных работ

6. Порядок проведения промежуточной аттестации

Порядок проведения промежуточной аттестации регламентируется локальными нормативными актами Университета:
- Положением о формах, периодичности, порядке текущего контроля успеваемости и промежуточной аттестации обучающихся по основным образовательным программам высшего образования – программам бакалавриата, программам специалитета, программам магистратуры в федеральном государственном бюджетном образовательном учреждении высшего образования «Московский государственный университет геодезии и картографии», п.6 «Организация и проведение промежуточной аттестации».
- Положением о порядке осуществления образовательного процесса, проведения текущего контроля успеваемости и промежуточной аттестации обучающихся с применением электронного обучения и дистанционных образовательных технологий в федеральном государственном бюджетном образовательном учреждении высшего образования «Московский государственный университет геодезии и картографии», п.8 «Проведение промежуточной аттестации с применением ЭО и ДОТ».

Промежуточная аттестация: Зачет, 8 семестр.
Зачёт по дисциплине проводится в формате тестирования для итоговой оценки знаний.

Студентам будет предложен тест, состоящий из 20 вопросов. На выполнение всей работы отводится шестьдесят минут, в течение которых запрещается использование любых вспомогательных материалов и электронных устройств.

Итоговый результат зачёта определяется в зависимости от процента правильных ответов. Для получения отметки «зачтено» необходимо верно ответить не менее чем на шестьдесят процентов вопросов, что соответствует 12 и более правильным ответам из 20. Результат ниже шестидесяти процентов правильных ответов, то есть 11 и менее верных ответов, признаётся неудовлетворительным и влечёт выставление отметки «не зачтено».

7. Оценочные материалы текущего контроля

Раздел 1. Введение в кроссплатформенную мобильную разработку
Контролируемые ИДК: ПК-2.1 ПК-2.2 ПК-2.3

Тема 1.1. Обзор мобильных платформ и подходов к разработке

Форма контроля/оценочное средство: Просмотр выполненных работ

Тема 1.2. Установка и настройка среды разработки
Форма контроля/оценочное средство: Просмотр выполненных работ

Тема 1.3. Основы языка Dart
Форма контроля/оценочное средство: Просмотр выполненных работ

Раздел 2. Построение пользовательского интерфейса
Контролируемые ИДК: ПК-2.1 ПК-2.2 ПК-2.3

Тема 2.1. Виджеты Flutter
Форма контроля/оценочное средство: Просмотр выполненных работ

Тема 2.2. Компоновка интерфейса
Форма контроля/оценочное средство: Просмотр выполненных работ

Тема 2.3. Material Design и адаптивный интерфейс
Форма контроля/оценочное средство: Просмотр выполненных работ

Тема 2.4. Работа с пользовательским вводом
Форма контроля/оценочное средство: Просмотр выполненных работ

Раздел 3. Навигация и управление состоянием
Контролируемые ИДК: ПК-2.1 ПК-2.2 ПК-2.3

Тема 3.1. Навигация между экранами
Форма контроля/оценочное средство: Просмотр выполненных работ

Тема 3.2. Управление состоянием с Provider
Форма контроля/оценочное средство: Просмотр выполненных работ

Тема 3.3. Архитектура BLoC
Форма контроля/оценочное средство: Просмотр выполненных работ

Раздел 4. Работа с данными
Контролируемые ИДК: ПК-2.1 ПК-2.2 ПК-2.3

Тема 4.1. Локальное хранение данных
Форма контроля/оценочное средство: Просмотр выполненных работ

Тема 4.2. Сетевое взаимодействие
Форма контроля/оценочное средство: Просмотр выполненных работ

Тема 4.3. Интеграция с Supabase
Форма контроля/оценочное средство: Просмотр выполненных работ

Раздел 5. Архитектурные паттерны
Контролируемые ИДК: ПК-2.1 ПК-2.2 ПК-2.3

Тема 5.1. Паттерн MVC

Форма контроля/оценочное средство: Просмотр выполненных работ

Тема 5.2. Паттерн MVVM
Форма контроля/оценочное средство: Просмотр выполненных работ

Тема 5.3. Clean Architecture
Форма контроля/оценочное средство: Просмотр выполненных работ

Раздел 6. Работа с устройством
Контролируемые ИДК: ПК-2.1 ПК-2.2 ПК-2.3

Тема 6.1. Геолокация и карты
Форма контроля/оценочное средство: Просмотр выполненных работ

Тема 6.2. Камера и галерея
Форма контроля/оценочное средство: Просмотр выполненных работ

Тема 6.3. Сенсоры и уведомления
Форма контроля/оценочное средство: Просмотр выполненных работ

Раздел 7. Интеграция методов машинного обучения
Контролируемые ИДК: ПК-2.1 ПК-2.2 ПК-2.3

Тема 7.1. TensorFlow Lite в Flutter
Форма контроля/оценочное средство: Просмотр выполненных работ

Тема 7.2. Обработка изображений с ML
Форма контроля/оценочное средство: Выполнение контрольных нормативов

Раздел 8. Тестирование и публикация
Контролируемые ИДК: ПК-2.1 ПК-2.2 ПК-2.3

Тема 8.1. Тестирование приложений
Форма контроля/оценочное средство: Просмотр выполненных работ

Тема 8.2. Отладка и профилирование
Форма контроля/оценочное средство: Просмотр выполненных работ

Тема 8.3. Сборка и публикация
Форма контроля/оценочное средство: Просмотр выполненных работ

8. Оценочные материалы промежуточной аттестации

8 семестр, Зачет
Контролируемые ИДК: ПК-2.1 ПК-2.2 ПК-2.3
Вопросы/Задания:

1. Какой язык программирования используется для разработки Flutter-приложений?

2. Как называется файл Flutter-проекта, содержащий список зависимостей (пакетов) проекта?

3. Как называется официальный набор инструментов для отладки и профилирования производительности Flutter-приложений, включающий анализ дерева виджетов и потребления памяти?

4. Дополните предложение термином:

Оператор _____ в Dart используется для безопасного доступа к значению nullable-переменной, возвращая null вместо возбуждения исключения, если переменная не инициализирована.

5. Дополните предложение термином:

Метод жизненного цикла StatefulWidget _____ вызывается первым, сразу при создании состояния виджета, и обычно используется для однократной инициализации.

6. Раскройте содержание концепции дерева виджетов (widget tree) и объекта BuildContext в Flutter: дайте определение, поясните их взаимосвязь.

7. Опишите проблему управления состоянием в Flutter-приложениях и раскройте содержание паттерна Provider как одного из способов её решения: дайте определение ChangeNotifier и Consumer, поясните их взаимодействие.

8. Опишите архитектуру BLoC (Business Logic Component): раскройте содержание понятий Event и State и поясните, как они связаны между собой в этой архитектуре.

9. Дайте определение архитектурного паттерна Clean Architecture, опишите его три основных слоя (Domain, Data, Presentation) и поясните роль внедрения зависимостей (Dependency Injection) в обеспечении тестируемости такой архитектуры.

10. Дайте определение процесса сборки Flutter-приложения в форматы APK и App Bundle, поясните различие режимов Debug и Release, а также раскройте назначение профилирования производительности приложения.

11. Расположите этапы отображения нового кадра пользовательского интерфейса Flutter после изменения состояния StatefulWidget в правильной последовательности.

1. Перестроение (повторный вызов метода build()) затронутой части дерева виджетов.
2. Событие, инициирующее изменение данных (например, нажатие пользователем кнопки).
3. Отрисовка (рендеринг) обновлённого дерева виджетов на экране устройства.
4. Вызов метода setState() с изменением значений внутри объекта состояния.
5. Планирование Flutter-движком перестроения затронутого поддерева виджетов на следующий кадр.

12. Расположите этапы навигации на новый экран с последующим возвратом данных на предыдущий экран в правильной последовательности.

1. Возврат данных на предыдущий экран вызовом Navigator.pop() с передаваемым значением.
2. Переход на новый экран вызовом Navigator.push().
3. Ожидание результата вызовом await на стороне предыдущего экрана и получение переданных данных.

13. Расположите этапы получения данных с внешнего REST API и их отображения в интерфейсе Flutter-приложения в правильной последовательности.

1. Разбор (декодирование) полученного JSON-ответа в объекты Dart.
2. Отправка HTTP-запроса к серверу с использованием пакета http.
3. Обновление состояния приложения полученными данными и перестроение интерфейса.
4. Получение ответа сервера с данными в формате JSON.

14. Расположите этапы подключения проекта к облачной платформе Supabase и аутентификации пользователя в правильной последовательности.

1. Отправка данных для входа (например, email и пароль) в Supabase через вызов метода

аутентификации.
2. Создание проекта в Supabase и получение ключей доступа (URL и API-ключа).
3. Проверка учётных данных на стороне Supabase и возврат токена сессии в случае успеха.
4. Инициализация клиента Supabase в Flutter-приложении с указанием полученных ключей.
5. Сохранение токена сессии на устройстве для последующих запросов от имени пользователя.
6. Отображение экрана авторизации (ввод email и пароля) пользователю приложения.

15. Расположите этапы получения фотографии с камеры устройства и её сохранения с использованием пакета image_picker в правильной последовательности.

1. Вызов метода pickImage() с указанием источника ImageSource.camera.
2. Запрос и получение разрешения на использование камеры устройства.
3. Получение объекта изображения (XFile) с путём к сделанному снимку.
4. Сохранение полученного изображения в постоянное хранилище приложения.

16. Установите соответствие между виджетом компоновки Flutter и его назначением.

Row - Заставляет дочерний виджет занять всё оставшееся доступное пространство внутри Row или Column
Column - Накладывает дочерние виджеты друг на друга (позиционирование поверх друг друга)
Stack - Располагает дочерние виджеты в один вертикальный столбец
Expanded - Располагает дочерние виджеты в один горизонтальный ряд

17. Установите соответствие между архитектурным паттерном и его ключевой особенностью.

MVC - ViewModel предоставляет представлению данные через механизм связывания (data binding), не будучи напрямую осведомлённым о конкретном представлении
MVVM - Приложение разделяется на независимые слои с направлением зависимостей от внешних слоёв к внутренним, обеспечивая независимость бизнес-логики от деталей реализации
Clean Architecture - Контроллер напрямую обрабатывает пользовательский ввод и обновляет модель, а представление напрямую обращается к модели для отображения данных

18. Установите соответствие между пакетом Flutter и его назначением.

Geolocator - Выполнение сетевых запросов к внешним серверам и API
image_picker - Получение текущих координат геолокации устройства
dio - Выбор изображений из галереи или съёмка фото с камеры устройства
shared_preferences - Загрузка и выполнение моделей машинного обучения в формате TensorFlow Lite
tflite_flutter - Хранение простых данных в формате «ключ – значение» на устройстве

19. Установите соответствие между видом тестирования Flutter-приложения (или проверки его работы) и его характеристикой.

Unit-тест - Измеряет расход вычислительных ресурсов (время построения кадра, потребление памяти) уже работающего приложения
Widget-тест - Проверяет работу приложения целиком в условиях, близких к реальным, на устройстве или эмуляторе
Integration-тест - Проверяет отдельную функцию или класс бизнес-логики в изоляции, без запуска интерфейса
Профилирование производительности - Проверяет отображение и поведение отдельного виджета в эмулированном дереве виджетов, без запуска на реальном устройстве

20. Установите соответствие между компонентом асинхронного программирования в Dart и его назначением

Future - Представляет последовательность асинхронных событий (значений), которые могут поступать многократно с течением времени
Stream - Приостанавливает выполнение асинхронной функции до завершения указанной операции, не блокируя при этом остальную часть приложения
await - Представляет результат асинхронной операции, который станет доступен один раз в

9. Материально-техническое и учебно-методическое обеспечение дисциплины

9.1. Перечень основной и дополнительной учебной литературы

Основная литература

1. Васильев Н. П. Введение в гибридные технологии разработки мобильных приложений: учебное пособие для вузов / Н. П. Васильев, А. М. Заяц. - 3-е изд., стер. - Санкт-Петербург: Лань, 2022. - 160 с. - 978-5-507-44502-8. - Текст: электронный // Издательство Лань: [сайт]. - URL: https://e.lanbook.com/book/230387 - Режим доступа: по подписке

Дополнительная литература

1. Петросян Л. Э. Разработка мобильных приложений на языке Kotlin: учебное пособие для вузов / Л. Э. Петросян, К. В. Гусев. - 2-е изд., стер. - Санкт-Петербург: Лань, 2025. - 104 с. - 978-5-507-52328-3. - Текст: электронный // Издательство Лань: [сайт]. - URL: https://e.lanbook.com/book/448577 - Режим доступа: по подписке

9.2. Профессиональные базы данных и ресурсы «Интернет», к которым обеспечивается доступ обучающихся

Профессиональные базы данных

Не используются.

Ресурсы «Интернет»

1. https://docs.flutter.dev/ - Официальная документация Flutter
2. https://sdo.miigaik.ru/ - Система дистанционного обучения МИИГАиК

9.3. Программное обеспечение и информационно-справочные системы, используемые при осуществлении образовательного процесса по дисциплине

Перечень программного обеспечения
(обновление производится по мере появления новых версий программы)

Не используется.

Перечень информационно-справочных систем
(обновление выполняется еженедельно)

Не используется.

9.4. Оборудование и технические средства обучения

Компьютерный класс

Автоматизированное рабочее место (АРМ) - 12 шт.
Интерактивная доска - 1 шт.
Проектор - 1 шт.

10. Методические указания по освоению дисциплины (модуля)

С целью успешного освоения данного предмета студенту необходимо выполнять домашние работы по указанным преподавателям темам и представленным на занятиях материалам. Для более успешного выполнения работы студенту предлагается использование учебно-методических материалов, соответствующих указанной теме.

Самостоятельная работа студентов описывается и регулируется методическими

рекомендациями по дисциплине.

Перечень вопросов для промежуточной аттестации:
1. Какой виджет Flutter используется для отображения прокручиваемого списка элементов?
2. Чем отличается StatelessWidget от StatefulWidget?
3. Какой метод вызывается для обновления UI в StatefulWidget?
4. Какой виджет используется для горизонтального расположения дочерних элементов?
5. Что возвращает асинхронная функция в Dart?
6. Какой пакет используется для управления состоянием по паттерну Provider?
7. Какой метод жизненного цикла StatefulWidget вызывается первым?
8. Какой виджет позволяет дочернему элементу занять все доступное пространство в Row или Column?
9. Какой класс используется для управления текстом в TextField?
10. Какой пакет используется для хранения простых данных (ключ-значение) на устройстве?
11. Какой метод Navigator используется для перехода на новый экран?
12. Что такое BuildContext в Flutter?
13. Какой виджет используется для создания базовой структуры экрана с AppBar?
14. Какой оператор используется для безопасного доступа к nullable переменной в Dart?
15. В архитектуре BLoC, что представляет собой Event?
16. Какой пакет используется для выполнения HTTP-запросов в Flutter?
17. Какой виджет используется для отображения изображения из сети?
18. Какой метод используется для закрытия текущего экрана и возврата данных?
19. Какой файл содержит зависимости Flutter-проекта?
20. Какой виджет позволяет прокручивать содержимое, если оно не помещается на экране?
21. Что делает ключевое слово async в Dart?
22. Какой компонент BLoC содержит бизнес-логику обработки событий?
23. Какой пакет используется для получения геолокации устройства?
24. Какой виджет используется для отображения индикатора загрузки?
25. Какой метод жизненного цикла вызывается при удалении StatefulWidget?
26. Какой паттерн управления состоянием основан на потоках событий и состояний?
27. Какой пакет используется для выбора изображений из галереи или камеры?
28. Какая команда используется для сборки APK-файла Flutter-приложения?
29. Какой виджет Flutter используется для обработки нажатий и жестов?
30. Какой язык программирования используется для разработки Flutter-приложений?

Описание возможностей изучения дисциплины лицами с ОВЗ и инвалидами

При необходимости обучения студентов-инвалидов и лиц с ограниченными возможностями здоровья аудиторные занятия могут быть заменены или дополнены изучением полнотекстовых лекций, презентаций, видео- и аудиоматериалов в электронной информационно-образовательной среде (ЭИОС) университета. Индивидуальные задания подбираются в адаптированных к ограничениям здоровья формах (письменно или устно, в форме презентаций). Выбор методов обучения зависит от их доступности для инвалидов и лиц с ограниченными возможностями здоровья.

Рабочая программа дисциплины для инвалидов и лиц с ограниченными возможностями здоровья разрабатывается индивидуально с учетом особенностей психофизического развития, индивидуальных возможностей и состояния здоровья таких обучающихся.