

**ОБЩЕСТВО С ОГРАНИЧЕННОЙ ОТВЕТСТВЕННОСТЬЮ
«ИННОПРОГ»**

УТВЕРЖДАЮ

Генеральный директор ООО

«ИННОПРОГ»



Венедиктов Р.В.

подпись

инициалы, фамилия

/ Венедиктов Р.В.

01 февраля 2025 г.

Приказ об утверждении
программы

от 01 февраля 2025 г. № ОБР-3

**ДОПОЛНИТЕЛЬНАЯ ПРОФЕССИОНАЛЬНАЯ ПРОГРАММА
ПРОФЕССИОНАЛЬНОЙ ПЕРЕПОДГОТОВКИ**

«Мобильный разработчик»

Вид программы:	дополнительная профессиональная программа профессиональной переподготовки
Новая квалификация:	Мобильный разработчик
Трудоемкость:	800 академических часов
Форма обучения:	очная, с применением электронного обучения и дистанционных образовательных технологий
Режим реализации:	10 месяцев (40 учебных недель), рекомендуемая нагрузка – 20 академических часов в неделю
Язык обучения:	русский

Иннополис, 2025

Содержание

1	Общая характеристика программы	5
1.1	Наименование программы	5
1.2	Нормативно-правовые основания разработки программы	5
1.3	Структура образовательной программы	6
1.4	Вид программы и целевая модель результата	7
1.5	Актуальность программы	7
1.6	Цель реализации программы	8
1.7	Задачи программы	8
1.8	Категория слушателей	9
1.9	Требования к уровню подготовки поступающего на обучение	9
1.10	Порядок приема, зачисления, отчисления, пересдачи и апелляции	10
1.11	Форма обучения, режим и язык обучения	10
1.12	Срок освоения и трудоемкость программы	11
1.13	Документ о квалификации	11
1.14	Характеристика новой квалификации и связанной с ней профессиональной деятельности	12
1.15	Связь Программы с профессиональными стандартами, трудовыми функциями и уровнями квалификации	13
1.16	Планируемые результаты освоения программы	14
1.16.1	Перечень формируемых профессиональных компетенций	14
1.16.2	Планируемые результаты обучения	15
2	Содержание программы	17
2.1	Учебный план	17
2.2	Календарный учебный график	17
2.3	Рабочие программы модулей	23
2.3.1	Модуль 1. Введение в профессию и организация обучения	26
2.3.2	Модуль 2. Основы Dart и алгоритмизация	28
2.3.3	Модуль 3. Объектно-ориентированное и асинхронное программирование на Dart	31
2.3.4	Модуль 4. Git, командная разработка и инструменты мобильного разработчика	34

2.3.5	Модуль 5. Основы SQL и моделирование данных мобильных приложений	36
2.3.6	Модуль 6. Flutter: основы интерфейса, виджеты и адаптивная верстка	39
2.3.7	Модуль 7. Навигация, состояние и архитектура Flutter-приложения	42
2.3.8	Модуль 8. Работа с API, JSON, HTTP и backend-интеграциями	45
2.3.9	Модуль 9. Локальное хранение данных и offline-first подход	47
2.3.10	Модуль 10. Платформенные возможности и нативные интеграции	50
2.3.11	Модуль 11. Тестирование, отладка, качество и безопасность мобильных приложений	53
2.3.12	Модуль 12. Сборка, публикация, аналитика и сопровождение приложений	55
2.3.13	Модуль 13. Проектная практика и консультации	58
2.3.14	Модуль 14. Итоговая аттестация	60
2.4	Проектная практика и консультации	62
3	Организационно-педагогические условия реализации программы	64
3.1	Общие условия реализации	64
3.2	Материально-технические условия	64
3.3	Электронная информационно-образовательная среда	64
3.4	Кадровые условия	65
3.5	Учебно-методическое обеспечение	65
3.6	Применение электронного обучения и дистанционных образовательных технологий	65
3.7	Информационная безопасность, персональные данные и этика .	66
3.8	Локальные процедуры обучения	66
3.9	Обновление программы	67
4	Оценка качества освоения программы	68
4.1	Общие положения	68
4.2	Текущий контроль успеваемости	68
4.3	Промежуточная аттестация	68

4.4	Пересдача, апелляция и подтверждение самостоятельности . . .	69
4.5	Допуск к итоговой аттестации	69
4.6	Итоговая аттестация	69
4.6.1	Примерная тематика проектных работ	70
4.6.2	Обязательные требования к итоговому проекту	71
4.6.3	Порядок проведения итоговой аттестации	72
4.6.4	Критерии оценивания итогового проекта	72
4.7	Внутренняя и внешняя оценка качества реализации программы	73
5	Методические материалы	74
5.1	Методические указания для слушателей	74
5.2	Методические указания для преподавателей и проверяющих . .	74
5.3	Рекомендуемая литература и ресурсы	74
	Приложение 1. Матрица формирования компетенций	75
	Приложение 2. Оценочные материалы промежуточной аттестации .	77
	Приложение 3. Оценочные материалы итоговой аттестации	79
	Приложение 4. Перечень локальных нормативных актов, на которые должна опираться реализация программы	82
	Приложение 5. Лист актуализации программы	83
	Приложение 6. Требования к репозиторию, README и демонстрации	84
	Приложение 7. Дорожная карта подготовки выпускного проекта . .	88
	Приложение 8. Контрольный лист соответствия требованиям про- граммы	94

1. Общая характеристика программы

1.1. Наименование программы

Дополнительная профессиональная программа профессиональной переподготовки «Мобильный разработчик» (далее – Программа) разработана для реализации обществом с ограниченной ответственностью «Иннопрог» в рамках дополнительного профессионального образования.

1.2. Нормативно-правовые основания разработки программы

Программа разработана в соответствии с:

- Федеральным законом от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации»;
- приказом Министерства науки и высшего образования Российской Федерации от 24.03.2025 № 266 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным профессиональным программам»;
- постановлением Правительства Российской Федерации от 11.10.2023 № 1678 «Об утверждении Правил применения организациями, осуществляющими образовательную деятельность, электронного обучения, дистанционных образовательных технологий при реализации образовательных программ»;
- постановлением Правительства Российской Федерации от 18.09.2020 № 1490 «О лицензировании образовательной деятельности»;
- постановлением Правительства Российской Федерации от 26.08.2013 № 729 «О федеральной информационной системе «Федеральный реестр сведений о документах об образовании и (или) о квалификации, документах об обучении»»;
- Федеральным законом от 27.07.2006 № 152-ФЗ «О персональных данных»;
- приказом Министерства труда и социальной защиты Российской Федерации от 20.07.2022 № 424н «Об утверждении профессионального стандарта «Программист»»;

- приказом Министерства труда и социальной защиты Российской Федерации от 18.01.2017 № 44н «Об утверждении профессионального стандарта «Разработчик Web и мультимедийных приложений»»;
- ГОСТ Р 7.0.97-2025 «Система стандартов по информации, библиотечно-му и издательскому делу. Организационно-распорядительная документация. Требования к оформлению документов»;
- локальными нормативными актами ООО «Иннопрог», регламентирующими разработку и утверждение образовательных программ, организацию образовательного процесса, применение электронного обучения и дистанционных образовательных технологий, текущий контроль, промежуточную и итоговую аттестацию, порядок выдачи документов о квалификации, защиту персональных данных и порядок оказания платных образовательных услуг.

При разработке содержания Программы также учтены требования федеральных государственных образовательных стандартов по укрупненной группе специальностей и направлений подготовки 09.00.00 «Информатика и вычислительная техника» в части результатов, связанных с разработкой программного обеспечения, базами данных, пользовательскими интерфейсами, проектной деятельностью и сопровождением информационных систем.

1.3. Структура образовательной программы

Настоящая Программа «Мобильный разработчик» оформлена как комплекс основных характеристик образования, организационно-педагогических условий, оценочных и методических материалов, необходимых для реализации дополнительной профессиональной программы профессиональной переподготовки.

В состав Программы включены:

- общая характеристика Программы, включая цель, задачи, категорию слушателей, требования к поступающим, форму обучения, срок освоения, трудоемкость, документ о квалификации и характеристику новой квалификации;

- учебный план, определяющий перечень модулей, трудоемкость, распределение часов по видам учебной работы, формы промежуточной и итоговой аттестации;
- календарный учебный график, определяющий рекомендуемую последовательность освоения модулей и период итоговой аттестации;
- рабочие программы модулей, раскрывающие цели, результаты, тематическое содержание, самостоятельную работу, проектные задания и формы контроля;
- организационно-педагогические условия реализации, включая кадровые, материально-технические, информационно-методические условия, электронную информационно-образовательную среду и порядок применения электронного обучения и дистанционных образовательных технологий;
- оценочные материалы для текущего контроля, промежуточной и итоговой аттестации, критерии допуска, пересдачи, апелляции и подтверждения самостоятельности выполнения работ;
- методические материалы и приложения, обеспечивающие единообразное применение Программы, проверку проектных работ, оформление результатов и актуализацию документа.

Структура Программы обеспечивает проверяемость объема, содержания, планируемых результатов, условий реализации, форм аттестации и документов, выдаваемых по итогам обучения.

1.4. Вид программы и целевая модель результата

Программа относится к дополнительным профессиональным программам профессиональной переподготовки и направлена на получение компетенции, необходимой для выполнения нового вида профессиональной деятельности, а также приобретение новой квалификации в сфере мобильной разработки на Dart и Flutter.

1.5. Актуальность программы

Программа обусловлена устойчивым спросом на специалистов, способных создавать мобильные приложения для Android и iOS, работать с пользовательскими сценариями, интерфейсом, состоянием приложения, локальными

ми данными, внешними API, платформенными возможностями устройств, тестированием, сборкой и сопровождением релизов. Кроссплатформенный подход на Flutter позволяет разрабатывать единый прикладной код для нескольких платформ и одновременно учитывать ограничения конкретной мобильной экосистемы. Практико-ориентированная структура программы обеспечивает переход от базового программирования на Dart к разработке полноценного мобильного приложения с документацией, тестами и демонстрацией результата.

1.6. Цель реализации программы

Цель реализации Программы – формирование у слушателя комплекса профессиональных компетенций, необходимых для выполнения нового вида профессиональной деятельности в сфере разработки мобильных приложений на Dart и Flutter, проектирования интерфейсов, управления состоянием, интеграции с API, работы с локальными данными, тестирования, сборки и подготовки приложения к выпуску.

1.7. Задачи программы

Задачами Программы являются:

- формирование устойчивой базы по языку Dart, алгоритмизации, коллекциям, функциям и обработке ошибок;
- освоение объектно-ориентированного и асинхронного программирования на Dart;
- формирование навыков проектирования пользовательских интерфейсов на Flutter;
- освоение навигации, управления состоянием и архитектурного разделения мобильного приложения;
- формирование навыков командной разработки с использованием Git, code review и технической документации;
- освоение SQL, локального хранения, кеширования, offline-first сценариев и синхронизации данных;
- приобретение навыков интеграции мобильного приложения с REST API, JSON, авторизацией и backend-сервисами;

- освоение платформенных возможностей мобильных устройств, включая разрешения, геолокацию, камеру, файлы, уведомления и deep links;
- формирование навыков тестирования, отладки, профилирования, анализа производительности и безопасности мобильных приложений;
- формирование навыков подготовки сборок, базового CI/CD, публикации, мониторинга ошибок и сопровождения релиза;
- развитие культуры инженерной разработки: документирование, тестирование, читаемость, структурированность и сопровождение кода.

1.8. Категория слушателей

К освоению Программы допускаются:

- лица, имеющие среднее профессиональное и (или) высшее образование;
- лица, получающие среднее профессиональное и (или) высшее образование.

1.9. Требования к уровню подготовки поступающего на обучение

Обязательным условием допуска к освоению Программы является наличие образования или получение образования, соответствующего требованиям законодательства к дополнительным профессиональным программам.

Рекомендуемый входной уровень подготовки:

- уверенное владение персональным компьютером и современными интернет-сервисами;
- базовая цифровая грамотность;
- готовность к регулярной самостоятельной работе, чтению документации и выполнению практических заданий;
- мотивация к освоению новой квалификации в сфере мобильной разработки;
- наличие компьютера, позволяющего установить Flutter SDK, среду разработки и эмулятор Android либо подключить физическое устройство для тестирования.

1.10. Порядок приема, зачисления, отчисления, пересдачи и апелляции

Прием на обучение осуществляется на основании заявления слушателя, документа, удостоверяющего личность, документа о среднем профессиональном и (или) высшем образовании либо справки об обучении для лица, получающего соответствующее образование, а также договора об образовании на обучение по дополнительной профессиональной программе. Перечень документов, порядок их представления, сроки рассмотрения заявления и основания отказа в приеме определяются локальными нормативными актами организации и договором об образовании.

Зачисление слушателя оформляется приказом организации после заключения договора, проверки входных документов и предоставления доступа к электронной информационно-образовательной среде. Отчисление, восстановление, перевод на индивидуальный график, изменение сроков обучения и прекращение образовательных отношений осуществляются в порядке, установленном локальными нормативными актами организации и договором об образовании.

Порядок пересдачи текущего контроля, промежуточной аттестации и итоговой аттестации, сроки устранения замечаний, основания допуска к повторной сдаче и порядок подачи апелляции определяются локальным актом о текущем контроле, промежуточной и итоговой аттестации. Слушателю обеспечивается возможность получить мотивированную обратную связь по результатам проверки и представить доработанную работу в установленные сроки.

1.11. Форма обучения, режим и язык обучения

Обучение по Программе осуществляется в очной форме с применением электронного обучения и дистанционных образовательных технологий. Электронное обучение и дистанционные образовательные технологии применяются при проведении учебных занятий, консультаций, самостоятельной работы, текущего контроля, промежуточной аттестации, проектной практики и итоговой аттестации при условии обеспечения доступа слушателя к электронной информационно-образовательной среде, учебным материалам, заданиям, средствам коммуникации и процедурам идентификации слушателя.

Язык обучения – русский. Для всех видов учебных занятий академический час устанавливается продолжительностью 45 минут. Образовательный процесс может осуществляться в течение всего календарного года. Конкретные даты начала и окончания обучения определяются приказом о зачислении, договором об образовании, расписанием и календарным учебным графиком.

Рекомендуемый режим освоения Программы – 20 академических часов в неделю в течение 10 месяцев (40 учебных недель). Допускается обучение по индивидуальному учебному плану в порядке, установленном локальным нормативным актом организации, при условии сохранения общей трудоемкости Программы и достижения планируемых результатов.

1.12. Срок освоения и трудоемкость программы

Трудоемкость Программы составляет 800 академических часов, включая все виды учебной работы слушателя, текущий контроль, промежуточную аттестацию и итоговую аттестацию.

Рекомендуемый срок освоения Программы составляет 10 месяцев (40 учебных недель). Допускается реализация Программы в ином календарном периоде при условии сохранения ее общей трудоемкости и достижения заявленных результатов обучения.

1.13. Документ о квалификации

Лицам, успешно освоившим Программу и прошедшим итоговую аттестацию, выдается диплом о профессиональной переподготовке установленного организацией образца.

Лицам, получающим среднее профессиональное и (или) высшее образование, диплом о профессиональной переподготовке выдается одновременно с получением соответствующего документа об образовании и о квалификации.

Лицам, не прошедшим итоговую аттестацию или получившим на итоговой аттестации неудовлетворительные результаты, а также лицам, освоившим часть Программы и (или) отчисленным из организации, выдается справка об обучении или о периоде обучения по образцу, самостоятельно установленному организацией.

Сведения о выданных дипломах о профессиональной переподготовке подлежат внесению в федеральную информационную систему «Федеральный реестр сведений о документах об образовании и (или) о квалификации, документах об обучении» (ФИС ФРДО) в случаях, составе и порядке, установленных законодательством Российской Федерации и локальным актом организации о выдаче и учете документов о квалификации.

1.14. Характеристика новой квалификации и связанной с ней профессиональной деятельности

Новая квалификация, приобретаемая по итогам освоения Программы:
Мобильный разработчик.

Область профессиональной деятельности выпускника:

- разработка, модификация, тестирование и сопровождение мобильных приложений;
- разработка кроссплатформенных приложений на Dart и Flutter;
- создание пользовательских интерфейсов и прикладной логики мобильных продуктов;
- интеграция приложений с API, локальными базами данных и платформенными возможностями устройств;
- участие в сборке, публикации, мониторинге и сопровождении мобильных релизов.

Объектами профессиональной деятельности выпускника являются:

- программный код и программные модули на Dart;
- Flutter-экраны, пользовательские сценарии, виджеты, состояния и навигация;
- локальные хранилища, API-интеграции, модели данных и сетевой слой;
- репозитории исходного кода, техническая документация, тесты, сборочные сценарии и релизные материалы;
- мобильные приложения для Android и iOS, их конфигурации, аналитика, журналы ошибок и пользовательская обратная связь.

Типовые виды профессиональной деятельности выпускника:

- разработка мобильных приложений на Flutter;

- проектирование интерфейса и пользовательских сценариев;
- работа с данными, API и локальным хранением;
- тестирование, отладка, профилирование и поддержка мобильных решений;
- подготовка сборок, публикация обновлений и сопровождение приложения после релиза;
- участие в командной разработке и выпуске версий программного продукта.

1.15. Связь Программы с профессиональными стандартами, трудовыми функциями и уровнями квалификации

Содержание Программы ориентировано на формирование компетенций для нового вида профессиональной деятельности в сфере разработки мобильного программного обеспечения. Основным профессиональным стандартом для соотнесения результатов обучения является профессиональный стандарт «Программист». Профессиональный стандарт «Разработчик Web и мультимедийных приложений» учитывается в части пользовательских интерфейсов, интерактивных компонентов, клиент-серверного взаимодействия и сопровождения цифровых продуктов.

Таблица 1 – Связь программы с профессиональными стандартами и трудовыми функциями

Профессиональный стандарт / блок	Трудовые функции, учитываемые при разработке Программы	Уровень квалификации	Модули и компетенции Программы
ПС «Программист», ОТФ А «Разработка и отладка программного кода»	А/01.3 – формализация и алгоритмизация задач; А/02.3 – написание программного кода; А/03.3 – оформление программного кода; А/04.3 – работа с системой управления версиями; А/05.3 – проверка и отладка кода.	3	Модули 2–5, 13–14; ПК-1, ПК-2, ПК-5, ПК-9, ПК-10.

Профессиональный стандарт / блок	Трудовые функции, учитываемые при разработке Программы	Уровень квалификации	Модули и компетенции Программы
ПС «Программист», ОТФ В «Проверка работоспособности и рефакторинг кода программного обеспечения»	В/01.4 – разработка процедур проверки; В/02.4 – разработка тестовых наборов данных; В/03.4 – проверка работоспособности; В/04.4 – рефакторинг, оптимизация и инспекция кода; В/06.4 – сборка программных модулей в проект.	4	Модули 6–12, 13–14; ПК-3, ПК-4, ПК-6, ПК-7, ПК-8, ПК-10.
ПС «Программист», ОТФ С «Интеграция программных модулей и компонентов и проверка работоспособности выпусков программного продукта»	С/01.5 – разработка процедур интеграции; С/02.5 – интеграция модулей и проверка работоспособности выпусков.	5	Модули 8–14; ПК-5, ПК-6, ПК-7, ПК-8, ПК-10.
ПС «Разработчик Web и мультимедийных приложений»	Разработка пользовательских интерфейсов, интерактивных компонентов, клиент-серверного взаимодействия, сценариев отображения данных и сопровождения цифровых продуктов.	4–5	Модули 6–8, 10–14; ПК-3, ПК-4, ПК-5, ПК-6, ПК-8, ПК-10.

1.16. Планируемые результаты освоения программы

1.16.1. Перечень формируемых профессиональных компетенций

Таблица 2 – Профессиональные компетенции выпускника

Код	Профессиональная компетенция
ПК-1	Анализировать требования к мобильному приложению, выделять пользовательские сценарии, ограничения платформ, критерии приемки и риски реализации.
ПК-2	Разрабатывать программный код на Dart, применять алгоритмические конструкции, коллекции, функции, объектно-ориентированный и асинхронный подход.
ПК-3	Проектировать и реализовывать пользовательские интерфейсы на Flutter, применять виджеты, адаптивную верстку, темы, формы, списки и анимации.
ПК-4	Проектировать архитектуру мобильного приложения, разделять слои интерфейса, состояния, бизнес-логики, данных и инфраструктурных интеграций.
ПК-5	Работать с данными мобильного приложения, включая SQL-моделирование, локальное хранение, JSON, HTTP, REST API, авторизацию и синхронизацию.
ПК-6	Интегрировать платформенные возможности Android и iOS, включая разрешения, камеру, файлы, геолокацию, уведомления, deep links и платформенные каналы.
ПК-7	Проверять качество мобильного приложения через отладку, логирование, unit-тесты, widget-тесты, интеграционные проверки, анализ производительности и безопасность.
ПК-8	Готовить мобильное приложение к сборке, релизу, публикации, сопровождению, сбору аналитики, мониторингу ошибок и выпуску обновлений.
ПК-9	Использовать Git, инструменты Flutter-разработчика, систему задач, код-ревью, документацию и правила командной разработки.
ПК-10	Представлять результат разработки, демонстрировать приложение, объяснять архитектурные решения, оформлять репозиторий и защищать итоговый проект.

1.16.2. Планируемые результаты обучения

Таблица 3 – Планируемые результаты обучения

Группа результатов	Содержание результата
Знать	синтаксис Dart, принципы null safety, объектно-ориентированное и асинхронное программирование; структуру Flutter-приложения; основы SQL, локального хранения и API; подходы к тестированию, сборке, публикации и сопровождению мобильных приложений.
Уметь	разрабатывать пользовательские интерфейсы на Flutter, проектировать навигацию и состояние, интегрировать приложение с API и локальными данными, использовать возможности устройства, выполнять тестирование, отладку, сборку и подготовку релиза.
Владеть навыками	ведения мобильного проекта в Git, оформления README, анализа ошибок, написания тестов, подготовки демонстрации, защиты архитектурных решений и сопровождения приложения после выпуска.
Иметь практический опыт	создания нескольких учебных проектов и итогового мобильного приложения с пользовательскими сценариями, локальным хранением, сетевой интеграцией, тестами, инструкцией запуска и защитой результата.

2. Содержание программы

2.1. Учебный план

Таблица 4 – Учебный план

№	Наименование модуля	Всего ч.	Лек. ч.	Прак. ч.	Сам. раб.	ПА ч.	ИА ч.	Форма контроля
1	Введение в профессию и организация обучения	12	6	3	3	0	0	входной контроль
2	Основы Dart и алгоритмизация	92	29	37	20	6	0	зачет
3	Объектно-ориентированное и асинхронное программирование на Dart	68	20	28	14	6	0	зачет
4	Git, командная разработка и инструменты мобильного разработчика	36	12	12	6	6	0	зачет
5	Основы SQL и моделирование данных мобильных приложений	48	15	18	9	6	0	зачет
6	Flutter: основы интерфейса, виджеты и адаптивная верстка	104	29	43	26	6	0	зачет
7	Навигация, состояние и архитектура Flutter-приложения	80	23	34	17	6	0	зачет
8	Работа с API, JSON, HTTP и backend-интеграциями	68	20	28	14	6	0	зачет
9	Локальное хранение данных и offline-first подход	56	17	22	11	6	0	зачет
10	Платформенные возможности и нативные интеграции	56	17	22	11	6	0	зачет
11	Тестирование, отладка, качество и безопасность мобильных приложений	56	17	20	14	5	0	зачет
12	Сборка, публикация, аналитика и сопровождение приложений	44	14	17	8	5	0	зачет
13	Проектная практика и консультации	44	5	28	11	0	0	зачет
14	Итоговая аттестация	36	0	0	0	0	36	защита проекта
	Итого	800	224	312	164	64	36	

2.2. Календарный учебный график

Таблица 5 – Календарный учебный график

Учеб. неделя	Разделы и виды учебной работы	Часы	Формат и контроль
1	Введение в профессию и организация обучения – 12 ч.; Основы Dart и алгоритмизация – 8 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС
2	Основы Dart и алгоритмизация – 20 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС
3	Основы Dart и алгоритмизация – 20 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС
4	Основы Dart и алгоритмизация – 20 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС
5	Основы Dart и алгоритмизация – 20 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС
6	Основы Dart и алгоритмизация – 4 ч.; Объектно-ориентированное и асинхронное программирование на Dart – 16 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС
7	Объектно-ориентированное и асинхронное программирование на Dart – 20 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС
8	Объектно-ориентированное и асинхронное программирование на Dart – 20 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС

Учеб. неделя	Разделы и виды учебной работы	Часы	Формат и контроль
9	Объектно-ориентированное и асинхронное программирование на Dart – 12 ч.; Git, командная разработка и инструменты мобильного разработчика – 8 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС
10	Git, командная разработка и инструменты мобильного разработчика – 20 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС
11	Git, командная разработка и инструменты мобильного разработчика – 8 ч.; Основы SQL и моделирование данных мобильных приложений – 12 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС
12	Основы SQL и моделирование данных мобильных приложений – 20 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС
13	Основы SQL и моделирование данных мобильных приложений – 16 ч.; Flutter: основы интерфейса, виджеты и адаптивная верстка – 4 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС
14	Flutter: основы интерфейса, виджеты и адаптивная верстка – 20 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС
15	Flutter: основы интерфейса, виджеты и адаптивная верстка – 20 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС

Учеб. неделя	Разделы и виды учебной работы	Часы	Формат и контроль
16	Flutter: основы интерфейса, виджеты и адаптивная верстка – 20 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС
17	Flutter: основы интерфейса, виджеты и адаптивная верстка – 20 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС
18	Flutter: основы интерфейса, виджеты и адаптивная верстка – 20 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС
19	Навигация, состояние и архитектура Flutter-приложения – 20 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС
20	Навигация, состояние и архитектура Flutter-приложения – 20 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС
21	Навигация, состояние и архитектура Flutter-приложения – 20 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС
22	Навигация, состояние и архитектура Flutter-приложения – 20 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС
23	Работа с API, JSON, HTTP и backend-интеграциями – 20 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС

Учеб. неделя	Разделы и виды учебной работы	Часы	Формат и контроль
24	Работа с API, JSON, HTTP и backend-интеграциями – 20 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС
25	Работа с API, JSON, HTTP и backend-интеграциями – 20 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС
26	Работа с API, JSON, HTTP и backend-интеграциями – 8 ч.; Локальное хранение данных и offline-first подход – 12 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС
27	Локальное хранение данных и offline-first подход – 20 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС
28	Локальное хранение данных и offline-first подход – 20 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС
29	Локальное хранение данных и offline-first подход – 4 ч.; Платформенные возможности и нативные интеграции – 16 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС
30	Платформенные возможности и нативные интеграции – 20 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС
31	Платформенные возможности и нативные интеграции – 20 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС

Учеб. неделя	Разделы и виды учебной работы	Часы	Формат и контроль
32	Тестирование, отладка, качество и безопасность мобильных приложений – 20 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС
33	Тестирование, отладка, качество и безопасность мобильных приложений – 20 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС
34	Тестирование, отладка, качество и безопасность мобильных приложений – 16 ч.; Сборка, публикация, аналитика и сопровождение приложений – 4 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС
35	Сборка, публикация, аналитика и сопровождение приложений – 20 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС
36	Сборка, публикация, аналитика и сопровождение приложений – 20 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС
37	Проектная практика и консультации – 20 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС
38	Проектная практика и консультации – 20 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС
39	Проектная практика и консультации – 4 ч.; Итоговая аттестация – 16 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС

Учеб. неделя	Разделы и виды учебной работы	Часы	Формат и контроль
40	Итоговая аттестация – 20 ч.	20	электронный курс, практические задания, консультации, проверка работ и фиксация результатов в ЭИОС

2.3. Рабочие программы модулей

В рабочих программах модулей определяются цели, результаты, тематическое содержание, формы организации занятий, самостоятельной работы и контроля. Перечень тем соотносится с фактической структурой учебных материалов и внутреннего контента направления «Мобильный разработчик».

Проектная траектория программы. Практическая часть Программы организована через последовательную разработку проектных работ. Проекты используются в текущем контроле, промежуточной аттестации, проектной практике и при подготовке итогового аттестационного проекта. Сложность работ повышается от простых Dart-программ и экранов Flutter к приложениям с навигацией, состоянием, локальным хранением, сетевыми интеграциями, тестами, сборкой и демонстрацией результата.

Таблица 6 – Проектная траектория программы

№	Проект	Содержание проектной работы	Связанные модули
1	Мобильный менеджер задач	приложение для создания, редактирования, фильтрации и закрытия задач с локальным хранением и удобным экраном списка.	связанные модули определяются индивидуальной траекторией, преимущественно модули 2–13
2	Трекер привычек и целей	приложение для ежедневных отметок, статистики прогресса, напоминаний и визуализации результатов пользователя.	связанные модули определяются индивидуальной траекторией, преимущественно модули 2–13

№	Проект	Содержание проектной работы	Связанные модули
3	Каталог товаров с избранным	мобильный каталог с карточками товаров, поиском, фильтрацией, избранным и сохранением выбранных позиций.	связанные модули определяются индивидуальной траекторией, преимущественно модули 2–13
4	Мобильное приложение прогноза погоды	приложение, получающее данные из внешнего API, отображающее текущую погоду, прогноз и состояние загрузки или ошибки.	связанные модули определяются индивидуальной траекторией, преимущественно модули 2–13
5	Приложение личных финансов	система учета доходов и расходов, категорий, периода, сводки и локального хранения финансовых записей.	связанные модули определяются индивидуальной траекторией, преимущественно модули 2–13
6	Приложение для бронирования столиков	мобильный интерфейс выбора даты, времени, количества гостей, контактных данных и статуса заявки на бронирование.	связанные модули определяются индивидуальной траекторией, преимущественно модули 2–13
7	Приложение записи на услуги	сервис для выбора специалиста, услуги, даты, времени, подтверждения записи и отображения истории посещений.	связанные модули определяются индивидуальной траекторией, преимущественно модули 2–13
8	Учебная CRM для малого бизнеса	приложение с клиентами, заказами, статусами, заметками, поиском и локальной базой данных.	связанные модули определяются индивидуальной траекторией, преимущественно модули 2–13
9	Приложение доставки с картой	приложение для отображения заказов, маршрутов, геолокации, статуса доставки и карточки заказа.	связанные модули определяются индивидуальной траекторией, преимущественно модули 2–13

№	Проект	Содержание проектной работы	Связанные модули
10	Мобильный интернет-магазин	приложение с каталогом, карточкой товара, корзиной, оформлением заказа, авторизацией и API-интеграцией.	связанные модули определяются индивидуальной траекторией, преимущественно модули 2–13
11	Новостное приложение	приложение для чтения ленты новостей из API, поиска, сохранения статей и offline-доступа к избранному.	связанные модули определяются индивидуальной траекторией, преимущественно модули 2–13
12	Фитнес-трекер тренировок	приложение для расписания тренировок, упражнений, подходов, прогресса и уведомлений.	связанные модули определяются индивидуальной траекторией, преимущественно модули 2–13
13	Приложение для заметок с синхронизацией	приложение для создания заметок, тегов, поиска, локального кеша и синхронизации с backend-сервисом.	связанные модули определяются индивидуальной траекторией, преимущественно модули 2–13
14	Приложение с авторизацией и личным кабинетом	мобильный проект с регистрацией, входом, защищенными экранами, профилем пользователя и обработкой ошибок API.	связанные модули определяются индивидуальной траекторией, преимущественно модули 2–13
15	Продакшн-версия Flutter-приложения	итоговый проект с архитектурой, локальным хранением, API, тестами, сборкой, релизной конфигурацией, README и демонстрационным сценарием.	связанные модули определяются индивидуальной траекторией, преимущественно модули 2–13

Слушатель может выполнять указанные проекты последовательно либо выбрать индивидуальную тему после согласования с преподавателем. Индивидуальная тема должна соответствовать результатам Программы, включать мо-

бильную разработку на Dart и Flutter, работу с данными, понятный пользовательский сценарий, репозиторий и материалы для проверки.

2.3.1. Модуль 1. Введение в профессию и организация обучения

Цель модуля: сформировать понимание целей программы, требований к результатам, роли мобильного разработчика и правил организации образовательного процесса.

Объем модуля: 12 академических часов, в том числе лекции – 6 ч., практические занятия – 3 ч., самостоятельная работа – 3 ч., контроль – 0 ч.

Планируемые результаты освоения модуля:

- понимать структуру программы и требования к результатам обучения;
- знать правила работы в электронной среде, порядок контроля и аттестации;
- выбрать предварительную проектную траекторию и подготовить репозиторий.

Таблица 7 – Тематический план модуля 1 «Введение в профессию и организация обучения»

№	Тема	Часы	Краткое содержание
1	Профессия мобильного разработчика и место Flutter в цифровых продуктах	3	лекционный материал, практикум, самостоятельная работа и контроль результата
2	Структура программы, контрольные точки и требования к проектам	3	лекционный материал, практикум, самостоятельная работа и контроль результата
3	Рабочая среда, учетные записи, репозиторий и коммуникация	3	лекционный материал, практикум, самостоятельная работа и контроль результата
4	Выбор индивидуальной проектной траектории	3	лекционный материал, практикум, самостоятельная работа и контроль результата

Содержание модуля: Модуль обеспечивает освоение ключевых инструментов и практик, закрепляемых через практические задания, проектные работы и проверку результата по установленным критериям.

Практическая подготовка включает:

- создание учебного репозитория и первичной структуры мобильного проекта;
- настройка среды разработки, Flutter SDK, эмулятора или физического устройства;
- заполнение индивидуальной карты обучения и выбор предварительной темы проекта.

Самостоятельная работа включает:

- изучение регламентов обучения и критериев оценивания;
- проверка доступа к электронной информационно-образовательной среде;
- подготовка списка вопросов по выбранной проектной траектории.

Проектное задание модуля:

- подготовить стартовый репозиторий с README, описанием целей обучения и выбранной проектной темой.

Чек-лист результата по модулю:

- результат размещен в репозитории и имеет осмысленную историю изменений;
- приложение или учебный фрагмент запускается по инструкции;
- код соответствует теме модуля и демонстрирует освоение заявленных результатов;
- в README отражены назначение работы, порядок запуска, проверенные сценарии и ограничения;
- слушатель может объяснить ключевые решения, ошибки и способы их устранения.

Промежуточная проверка результата. зачет проводится в форме проверки готовности среды разработки, репозитория и индивидуального плана обучения.

Минимальный уровень освоения предполагает, что слушатель выполняет обязательную часть задания, может запустить проект, объяснить основные решения и исправить типовые ошибки по замечаниям преподавателя. Повышенный уровень предполагает аккуратную архитектуру, обработку крайних случаев, наличие тестов, понятную документацию, расширенную функциональность и самостоятельное обоснование выбранных решений.

Типовые ошибки, учитываемые при проверке:

- не настроены Flutter SDK, Dart SDK или среда разработки;
- отсутствует доступ к репозиторию;
- не выбран предварительный проектный контур;
- слушатель не ознакомился с правилами аттестации.

2.3.2. Модуль 2. Основы Dart и алгоритмизация

Цель модуля: сформировать базовые навыки программирования на Dart, решения алгоритмических задач, работы с коллекциями, функциями и типовой логикой мобильных приложений.

Объем модуля: 92 академических часов, в том числе лекции – 29 ч., практические занятия – 37 ч., самостоятельная работа – 20 ч., контроль – 6 ч.

Планируемые результаты освоения модуля:

- писать корректный код на Dart с переменными, условиями, циклами, функциями и коллекциями;
- решать прикладные задачи обработки данных мобильного приложения;
- читать сообщения ошибок и выполнять базовую отладку.

Таблица 8 – Тематический план модуля 2 «Основы Dart и алгоритмизация»

№	Тема	Часы	Краткое содержание
1	Экосистема Dart, структура проекта и запуск программы	6	лекционный материал, практикум, самостоятельная работа и контроль результата
2	Переменные, типы данных, выражения и null safety	9	лекционный материал, практикум, самостоятельная работа и контроль результата
3	Условные конструкции и логические выражения	9	лекционный материал, практикум, самостоятельная работа и контроль результата
4	Циклы, перебор данных и базовые алгоритмы	9	лекционный материал, практикум, самостоятельная работа и контроль результата

№	Тема	Часы	Краткое содержание
5	Функции, параметры, возвращаемые значения и замыкания	9	лекционный материал, практикум, самостоятельная работа и контроль результата
6	Коллекции List, Set, Map и типовые операции	12	лекционный материал, практикум, самостоятельная работа и контроль результата
7	Строки, даты, перечисления и форматирование данных	7	лекционный материал, практикум, самостоятельная работа и контроль результата
8	Исключения и обработка ошибок ввода	7	лекционный материал, практикум, самостоятельная работа и контроль результата
9	Пакеты, pub.dev и организация зависимостей	6	лекционный материал, практикум, самостоятельная работа и контроль результата
10	Алгоритмический практикум на коллекциях	12	лекционный материал, практикум, самостоятельная работа и контроль результата
11	Мини-проект на Dart с консольным интерфейсом	6	лекционный материал, практикум, самостоятельная работа и контроль результата

Содержание модуля: В модуле последовательно изучаются базовые конструкции Dart и приемы алгоритмизации, необходимые для мобильной разработки. Слушатель осваивает типы данных, null safety, функции, коллекции, обработку ошибок, работу с пакетами и декомпозицию задач. Практическая часть строится вокруг небольших программ, которые моделируют обработку пользовательского ввода, списков, настроек, карточек товаров, задач и других данных будущего мобильного приложения.

Практическая подготовка включает:

- выполнение практикума по темам модуля 2;
- доработка учебного или итогового Flutter-проекта с применением изученных инструментов;

- оформление результата в репозитории с понятной историей изменений, скриншотами и кратким описанием.

Самостоятельная работа включает:

- повторение теоретического материала и разбор типовых ошибок;
- самостоятельное решение дополнительных задач и улучшение пользовательского сценария;
- подготовка фрагмента проекта к проверке и защите.

Проектное задание модуля:

- выполнить проектное задание по модулю 2, показать работоспособность приложения или его части и описать результат в README;
- зафиксировать ограничения решения, список проверенных сценариев и план доработки.

Чек-лист результата по модулю:

- результат размещен в репозитории и имеет осмысленную историю изменений;
- приложение или учебный фрагмент запускается по инструкции;
- код соответствует теме модуля и демонстрирует освоение заявленных результатов;
- в README отражены назначение работы, порядок запуска, проверенные сценарии и ограничения;
- слушатель может объяснить ключевые решения, ошибки и способы их устранения.

Промежуточная проверка результата. зачет проводится по результатам практического задания, проверки репозитория, краткого устного опроса и анализа качества выполненного кода.

Минимальный уровень освоения предполагает, что слушатель выполняет обязательную часть задания, может запустить проект, объяснить основные решения и исправить типовые ошибки по замечаниям преподавателя. Повышенный уровень предполагает аккуратную архитектуру, обработку крайних случаев, наличие тестов, понятную документацию, расширенную функциональность и самостоятельное обоснование выбранных решений.

Типовые ошибки, учитываемые при проверке:

- код не компилируется или приложение не запускается;
- решение не покрывает обязательные требования задания;
- отсутствует разделение ответственности между компонентами;
- не обработаны типовые ошибки пользователя и состояния загрузки;
- нет понятной инструкции запуска и демонстрационного сценария.

2.3.3. Модуль 3. Объектно-ориентированное и асинхронное программирование на Dart

Цель модуля: сформировать навыки объектно-ориентированного и асинхронного программирования на Dart, необходимые для разработки поддерживаемых Flutter-приложений.

Объем модуля: 68 академических часов, в том числе лекции – 20 ч., практические занятия – 28 ч., самостоятельная работа – 14 ч., контроль – 6 ч.

Планируемые результаты освоения модуля:

- проектировать классы, модели данных и сервисные компоненты на Dart;
- использовать Future, async, await и Streams в прикладных сценариях;
- подготавливать модели и бизнес-логику для Flutter-приложения.

Таблица 9 – Тематический план модуля 3 «Объектно-ориентированное и асинхронное программирование на Dart»

№	Тема	Часы	Краткое содержание
1	Классы, объекты, поля и методы	8	лекционный материал, практикум, самостоятельная работа и контроль результата
2	Конструкторы, именованные конструкторы и фабрики	7	лекционный материал, практикум, самостоятельная работа и контроль результата
3	Инкапсуляция, неизменяемые модели и copyWith	8	лекционный материал, практикум, самостоятельная работа и контроль результата
4	Наследование, интерфейсы, абстрактные классы и mixins	10	лекционный материал, практикум, самостоятельная работа и контроль результата

№	Тема	Часы	Краткое содержание
5	Обобщения, расширения и типобезопасность	7	лекционный материал, практикум, самостоятельная работа и контроль результата
6	Асинхронность: Future, async и await	10	лекционный материал, практикум, самостоятельная работа и контроль результата
7	Streams и реактивные сценарии	7	лекционный материал, практикум, самостоятельная работа и контроль результата
8	Модели данных, сериализация и подготовка к Flutter-проекту	5	лекционный материал, практикум, самостоятельная работа и контроль результата
9	Практикум объектной и асинхронной модели	6	лекционный материал, практикум, самостоятельная работа и контроль результата

Содержание модуля: Модуль переводит слушателя от отдельных функций к проектированию модели приложения. Рассматриваются классы, конструкторы, интерфейсы, mixins, расширения, обобщения, асинхронные операции и потоки данных. В центре внимания находится способность описывать доменные сущности, сервисы, состояния и модели, которые затем используются во Flutter-интерфейсе.

Практическая подготовка включает:

- выполнение практикума по темам модуля 3;
- доработка учебного или итогового Flutter-проекта с применением изученных инструментов;
- оформление результата в репозитории с понятной историей изменений, скриншотами и кратким описанием.

Самостоятельная работа включает:

- повторение теоретического материала и разбор типовых ошибок;
- самостоятельное решение дополнительных задач и улучшение пользовательского сценария;

- подготовка фрагмента проекта к проверке и защите.

Проектное задание модуля:

- выполнить проектное задание по модулю 3, показать работоспособность приложения или его части и описать результат в README;
- зафиксировать ограничения решения, список проверенных сценариев и план доработки.

Чек-лист результата по модулю:

- результат размещен в репозитории и имеет осмысленную историю изменений;
- приложение или учебный фрагмент запускается по инструкции;
- код соответствует теме модуля и демонстрирует освоение заявленных результатов;
- в README отражены назначение работы, порядок запуска, проверенные сценарии и ограничения;
- слушатель может объяснить ключевые решения, ошибки и способы их устранения.

Промежуточная проверка результата. зачет проводится по результатам практического задания, проверки репозитория, краткого устного опроса и анализа качества выполненного кода.

Минимальный уровень освоения предполагает, что слушатель выполняет обязательную часть задания, может запустить проект, объяснить основные решения и исправить типовые ошибки по замечаниям преподавателя. Повышенный уровень предполагает аккуратную архитектуру, обработку крайних случаев, наличие тестов, понятную документацию, расширенную функциональность и самостоятельное обоснование выбранных решений.

Типовые ошибки, учитываемые при проверке:

- код не компилируется или приложение не запускается;
- решение не покрывает обязательные требования задания;
- отсутствует разделение ответственности между компонентами;
- не обработаны типовые ошибки пользователя и состояния загрузки;
- нет понятной инструкции запуска и демонстрационного сценария.

2.3.4. Модуль 4. Git, командная разработка и инструменты мобильного разработчика

Цель модуля: сформировать инженерную культуру мобильного разработчика, включая Git, инструменты Flutter, проверку кода, ведение задач и оформление результатов.

Объем модуля: 36 академических часов, в том числе лекции – 12 ч., практические занятия – 12 ч., самостоятельная работа – 6 ч., контроль – 6 ч.

Планируемые результаты освоения модуля:

- вести проект в Git и оформлять историю изменений;
- использовать Flutter CLI, Dart analyzer, форматирование и lint-правила;
- оформлять README и участвовать в code review.

Таблица 10 – Тематический план модуля 4 «Git, командная разработка и инструменты мобильного разработчика»

№	Тема	Часы	Краткое содержание
1	Git: репозиторий, коммиты, ветки и история изменений	6	лекционный материал, практикум, самостоятельная работа и контроль результата
2	Pull request, code review и командная работа	6	лекционный материал, практикум, самостоятельная работа и контроль результата
3	Flutter CLI, Dart analyzer, форматирование и lint-правила	6	лекционный материал, практикум, самостоятельная работа и контроль результата
4	Среда разработки, эмуляторы, устройства и диагностика	6	лекционный материал, практикум, самостоятельная работа и контроль результата
5	Issue tracker, декомпозиция задач и планирование спринта	6	лекционный материал, практикум, самостоятельная работа и контроль результата
6	README, changelog и техническая документация проекта	6	лекционный материал, практикум, самостоятельная работа и контроль результата

Содержание модуля: Модуль формирует единый инженерный порядок работы с мобильным проектом. Слушатель настраивает репозиторий, ведет ветки, оформляет коммиты, использует инструменты анализа и форматирования Dart-кода, готовит README и осваивает практику ревью. Результаты модуля применяются во всех последующих практических работах.

Практическая подготовка включает:

- выполнение практикума по темам модуля 4;
- доработка учебного или итогового Flutter-проекта с применением изученных инструментов;
- оформление результата в репозитории с понятной историей изменений, скриншотами и кратким описанием.

Самостоятельная работа включает:

- повторение теоретического материала и разбор типовых ошибок;
- самостоятельное решение дополнительных задач и улучшение пользовательского сценария;
- подготовка фрагмента проекта к проверке и защите.

Проектное задание модуля:

- выполнить проектное задание по модулю 4, показать работоспособность приложения или его части и описать результат в README;
- зафиксировать ограничения решения, список проверенных сценариев и план доработки.

Чек-лист результата по модулю:

- результат размещен в репозитории и имеет осмысленную историю изменений;
- приложение или учебный фрагмент запускается по инструкции;
- код соответствует теме модуля и демонстрирует освоение заявленных результатов;
- в README отражены назначение работы, порядок запуска, проверенные сценарии и ограничения;
- слушатель может объяснить ключевые решения, ошибки и способы их устранения.

Промежуточная проверка результата. зачет проводится по результатам практического задания, проверки репозитория, краткого устного опроса и анализа качества выполненного кода.

Минимальный уровень освоения предполагает, что слушатель выполняет обязательную часть задания, может запустить проект, объяснить основные решения и исправить типовые ошибки по замечаниям преподавателя. Повышенный уровень предполагает аккуратную архитектуру, обработку крайних случаев, наличие тестов, понятную документацию, расширенную функциональность и самостоятельное обоснование выбранных решений.

Типовые ошибки, учитываемые при проверке:

- код не компилируется или приложение не запускается;
- решение не покрывает обязательные требования задания;
- отсутствует разделение ответственности между компонентами;
- не обработаны типовые ошибки пользователя и состояния загрузки;
- нет понятной инструкции запуска и демонстрационного сценария.

2.3.5. Модуль 5. Основы SQL и моделирование данных мобильных приложений

Цель модуля: сформировать понимание реляционных данных, SQL и принципов моделирования данных, применимых к локальному и серверному хранению мобильного приложения.

Объем модуля: 48 академических часов, в том числе лекции – 15 ч., практические занятия – 18 ч., самостоятельная работа – 9 ч., контроль – 6 ч.

Планируемые результаты освоения модуля:

- проектировать простые структуры данных и связи;
- писать SQL-запросы и понимать принципы локального и серверного хранения;
- составлять модель данных для мобильного приложения.

Таблица 11 – Тематический план модуля 5 «Основы SQL и моделирование данных мобильных приложений»

№	Тема	Часы	Краткое содержание
1	Реляционная модель и назначение SQL в мобильной разработке	6	лекционный материал, практикум, самостоятельная работа и контроль результата
2	SELECT, WHERE, ORDER BY, LIMIT и базовая выборка данных	6	лекционный материал, практикум, самостоятельная работа и контроль результата
3	JOIN, связи таблиц и нормализация на учебном уровне	6	лекционный материал, практикум, самостоятельная работа и контроль результата
4	Группировка, агрегаты и аналитические выборки	6	лекционный материал, практикум, самостоятельная работа и контроль результата
5	INSERT, UPDATE, DELETE и транзакционная аккуратность	6	лекционный материал, практикум, самостоятельная работа и контроль результата
6	Моделирование данных мобильного приложения	6	лекционный материал, практикум, самостоятельная работа и контроль результата
7	SQLite, PostgreSQL и различия локального и серверного хранения	6	лекционный материал, практикум, самостоятельная работа и контроль результата
8	Практикум проектирования схемы данных	6	лекционный материал, практикум, самостоятельная работа и контроль результата

Содержание модуля: Модуль объясняет, как мобильное приложение хранит, получает и структурирует данные. SQL изучается как инструмент понимания таблиц, связей, фильтрации и агрегирования. Отдельное внимание уделяется различиям между локальной базой на устройстве, серверной базой данных и API, через который мобильный клиент получает данные.

Практическая подготовка включает:

- выполнение практикума по темам модуля 5;

- доработка учебного или итогового Flutter-проекта с применением изученных инструментов;
- оформление результата в репозитории с понятной историей изменений, скриншотами и кратким описанием.

Самостоятельная работа включает:

- повторение теоретического материала и разбор типовых ошибок;
- самостоятельное решение дополнительных задач и улучшение пользовательского сценария;
- подготовка фрагмента проекта к проверке и защите.

Проектное задание модуля:

- выполнить проектное задание по модулю 5, показать работоспособность приложения или его части и описать результат в README;
- зафиксировать ограничения решения, список проверенных сценариев и план доработки.

Чек-лист результата по модулю:

- результат размещен в репозитории и имеет осмысленную историю изменений;
- приложение или учебный фрагмент запускается по инструкции;
- код соответствует теме модуля и демонстрирует освоение заявленных результатов;
- в README отражены назначение работы, порядок запуска, проверенные сценарии и ограничения;
- слушатель может объяснить ключевые решения, ошибки и способы их устранения.

Промежуточная проверка результата. зачет проводится по результатам практического задания, проверки репозитория, краткого устного опроса и анализа качества выполненного кода.

Минимальный уровень освоения предполагает, что слушатель выполняет обязательную часть задания, может запустить проект, объяснить основные решения и исправить типовые ошибки по замечаниям преподавателя. Повышенный уровень предполагает аккуратную архитектуру, обработку крайних случа-

ев, наличие тестов, понятную документацию, расширенную функциональность и самостоятельное обоснование выбранных решений.

Типовые ошибки, учитываемые при проверке:

- код не компилируется или приложение не запускается;
- решение не покрывает обязательные требования задания;
- отсутствует разделение ответственности между компонентами;
- не обработаны типовые ошибки пользователя и состояния загрузки;
- нет понятной инструкции запуска и демонстрационного сценария.

2.3.6. Модуль 6. Flutter: основы интерфейса, виджеты и адаптивная верстка

Цель модуля: сформировать навыки разработки пользовательского интерфейса Flutter-приложения с учетом адаптивности, доступности, валидации и удобства пользователя.

Объем модуля: 104 академических часов, в том числе лекции – 29 ч., практические занятия – 43 ч., самостоятельная работа – 26 ч., контроль – 6 ч.

Планируемые результаты освоения модуля:

- создавать адаптивные пользовательские интерфейсы на Flutter;
- использовать виджеты, формы, списки, темы, ассеты и анимации;
- проверять интерфейс на разных размерах экранов.

Таблица 12 – Тематический план модуля 6 «Flutter: основы интерфейса, виджеты и адаптивная верстка»

№	Тема	Часы	Краткое содержание
1	Структура Flutter-приложения и жизненный цикл запуска	6	лекционный материал, практикум, самостоятельная работа и контроль результата
2	Дерево виджетов, StatelessWidget и StatefulWidget	9	лекционный материал, практикум, самостоятельная работа и контроль результата
3	Layout: Row, Column, Stack, Flex, Expanded и constraints	11	лекционный материал, практикум, самостоятельная работа и контроль результата

№	Тема	Часы	Краткое содержание
4	Material и Cupertino компоненты	9	лекционный материал, практикум, самостоятельная работа и контроль результата
5	Формы, ввод данных, валидация и сообщения об ошибках	11	лекционный материал, практикум, самостоятельная работа и контроль результата
6	Списки, сетки, прокрутка и отображение больших наборов данных	9	лекционный материал, практикум, самостоятельная работа и контроль результата
7	Темы, цвета, шрифты, изображения и ассеты	9	лекционный материал, практикум, самостоятельная работа и контроль результата
8	Адаптивная верстка под разные размеры экранов	9	лекционный материал, практикум, самостоятельная работа и контроль результата
9	Доступность, локализация и базовые требования к интерфейсу	6	лекционный материал, практикум, самостоятельная работа и контроль результата
10	Анимации и интерактивные состояния	8	лекционный материал, практикум, самостоятельная работа и контроль результата
11	UI-проект с несколькими экранами	11	лекционный материал, практикум, самостоятельная работа и контроль результата
12	Hot reload, инспектор виджетов и диагностика интерфейса	6	лекционный материал, практикум, самостоятельная работа и контроль результата

Содержание модуля: Модуль является основой визуальной части программы. Слушатель изучает дерево виджетов, layout, формы, списки, темы, ассеты, адаптивность, доступность, локализацию и анимации. Практические задания направлены на создание нескольких экранов, обработку ввода и реализацию понятных пользовательских состояний.

Практическая подготовка включает:

- выполнение практикума по темам модуля 6;
- доработка учебного или итогового Flutter-проекта с применением изученных инструментов;
- оформление результата в репозитории с понятной историей изменений, скриншотами и кратким описанием.

Самостоятельная работа включает:

- повторение теоретического материала и разбор типовых ошибок;
- самостоятельное решение дополнительных задач и улучшение пользовательского сценария;
- подготовка фрагмента проекта к проверке и защите.

Проектное задание модуля:

- выполнить проектное задание по модулю 6, показать работоспособность приложения или его части и описать результат в README;
- зафиксировать ограничения решения, список проверенных сценариев и план доработки.

Чек-лист результата по модулю:

- результат размещен в репозитории и имеет осмысленную историю изменений;
- приложение или учебный фрагмент запускается по инструкции;
- код соответствует теме модуля и демонстрирует освоение заявленных результатов;
- в README отражены назначение работы, порядок запуска, проверенные сценарии и ограничения;
- слушатель может объяснить ключевые решения, ошибки и способы их устранения.

Промежуточная проверка результата. зачет проводится по результатам практического задания, проверки репозитория, краткого устного опроса и анализа качества выполненного кода.

Минимальный уровень освоения предполагает, что слушатель выполняет обязательную часть задания, может запустить проект, объяснить основные решения и исправить типовые ошибки по замечаниям преподавателя. Повышенный уровень предполагает аккуратную архитектуру, обработку крайних случа-

ев, наличие тестов, понятную документацию, расширенную функциональность и самостоятельное обоснование выбранных решений.

Типовые ошибки, учитываемые при проверке:

- код не компилируется или приложение не запускается;
- решение не покрывает обязательные требования задания;
- отсутствует разделение ответственности между компонентами;
- не обработаны типовые ошибки пользователя и состояния загрузки;
- нет понятной инструкции запуска и демонстрационного сценария.

2.3.7. Модуль 7. Навигация, состояние и архитектура Flutter-приложения

Цель модуля: сформировать навыки проектирования навигации, управления состоянием и архитектурного разделения Flutter-приложения.

Объем модуля: 80 академических часов, в том числе лекции – 23 ч., практические занятия – 34 ч., самостоятельная работа – 17 ч., контроль – 6 ч.

Планируемые результаты освоения модуля:

- проектировать навигацию и пользовательские сценарии;
- использовать выбранный подход к управлению состоянием;
- разделять приложение на слои и поддерживать читаемую архитектуру.

Таблица 13 – Тематический план модуля 7 «Навигация, состояние и архитектура Flutter-приложения»

№	Тема	Часы	Краткое содержание
1	Навигация между экранами и передача параметров	9	лекционный материал, практикум, самостоятельная работа и контроль результата
2	Маршруты, вложенная навигация и защита экранов	6	лекционный материал, практикум, самостоятельная работа и контроль результата
3	Локальное состояние виджетов и подъем состояния	7	лекционный материал, практикум, самостоятельная работа и контроль результата
4	Управление состоянием через Provider, Riverpod или BLoC	14	лекционный материал, практикум, самостоятельная работа и контроль результата

№	Тема	Часы	Краткое содержание
5	Слои приложения: presentation, domain, data	12	лекционный материал, практикум, самостоятельная работа и контроль результата
6	Repository, service и dependency injection на учебном уровне	9	лекционный материал, практикум, самостоятельная работа и контроль результата
7	Состояния загрузки, пустого результата и ошибки	7	лекционный материал, практикум, самостоятельная работа и контроль результата
8	Рефакторинг крупного Flutter-приложения	9	лекционный материал, практикум, самостоятельная работа и контроль результата
9	Архитектурный практикум	7	лекционный материал, практикум, самостоятельная работа и контроль результата

Содержание модуля: Модуль посвящен переходу от набора экранов к структурированному приложению. Слушатель изучает навигацию, передачу параметров, управление состоянием, разделение слоев и рефакторинг. Итогом становится приложение, где интерфейс, бизнес-логика, сетевой или локальный слой данных разделены и могут проверяться независимо.

Практическая подготовка включает:

- выполнение практикума по темам модуля 7;
- доработка учебного или итогового Flutter-проекта с применением изученных инструментов;
- оформление результата в репозитории с понятной историей изменений, скриншотами и кратким описанием.

Самостоятельная работа включает:

- повторение теоретического материала и разбор типовых ошибок;
- самостоятельное решение дополнительных задач и улучшение пользовательского сценария;
- подготовка фрагмента проекта к проверке и защите.

Проектное задание модуля:

- выполнить проектное задание по модулю 7, показать работоспособность приложения или его части и описать результат в README;
- зафиксировать ограничения решения, список проверенных сценариев и план доработки.

Чек-лист результата по модулю:

- результат размещен в репозитории и имеет осмысленную историю изменений;
- приложение или учебный фрагмент запускается по инструкции;
- код соответствует теме модуля и демонстрирует освоение заявленных результатов;
- в README отражены назначение работы, порядок запуска, проверенные сценарии и ограничения;
- слушатель может объяснить ключевые решения, ошибки и способы их устранения.

Промежуточная проверка результата. зачет проводится по результатам практического задания, проверки репозитория, краткого устного опроса и анализа качества выполненного кода.

Минимальный уровень освоения предполагает, что слушатель выполняет обязательную часть задания, может запустить проект, объяснить основные решения и исправить типовые ошибки по замечаниям преподавателя. Повышенный уровень предполагает аккуратную архитектуру, обработку крайних случаев, наличие тестов, понятную документацию, расширенную функциональность и самостоятельное обоснование выбранных решений.

Типовые ошибки, учитываемые при проверке:

- код не компилируется или приложение не запускается;
- решение не покрывает обязательные требования задания;
- отсутствует разделение ответственности между компонентами;
- не обработаны типовые ошибки пользователя и состояния загрузки;
- нет понятной инструкции запуска и демонстрационного сценария.

2.3.8. Модуль 8. Работа с API, JSON, HTTP и backend-интеграциями

Цель модуля: сформировать навыки интеграции мобильного приложения с внешними API, backend-сервисами, авторизацией, JSON-моделями и сетевой обработкой ошибок.

Объем модуля: 68 академических часов, в том числе лекции – 20 ч., практические занятия – 28 ч., самостоятельная работа – 14 ч., контроль – 6 ч.

Планируемые результаты освоения модуля:

- интегрировать приложение с REST API;
- обрабатывать JSON, статусы ответов, ошибки сети и авторизацию;
- строить устойчивый сетевой слой и демонстрировать работу с внешними данными.

Таблица 14 – Тематический план модуля 8 «Работа с API, JSON, HTTP и backend-интеграциями»

№	Тема	Часы	Краткое содержание
1	HTTP, REST, JSON и статусы ответов	9	лекционный материал, практикум, самостоятельная работа и контроль результата
2	Клиентские библиотеки и организация сетевого слоя	8	лекционный материал, практикум, самостоятельная работа и контроль результата
3	Сериализация и десериализация моделей	9	лекционный материал, практикум, самостоятельная работа и контроль результата
4	Авторизация, токены и защищенные запросы	9	лекционный материал, практикум, самостоятельная работа и контроль результата
5	Перехватчики, повтор запросов и обработка сетевых ошибок	8	лекционный материал, практикум, самостоятельная работа и контроль результата
6	Пагинация, фильтрация и поиск через API	8	лекционный материал, практикум, самостоятельная работа и контроль результата

№	Тема	Часы	Краткое содержание
7	Mock API, тестовые данные и контракт взаимодействия	6	лекционный материал, практикум, самостоятельная работа и контроль результата
8	WebSocket и push-сценарии на обзорном уровне	5	лекционный материал, практикум, самостоятельная работа и контроль результата
9	Интеграционный проект с внешним API	6	лекционный материал, практикум, самостоятельная работа и контроль результата

Содержание модуля: Модуль формирует навыки подключения мобильного приложения к внешнему миру. Рассматриваются HTTP, REST, JSON, авторизация, токены, обработка сетевых ошибок, повтор запросов, пагинация, mock API и контракт взаимодействия. Практическая работа требует показать реальные данные в интерфейсе и корректно обработать загрузку, ошибку и пустой результат.

Практическая подготовка включает:

- выполнение практикума по темам модуля 8;
- доработка учебного или итогового Flutter-проекта с применением изученных инструментов;
- оформление результата в репозитории с понятной историей изменений, скриншотами и кратким описанием.

Самостоятельная работа включает:

- повторение теоретического материала и разбор типовых ошибок;
- самостоятельное решение дополнительных задач и улучшение пользовательского сценария;
- подготовка фрагмента проекта к проверке и защите.

Проектное задание модуля:

- выполнить проектное задание по модулю 8, показать работоспособность приложения или его части и описать результат в README;

- зафиксировать ограничения решения, список проверенных сценариев и план доработки.

Чек-лист результата по модулю:

- результат размещен в репозитории и имеет осмысленную историю изменений;
- приложение или учебный фрагмент запускается по инструкции;
- код соответствует теме модуля и демонстрирует освоение заявленных результатов;
- в README отражены назначение работы, порядок запуска, проверенные сценарии и ограничения;
- слушатель может объяснить ключевые решения, ошибки и способы их устранения.

Промежуточная проверка результата. зачет проводится по результатам практического задания, проверки репозитория, краткого устного опроса и анализа качества выполненного кода.

Минимальный уровень освоения предполагает, что слушатель выполняет обязательную часть задания, может запустить проект, объяснить основные решения и исправить типовые ошибки по замечаниям преподавателя. Повышенный уровень предполагает аккуратную архитектуру, обработку крайних случаев, наличие тестов, понятную документацию, расширенную функциональность и самостоятельное обоснование выбранных решений.

Типовые ошибки, учитываемые при проверке:

- код не компилируется или приложение не запускается;
- решение не покрывает обязательные требования задания;
- отсутствует разделение ответственности между компонентами;
- не обработаны типовые ошибки пользователя и состояния загрузки;
- нет понятной инструкции запуска и демонстрационного сценария.

2.3.9. Модуль 9. Локальное хранение данных и offline-first подход

Цель модуля: сформировать навыки локального хранения данных, кеширования, offline-first сценариев, миграций и синхронизации данных мобильного приложения.

Объем модуля: 56 академических часов, в том числе лекции – 17 ч., практические занятия – 22 ч., самостоятельная работа – 11 ч., контроль – 6 ч.

Планируемые результаты освоения модуля:

- использовать локальное хранилище для настроек, кеша и доменных данных;
- проектировать offline-first сценарии и миграции;
- обрабатывать синхронизацию и конфликты на учебном уровне.

Таблица 15 – Тематический план модуля 9 «Локальное хранение данных и offline-first подход»

№	Тема	Часы	Краткое содержание
1	Shared Preferences и хранение простых настроек	7	лекционный материал, практикум, самостоятельная работа и контроль результата
2	Secure Storage и хранение чувствительных данных	7	лекционный материал, практикум, самостоятельная работа и контроль результата
3	SQLite, sqflite и Drift как инструменты локальной базы	10	лекционный материал, практикум, самостоятельная работа и контроль результата
4	Hive, Isar и NoSQL-подход в мобильном приложении	7	лекционный материал, практикум, самостоятельная работа и контроль результата
5	Кеширование данных и offline-first сценарии	9	лекционный материал, практикум, самостоятельная работа и контроль результата
6	Миграции локального хранилища	5	лекционный материал, практикум, самостоятельная работа и контроль результата
7	Синхронизация, конфликты и устойчивость к потере сети	5	лекционный материал, практикум, самостоятельная работа и контроль результата
8	Практикум локального хранения	6	лекционный материал, практикум, самостоятельная работа и контроль результата

Содержание модуля: Модуль раскрывает хранение данных на устройстве. Слушатель выбирает инструмент хранения под задачу, проектирует кеш, хранит настройки, использует безопасное хранилище и осваивает offline-first сценарии. Отдельно рассматриваются миграции, конфликты синхронизации и устойчивость приложения при временном отсутствии сети.

Практическая подготовка включает:

- выполнение практикума по темам модуля 9;
- доработка учебного или итогового Flutter-проекта с применением изученных инструментов;
- оформление результата в репозитории с понятной историей изменений, скриншотами и кратким описанием.

Самостоятельная работа включает:

- повторение теоретического материала и разбор типовых ошибок;
- самостоятельное решение дополнительных задач и улучшение пользовательского сценария;
- подготовка фрагмента проекта к проверке и защите.

Проектное задание модуля:

- выполнить проектное задание по модулю 9, показать работоспособность приложения или его части и описать результат в README;
- зафиксировать ограничения решения, список проверенных сценариев и план доработки.

Чек-лист результата по модулю:

- результат размещен в репозитории и имеет осмысленную историю изменений;
- приложение или учебный фрагмент запускается по инструкции;
- код соответствует теме модуля и демонстрирует освоение заявленных результатов;
- в README отражены назначение работы, порядок запуска, проверенные сценарии и ограничения;
- слушатель может объяснить ключевые решения, ошибки и способы их устранения.

Промежуточная проверка результата. зачет проводится по результатам практического задания, проверки репозитория, краткого устного опроса и анализа качества выполненного кода.

Минимальный уровень освоения предполагает, что слушатель выполняет обязательную часть задания, может запустить проект, объяснить основные решения и исправить типовые ошибки по замечаниям преподавателя. Повышенный уровень предполагает аккуратную архитектуру, обработку крайних случаев, наличие тестов, понятную документацию, расширенную функциональность и самостоятельное обоснование выбранных решений.

Типовые ошибки, учитываемые при проверке:

- код не компилируется или приложение не запускается;
- решение не покрывает обязательные требования задания;
- отсутствует разделение ответственности между компонентами;
- не обработаны типовые ошибки пользователя и состояния загрузки;
- нет понятной инструкции запуска и демонстрационного сценария.

2.3.10. Модуль 10. Платформенные возможности и нативные интеграции

Цель модуля: сформировать навыки интеграции Flutter-приложения с платформенными возможностями устройств и понимание ограничений Android и iOS.

Объем модуля: 56 академических часов, в том числе лекции – 17 ч., практические занятия – 22 ч., самостоятельная работа – 11 ч., контроль – 6 ч.

Планируемые результаты освоения модуля:

- подключать платформенные возможности устройства;
- запрашивать разрешения и корректно обрабатывать отказ пользователя;
- использовать платформенные каналы для учебной нативной интеграции.

Таблица 16 – Тематический план модуля 10 «Платформенные возможности и нативные интеграции»

№	Тема	Часы	Краткое содержание
1	Платформенные различия Android и iOS	6	лекционный материал, практикум, самостоятельная работа и контроль результата

№	Тема	Часы	Краткое содержание
2	Разрешения пользователя и корректная обработка отказов	6	лекционный материал, практикум, самостоятельная работа и контроль результата
3	Камера, галерея, файлы и медиаданные	8	лекционный материал, практикум, самостоятельная работа и контроль результата
4	Геолокация, карты и фоновые ограничения	8	лекционный материал, практикум, самостоятельная работа и контроль результата
5	Локальные и push-уведомления	8	лекционный материал, практикум, самостоятельная работа и контроль результата
6	Deep links и открытие приложения по внешней ссылке	6	лекционный материал, практикум, самостоятельная работа и контроль результата
7	Платформенные каналы и вызов нативного кода	8	лекционный материал, практикум, самостоятельная работа и контроль результата
8	Интеграционный практикум с возможностями устройства	6	лекционный материал, практикум, самостоятельная работа и контроль результата

Содержание модуля: Модуль показывает, как Flutter-приложение взаимодействует с возможностями устройства и платформы. Слушатель работает с разрешениями, камерой, файлами, геолокацией, уведомлениями, deep links и платформенными каналами. Практика направлена на корректную обработку ограничений Android и iOS.

Практическая подготовка включает:

- выполнение практикума по темам модуля 10;
- доработка учебного или итогового Flutter-проекта с применением изученных инструментов;
- оформление результата в репозитории с понятной историей изменений, скриншотами и кратким описанием.

Самостоятельная работа включает:

- повторение теоретического материала и разбор типовых ошибок;
- самостоятельное решение дополнительных задач и улучшение пользовательского сценария;
- подготовка фрагмента проекта к проверке и защите.

Проектное задание модуля:

- выполнить проектное задание по модулю 10, показать работоспособность приложения или его части и описать результат в README;
- зафиксировать ограничения решения, список проверенных сценариев и план доработки.

Чек-лист результата по модулю:

- результат размещен в репозитории и имеет осмысленную историю изменений;
- приложение или учебный фрагмент запускается по инструкции;
- код соответствует теме модуля и демонстрирует освоение заявленных результатов;
- в README отражены назначение работы, порядок запуска, проверенные сценарии и ограничения;
- слушатель может объяснить ключевые решения, ошибки и способы их устранения.

Промежуточная проверка результата. зачет проводится по результатам практического задания, проверки репозитория, краткого устного опроса и анализа качества выполненного кода.

Минимальный уровень освоения предполагает, что слушатель выполняет обязательную часть задания, может запустить проект, объяснить основные решения и исправить типовые ошибки по замечаниям преподавателя. Повышенный уровень предполагает аккуратную архитектуру, обработку крайних случаев, наличие тестов, понятную документацию, расширенную функциональность и самостоятельное обоснование выбранных решений.

Типовые ошибки, учитываемые при проверке:

- код не компилируется или приложение не запускается;
- решение не покрывает обязательные требования задания;

- отсутствует разделение ответственности между компонентами;
- не обработаны типовые ошибки пользователя и состояния загрузки;
- нет понятной инструкции запуска и демонстрационного сценария.

2.3.11. Модуль 11. Тестирование, отладка, качество и безопасность мобильных приложений

Цель модуля: сформировать навыки проверки качества мобильного приложения через тестирование, отладку, анализ производительности, безопасность и code review.

Объем модуля: 56 академических часов, в том числе лекции – 17 ч., практические занятия – 20 ч., самостоятельная работа – 14 ч., контроль – 5 ч.

Планируемые результаты освоения модуля:

- писать unit, widget и интеграционные тесты;
- отлаживать приложение через DevTools, логи и профилирование;
- проверять безопасность, производительность и доступность интерфейса.

Таблица 17 – Тематический план модуля 11 «Тестирование, отладка, качество и безопасность мобильных приложений»

№	Тема	Часы	Краткое содержание
1	Unit-тестирование бизнес-логики на Dart	10	лекционный материал, практикум, самостоятельная работа и контроль результата
2	Widget-тестирование интерфейса Flutter	10	лекционный материал, практикум, самостоятельная работа и контроль результата
3	Интеграционные тесты пользовательских сценариев	10	лекционный материал, практикум, самостоятельная работа и контроль результата
4	Отладка, DevTools, profiler и анализ потребления ресурсов	10	лекционный материал, практикум, самостоятельная работа и контроль результата
5	Логирование, crash reporting и диагностика пользовательских ошибок	2	лекционный материал, практикум, самостоятельная работа и контроль результата

№	Тема	Часы	Краткое содержание
6	Безопасность мобильного приложения и защита данных	3	лекционный материал, практикум, самостоятельная работа и контроль результата
7	Чек-лист качества кода и code review	7	лекционный материал, практикум, самостоятельная работа и контроль результата
8	Проверка доступности и производительности интерфейса	4	лекционный материал, практикум, самостоятельная работа и контроль результата

Содержание модуля: Модуль посвящен качеству результата. Слушатель осваивает unit-тесты, widget-тесты, интеграционные сценарии, профилирование, логи, crash reporting, безопасность, проверку доступности и производительности. Результаты модуля позволяют оценивать приложение по инженерным критериям, которые применяются при итоговой защите.

Практическая подготовка включает:

- выполнение практикума по темам модуля 11;
- доработка учебного или итогового Flutter-проекта с применением изученных инструментов;
- оформление результата в репозитории с понятной историей изменений, скриншотами и кратким описанием.

Самостоятельная работа включает:

- повторение теоретического материала и разбор типовых ошибок;
- самостоятельное решение дополнительных задач и улучшение пользовательского сценария;
- подготовка фрагмента проекта к проверке и защите.

Проектное задание модуля:

- выполнить проектное задание по модулю 11, показать работоспособность приложения или его части и описать результат в README;
- зафиксировать ограничения решения, список проверенных сценариев и план доработки.

Чек-лист результата по модулю:

- результат размещен в репозитории и имеет осмысленную историю изменений;
- приложение или учебный фрагмент запускается по инструкции;
- код соответствует теме модуля и демонстрирует освоение заявленных результатов;
- в README отражены назначение работы, порядок запуска, проверенные сценарии и ограничения;
- слушатель может объяснить ключевые решения, ошибки и способы их устранения.

Промежуточная проверка результата. зачет проводится по результатам практического задания, проверки репозитория, краткого устного опроса и анализа качества выполненного кода.

Минимальный уровень освоения предполагает, что слушатель выполняет обязательную часть задания, может запустить проект, объяснить основные решения и исправить типовые ошибки по замечаниям преподавателя. Повышенный уровень предполагает аккуратную архитектуру, обработку крайних случаев, наличие тестов, понятную документацию, расширенную функциональность и самостоятельное обоснование выбранных решений.

Типовые ошибки, учитываемые при проверке:

- код не компилируется или приложение не запускается;
- решение не покрывает обязательные требования задания;
- отсутствует разделение ответственности между компонентами;
- не обработаны типовые ошибки пользователя и состояния загрузки;
- нет понятной инструкции запуска и демонстрационного сценария.

2.3.12. Модуль 12. Сборка, публикация, аналитика и сопровождение приложений

Цель модуля: сформировать навыки подготовки приложения к сборке, релизу, публикации, аналитике, мониторингу и сопровождению после выпуска.

Объем модуля: 44 академических часов, в том числе лекции – 14 ч., практические занятия – 17 ч., самостоятельная работа – 8 ч., контроль – 5 ч.

Планируемые результаты освоения модуля:

- готовить релизные сборки и окружения;
- настраивать базовый CI/CD и релизную документацию;
- планировать поддержку приложения, аналитику и мониторинг ошибок.

Таблица 18 – Тематический план модуля 12 «Сборка, публикация, аналитика и сопровождение приложений»

№	Тема	Часы	Краткое содержание
1	Build modes, flavors и переменные окружения	6	лекционный материал, практикум, самостоятельная работа и контроль результата
2	Подпись Android-сборки и подготовка iOS-сборки	6	лекционный материал, практикум, самостоятельная работа и контроль результата
3	CI/CD для мобильного проекта	8	лекционный материал, практикум, самостоятельная работа и контроль результата
4	Требования магазинов приложений и подготовка карточки релиза	6	лекционный материал, практикум, самостоятельная работа и контроль результата
5	Аналитика событий и воронки использования	5	лекционный материал, практикум, самостоятельная работа и контроль результата
6	Мониторинг ошибок, crash reports и поддержка релиза	5	лекционный материал, практикум, самостоятельная работа и контроль результата
7	Версионирование, release notes и управление обновлениями	3	лекционный материал, практикум, самостоятельная работа и контроль результата
8	Регламент сопровождения и обработки обратной связи	5	лекционный материал, практикум, самостоятельная работа и контроль результата

Содержание модуля: Модуль завершает техническую траекторию разработки мобильного продукта. Слушатель изучает build modes, flavors, подпись сборок, базовый CI/CD, требования публикации, аналитику, мониторинг оши-

бок, версионирование и сопровождение релиза. Практическая часть направлена на подготовку приложения к демонстрации и условному выпуску.

Практическая подготовка включает:

- выполнение практикума по темам модуля 12;
- доработка учебного или итогового Flutter-проекта с применением изученных инструментов;
- оформление результата в репозитории с понятной историей изменений, скриншотами и кратким описанием.

Самостоятельная работа включает:

- повторение теоретического материала и разбор типовых ошибок;
- самостоятельное решение дополнительных задач и улучшение пользовательского сценария;
- подготовка фрагмента проекта к проверке и защите.

Проектное задание модуля:

- выполнить проектное задание по модулю 12, показать работоспособность приложения или его части и описать результат в README;
- зафиксировать ограничения решения, список проверенных сценариев и план доработки.

Чек-лист результата по модулю:

- результат размещен в репозитории и имеет осмысленную историю изменений;
- приложение или учебный фрагмент запускается по инструкции;
- код соответствует теме модуля и демонстрирует освоение заявленных результатов;
- в README отражены назначение работы, порядок запуска, проверенные сценарии и ограничения;
- слушатель может объяснить ключевые решения, ошибки и способы их устранения.

Промежуточная проверка результата. зачет проводится по результатам практического задания, проверки репозитория, краткого устного опроса и анализа качества выполненного кода.

Минимальный уровень освоения предполагает, что слушатель выполняет обязательную часть задания, может запустить проект, объяснить основные решения и исправить типовые ошибки по замечаниям преподавателя. Повышенный уровень предполагает аккуратную архитектуру, обработку крайних случаев, наличие тестов, понятную документацию, расширенную функциональность и самостоятельное обоснование выбранных решений.

Типовые ошибки, учитываемые при проверке:

- код не компилируется или приложение не запускается;
- решение не покрывает обязательные требования задания;
- отсутствует разделение ответственности между компонентами;
- не обработаны типовые ошибки пользователя и состояния загрузки;
- нет понятной инструкции запуска и демонстрационного сценария.

2.3.13. Модуль 13. Проектная практика и консультации

Цель модуля: обеспечить методическое сопровождение итогового проекта от выбора темы до предзащиты и доведения результата до демонстрируемого состояния.

Объем модуля: 44 академических часов, в том числе лекции – 5 ч., практические занятия – 28 ч., самостоятельная работа – 11 ч., контроль – 0 ч.

Планируемые результаты освоения модуля:

- сформировать техническое задание итогового проекта;
- реализовать основную функциональность мобильного приложения;
- устранить замечания по коду, интерфейсу, тестам и документации.

Таблица 19 – Тематический план модуля 13 «Проектная практика и консультации»

№	Тема	Часы	Краткое содержание
1	Выбор темы итогового проекта и согласование требований	7	лекционный материал, практикум, самостоятельная работа и контроль результата
2	Проектирование архитектуры, экранов, данных и API	7	лекционный материал, практикум, самостоятельная работа и контроль результата

№	Тема	Часы	Краткое содержание
3	Разработка основной функциональности приложения	20	лекционный материал, практикум, самостоятельная работа и контроль результата
4	Тестирование, документация и демонстрационный сценарий	10	лекционный материал, практикум, самостоятельная работа и контроль результата
5	Консультации, ревью кода и предзащита	0	лекционный материал, практикум, самостоятельная работа и контроль результата

Содержание модуля: Проектная практика объединяет результаты всех модулей. Слушатель выбирает тему, фиксирует требования, проектирует архитектуру, реализует ключевые пользовательские сценарии, готовит тесты, документацию, демонстрационные данные и проходит предзащиту.

Практическая подготовка включает:

- выполнение практикума по темам модуля 13;
- доработка учебного или итогового Flutter-проекта с применением изученных инструментов;
- оформление результата в репозитории с понятной историей изменений, скриншотами и кратким описанием.

Самостоятельная работа включает:

- повторение теоретического материала и разбор типовых ошибок;
- самостоятельное решение дополнительных задач и улучшение пользовательского сценария;
- подготовка фрагмента проекта к проверке и защите.

Проектное задание модуля:

- выполнить проектное задание по модулю 13, показать работоспособность приложения или его части и описать результат в README;
- зафиксировать ограничения решения, список проверенных сценариев и план доработки.

Чек-лист результата по модулю:

- результат размещен в репозитории и имеет осмысленную историю изменений;
- приложение или учебный фрагмент запускается по инструкции;
- код соответствует теме модуля и демонстрирует освоение заявленных результатов;
- в README отражены назначение работы, порядок запуска, проверенные сценарии и ограничения;
- слушатель может объяснить ключевые решения, ошибки и способы их устранения.

Промежуточная проверка результата. зачет проводится по результатам практического задания, проверки репозитория, краткого устного опроса и анализа качества выполненного кода.

Минимальный уровень освоения предполагает, что слушатель выполняет обязательную часть задания, может запустить проект, объяснить основные решения и исправить типовые ошибки по замечаниям преподавателя. Повышенный уровень предполагает аккуратную архитектуру, обработку крайних случаев, наличие тестов, понятную документацию, расширенную функциональность и самостоятельное обоснование выбранных решений.

Типовые ошибки, учитываемые при проверке:

- код не компилируется или приложение не запускается;
- решение не покрывает обязательные требования задания;
- отсутствует разделение ответственности между компонентами;
- не обработаны типовые ошибки пользователя и состояния загрузки;
- нет понятной инструкции запуска и демонстрационного сценария.

2.3.14. Модуль 14. Итоговая аттестация

Цель модуля: оценить готовность слушателя к выполнению нового вида профессиональной деятельности на основе защиты итогового мобильного проекта и анализа его инженерного качества.

Объем модуля: 36 академических часов, в том числе лекции – 0 ч., практические занятия – 0 ч., самостоятельная работа – 0 ч., контроль – 36 ч.

Планируемые результаты освоения модуля:

- представить итоговый проект;

- обосновать архитектурные и продуктовые решения;
- ответить на вопросы по материалам всех модулей.

Таблица 20 – Тематический план модуля 14 «Итоговая аттестация»

№	Тема	Часы	Краткое содержание
1	Подготовка итогового проекта к защите	0	лекционный материал, практикум, самостоятельная работа и контроль результата
2	Проверка репозитория, README, инструкции запуска и тестовых данных	0	лекционный материал, практикум, самостоятельная работа и контроль результата
3	Защита итогового проекта	36	лекционный материал, практикум, самостоятельная работа и контроль результата
4	Ответы на вопросы и экспертная оценка результата	0	лекционный материал, практикум, самостоятельная работа и контроль результата

Содержание модуля: Итоговая аттестация проводится через защиту проекта. Комиссия оценивает приложение, исходный код, архитектуру, запуск, тесты, документацию, соответствие теме, способность слушателя объяснить решения и исправить замечания.

Практическая подготовка включает:

- финальная проверка проекта по чек-листу;
- подготовка демонстрационного сценария, презентации экранов и инструкции запуска;
- защита проекта и ответы на вопросы комиссии.

Самостоятельная работа включает:

- устранение замечаний предзащиты;
- подготовка README, инструкции запуска, тестовых данных и релизной сборки;
- самопроверка по критериям итоговой аттестации.

Проектное задание модуля:

- представить итоговое мобильное приложение с исходным кодом, документацией, демонстрацией и объяснением ключевых решений.

Чек-лист результата по модулю:

- результат размещен в репозитории и имеет осмысленную историю изменений;
- приложение или учебный фрагмент запускается по инструкции;
- код соответствует теме модуля и демонстрирует освоение заявленных результатов;
- в README отражены назначение работы, порядок запуска, проверенные сценарии и ограничения;
- слушатель может объяснить ключевые решения, ошибки и способы их устранения.

Промежуточная проверка результата. итоговая аттестация проводится в форме защиты выпускного проекта с демонстрацией работоспособности, репозитория, документации и ответами на вопросы.

Минимальный уровень освоения предполагает, что слушатель выполняет обязательную часть задания, может запустить проект, объяснить основные решения и исправить типовые ошибки по замечаниям преподавателя. Повышенный уровень предполагает аккуратную архитектуру, обработку крайних случаев, наличие тестов, понятную документацию, расширенную функциональность и самостоятельное обоснование выбранных решений.

Типовые ошибки, учитываемые при проверке:

- проект не запускается по инструкции;
- отсутствует значимая часть заявленной функциональности;
- нет репозитория или истории разработки;
- не представлены доказательства самостоятельного выполнения;
- слушатель не может объяснить ключевые решения.

2.4. Проектная практика и консультации

Проектная практика направлена на объединение результатов всех модулей в итоговом мобильном приложении. Слушатель согласует тему, фиксирует требования, проектирует структуру экранов, модели данных, архитектуру,

сценарии хранения и сетевого взаимодействия, затем реализует приложение в репозитории и готовит демонстрацию.

В рамках проектной практики слушатель обязан представить:

- краткое описание предметной области и пользователей приложения;
- перечень пользовательских сценариев и критериев приемки;
- структуру экранов, навигации и ключевых состояний;
- описание используемых источников данных, локального хранения и API;
- репозиторий с исходным кодом, историей изменений и README;
- инструкцию запуска, скриншоты или видеодемонстрацию;
- результаты самопроверки по чек-листу качества и перечень ограничений решения.

3. Организационно-педагогические условия реализации программы

3.1. Общие условия реализации

Реализация Программы обеспечивается учебно-методическими материалами, электронной информационно-образовательной средой, системой коммуникации с преподавателями, практическими заданиями, критериями оценивания и регламентом обратной связи. Организация обеспечивает учет результатов обучения, хранение сведений о выполненных работах и проведение текущего контроля, промежуточной и итоговой аттестации.

3.2. Материально-технические условия

Для освоения Программы слушателю рекомендуется иметь персональный компьютер или ноутбук с доступом в сеть Интернет, установленной средой разработки, Flutter SDK, Dart SDK, эмулятором Android или возможностью подключить физическое устройство. При наличии технической возможности слушатель может использовать среду для подготовки iOS-сборок. Отсутствие устройства Apple компенсируется выполнением кроссплатформенной разработки, тестированием на Android и описанием особенностей iOS-конфигурации в проектной документации.

Рекомендуемое программное обеспечение:

- операционная система Windows, macOS или Linux;
- Visual Studio Code или Android Studio;
- Flutter SDK, Dart SDK, Android SDK и эмулятор Android;
- Git и доступ к удаленному репозиторию;
- клиент для работы с API и тестовыми запросами;
- инструменты просмотра логов, профилирования и проверки производительности;
- средства видеосвязи и обмена сообщениями для консультаций и защиты.

3.3. Электронная информационно-образовательная среда

Электронная информационно-образовательная среда обеспечивает размещение учебных материалов, заданий, тестов, записей занятий, ссылок на

консультации, сроков сдачи работ, результатов проверки и обратной связи. В ЭИОС фиксируются доступ слушателя к материалам, факты выполнения заданий, результаты текущего контроля, промежуточной аттестации и итоговой аттестации.

Идентификация и аутентификация слушателя, порядок применения прокторинга при необходимости, фиксация результатов контрольных мероприятий, хранение учебных работ, защита персональных данных и разграничение доступа определяются локальным нормативным актом организации о применении электронного обучения и дистанционных образовательных технологий.

3.4. Кадровые условия

Реализация Программы обеспечивается педагогическими работниками и специалистами-практиками, обладающими компетенциями в области программирования, мобильной разработки, разработки пользовательских интерфейсов, работы с данными, тестирования, сопровождения программных продуктов и применения электронного обучения.

3.5. Учебно-методическое обеспечение

Учебно-методическое обеспечение включает рабочие программы модулей, электронные учебные материалы, практические задания, примеры кода, чек-листы проверки, критерии оценивания, методические рекомендации по выполнению проектов и материалы для подготовки к итоговой аттестации.

3.6. Применение электронного обучения и дистанционных образовательных технологий

Программа реализуется с применением электронного обучения и дистанционных образовательных технологий. Электронное обучение используется для предоставления учебных материалов, заданий, записей, методических рекомендаций, средств коммуникации, фиксации результатов контроля и обратной связи. Дистанционные образовательные технологии используются для организации взаимодействия слушателя и педагогических работников на расстоянии, проведения консультаций, проверки работ, промежуточной и итоговой аттестации.

При реализации Программы организация обеспечивает идентификацию слушателя в электронной информационно-образовательной среде, доступ к материалам в течение периода обучения, возможность сдачи работ, получение обратной связи и фиксацию результатов освоения Программы. Технические требования к устройству, программному обеспечению, каналам связи и учетным данным доводятся до слушателя до начала обучения или при предоставлении доступа к образовательной среде.

3.7. Информационная безопасность, персональные данные и этика

При реализации Программы обеспечивается соблюдение требований законодательства о персональных данных, правил работы с учетными записями, учебными материалами, проектными файлами и результатами аттестации. Доступ к электронной информационно-образовательной среде предоставляется индивидуально и не подлежит передаче третьим лицам.

Слушатель обязан соблюдать академическую добросовестность, не представлять чужие работы как собственные, корректно указывать источники заимствований, не нарушать права третьих лиц на программный код, данные, изображения, тексты и иные материалы. При выявлении признаков несамостоятельного выполнения работы организация вправе назначить дополнительную проверку, собеседование, повторную сдачу или иные процедуры, предусмотренные локальными нормативными актами.

3.8. Локальные процедуры обучения

Организация реализации Программы обеспечивается локальными нормативными актами ООО «Иннопрог», регулирующими прием и зачисление слушателей, режим занятий, применение электронного обучения и дистанционных образовательных технологий, текущий контроль, промежуточную и итоговую аттестацию, пересдачи, апелляции, отчисление, восстановление, выдачу документов о квалификации, внесение сведений в ФИС ФРДО и обработку персональных данных.

Локальные процедуры применяются в части, не противоречащей законодательству Российской Федерации, условиям договора об образовании и настоящей Программе. Слушатель информируется о значимых правилах реализации

Программы до начала обучения или при предоставлении доступа к образовательной среде.

3.9. Обновление программы

Содержание Программы может актуализироваться с учетом изменений законодательства Российской Федерации, требований профессиональной деятельности, развития технологий Dart, Flutter, Android, iOS, инструментов тестирования, сборки, публикации и сопровождения мобильных приложений.

4. Оценка качества освоения программы

4.1. Общие положения

Оценка качества освоения Программы включает текущий контроль успеваемости, промежуточную аттестацию по модулям, проектную практику и итоговую аттестацию. Оценивание направлено на подтверждение достижения планируемых результатов обучения и готовности слушателя применять полученные компетенции в практической деятельности.

4.2. Текущий контроль успеваемости

Текущий контроль осуществляется в ходе изучения модулей и включает проверку тестов, практических заданий, репозитория, мини-проектов, ответов на вопросы, участия в консультациях и качества доработки по замечаниям преподавателя.

При текущем контроле оцениваются:

- соответствие выполненной работы заданию;
- работоспособность кода и приложения;
- структура проекта и читаемость кода;
- корректность интерфейса и пользовательских сценариев;
- обработка ошибок, пустых состояний, сетевых сбоев и ограничений устройства;
- наличие инструкции запуска, скриншотов, тестовых данных и краткого описания;
- самостоятельность выполнения и способность объяснить результат.

4.3. Промежуточная аттестация

Промежуточная аттестация проводится по модулям в форме зачета. Зачет может включать тестирование, практическое задание, защиту мини-проекта, проверку репозитория, устное собеседование и анализ исправлений после обратной связи.

Зачет считается пройденным, если слушатель выполнил обязательную часть практического задания, представил работоспособный результат, оформил

материалы проверки и подтвердил понимание решений при устном опросе. При наличии замечаний слушатель получает перечень доработок и может представить исправленную работу в порядке, установленном локальным актом организации.

4.4. Пересдача, апелляция и подтверждение самостоятельности

Порядок пересдачи, сроки устранения замечаний, количество попыток, основания допуска к повторной сдаче и порядок подачи апелляции определяются локальным актом организации. При рассмотрении спорной оценки учитываются текст задания, критерии проверки, представленный репозиторий, история изменений, комментарии преподавателя и пояснения слушателя.

Самостоятельность выполнения подтверждается историей репозитория, устной защитой, способностью объяснить код, архитектуру, ограничения, ошибки и принятые решения. При выявлении признаков несамостоятельного выполнения организация вправе назначить дополнительное собеседование или индивидуальное практическое задание.

4.5. Допуск к итоговой аттестации

К итоговой аттестации допускается слушатель, который освоил учебный план, выполнил обязательные практические задания, прошел промежуточную аттестацию, представил итоговый проект, оформил репозиторий и подготовил материалы защиты.

Основаниями для недопуска могут являться отсутствие итогового проекта, отсутствие доступа к репозиторию, невозможность запуска приложения, невыполнение обязательных модулей, нарушение сроков устранения критических замечаний или невозможность подтвердить самостоятельность выполнения.

4.6. Итоговая аттестация

Итоговая аттестация проводится в форме защиты итогового мобильного проекта. Слушатель демонстрирует приложение, пользовательские сценарии, структуру кода, архитектуру, работу с данными, тесты, инструкцию запуска и отвечает на вопросы комиссии по материалам всех модулей.

4.6.1. Примерная тематика проектных работ

Таблица 21 – Примерная тематика итоговых проектных работ

№	Тема проекта	Ожидаемый результат
1	Мобильный менеджер задач	приложение для создания, редактирования, фильтрации и закрытия задач с локальным хранением и удобным экраном списка.
2	Трекер привычек и целей	приложение для ежедневных отметок, статистики прогресса, напоминаний и визуализации результатов пользователя.
3	Каталог товаров с избранным	мобильный каталог с карточками товаров, поиском, фильтрацией, избранным и сохранением выбранных позиций.
4	Мобильное приложение прогноза погоды	приложение, получающее данные из внешнего API, отображающее текущую погоду, прогноз и состояние загрузки или ошибки.
5	Приложение личных финансов	система учета доходов и расходов, категорий, периода, сводки и локального хранения финансовых записей.
6	Приложение для бронирования столиков	мобильный интерфейс выбора даты, времени, количества гостей, контактных данных и статуса заявки на бронирование.
7	Приложение записи на услуги	сервис для выбора специалиста, услуги, даты, времени, подтверждения записи и отображения истории посещений.
8	Учебная CRM для малого бизнеса	приложение с клиентами, заказами, статусами, заметками, поиском и локальной базой данных.
9	Приложение доставки с картой	приложение для отображения заказов, маршрутов, геолокации, статуса доставки и карточки заказа.

№	Тема проекта	Ожидаемый результат
10	Мобильный интернет-магазин	приложение с каталогом, карточкой товара, корзиной, оформлением заказа, авторизацией и API-интеграцией.
11	Новостное приложение	приложение для чтения ленты новостей из API, поиска, сохранения статей и offline-доступа к избранному.
12	Фитнес-трекер тренировок	приложение для расписания тренировок, упражнений, подходов, прогресса и уведомлений.
13	Приложение для заметок с синхронизацией	приложение для создания заметок, тегов, поиска, локального кеша и синхронизации с backend-сервисом.
14	Приложение с авторизацией и личным кабинетом	мобильный проект с регистрацией, входом, защищенными экранами, профилем пользователя и обработкой ошибок API.
15	Продакшн-версия Flutter-приложения	итоговый проект с архитектурой, локальным хранением, API, тестами, сборкой, релизной конфигурацией, README и демонстрационным сценарием.

4.6.2. Обязательные требования к итоговому проекту

Итоговый проект должен включать:

- мобильное приложение на Dart и Flutter;
- минимум четыре содержательных экрана или один сложный сценарий с несколькими состояниями;
- навигацию между экранами и передачу данных;
- управление состоянием через выбранный подход;
- работу с локальными данными или внешним API;
- обработку загрузки, пустого состояния, ошибки и успешного результата;
- минимум один вид автоматизированной проверки: unit-тест, widget-тест или интеграционный сценарий;

- README с описанием проекта, инструкцией запуска, стеком, скриншотами и перечнем ограничений;
- демонстрационные данные или сценарий проверки;
- историю изменений в репозитории.

4.6.3. Порядок проведения итоговой аттестации

Итоговая аттестация проводится комиссией. Защита включает краткое представление темы и целей проекта, демонстрацию пользовательских сценариев, обзор архитектуры и структуры репозитория, объяснение работы с данными, тестов, инструкции запуска и ответы на вопросы. Комиссия вправе попросить слушателя внести небольшое изменение в проект, объяснить отдельный фрагмент кода, показать обработку ошибки или описать дальнейшее развитие приложения.

4.6.4. Критерии оценивания итогового проекта

Таблица 22 – Критерии оценивания итогового проекта

Критерий	Баллы	Показатели оценки
Функциональность и пользовательские сценарии	25	приложение реализует заявленные сценарии, корректно обрабатывает ввод, ошибки, пустые состояния, навигацию и ключевые действия пользователя.
Качество интерфейса Flutter	15	интерфейс структурирован, адаптивен, аккуратно использует виджеты, темы, формы, списки, состояния загрузки и сообщения пользователю.
Архитектура и качество кода	20	код разделен на слои, соблюдается декомпозиция, состояние приложения организовано осознанно, зависимости и модели данных оформлены корректно.
Работа с данными и интеграциями	15	реализована работа с локальным хранением или API, присутствует обработка ошибок, сериализация, кеширование или синхронизация по теме проекта.

Критерий	Баллы	Показатели оценки
Тестирование, надежность и безопасность	10	представлены тесты или проверочные сценарии, учтены чувствительные данные, разрешения, ошибки сети и качество обработки исключений.
Документация, репозиторий и запуск	10	README, инструкция запуска, скриншоты, структура проекта и история коммитов позволяют проверить результат без дополнительного разъяснения.
Защита и ответы на вопросы	5	слушатель уверенно объясняет решения, ограничения, ошибки, выбор инструментов и дальнейшее развитие проекта.

Итоговая оценка выставляется по 100-балльной шкале. Минимальный порог положительного результата – 70 баллов. При наличии критических нарушений комиссия вправе выставить неудовлетворительный результат независимо от суммы баллов.

4.7. Внутренняя и внешняя оценка качества реализации программы

Внутренняя оценка качества реализации Программы включает анализ результатов текущего контроля, промежуточной аттестации, итоговой аттестации, обратной связи слушателей, полноты учебно-методических материалов, соблюдения сроков проверки работ и качества сопровождения проектной практики.

Внешняя независимая оценка качества может проводиться путем привлечения работодателей, профильных специалистов, экспертов по мобильной разработке, анализа выпускных проектов, экспертной оценки материалов Программы, отзывов заказчиков образовательных услуг и мониторинга профессиональных результатов выпускников.

5. Методические материалы

5.1. Методические указания для слушателей

Слушателю рекомендуется изучать материалы последовательно, выполнять практические задания в установленные сроки, вести репозиторий с первых недель обучения, фиксировать вопросы и ошибки, регулярно проверять запуск приложения на устройстве или эмуляторе, оформлять README после каждого значимого этапа и использовать чек-листы перед сдачей работы.

При выполнении мобильных проектов необходимо уделять внимание пользовательскому сценарию, устойчивости приложения, читаемости интерфейса, обработке ошибок, безопасности данных, корректному запросу разрешений и понятной инструкции запуска.

5.2. Методические указания для преподавателей и проверяющих

Проверка работ должна учитывать не только факт выполнения задания, но и качество инженерного решения: структуру проекта, декомпозицию, читаемость кода, корректность пользовательских сценариев, обработку ошибок, работу с данными, наличие тестов, документацию и способность слушателя объяснить результат. Обратная связь должна быть конкретной, проверяемой и связанной с критериями оценивания.

5.3. Рекомендуемая литература и ресурсы

Рекомендуемые источники и ресурсы:

- официальная документация Dart;
- официальная документация Flutter;
- документация Flutter DevTools;
- материалы по Material Design и адаптивным интерфейсам;
- документация по Android Studio, эмуляторам и сборочным инструментам;
- документация по REST API, JSON, HTTP и безопасной работе с токенами;
- материалы по Git, code review, README и командной разработке;
- учебные материалы ООО «Иннопрог», размещенные в электронной информационно-образовательной среде.

Приложение 1. Матрица формирования компетенций

Таблица 23 – Матрица формирования компетенций

№ модуля	Модуль	Компетенции	Формы подтверждения
1	Введение в профессию и организация обучения	ПК-1–ПК-10	результат проверяется через практические задания, зачет, проектную работу или итоговую защиту
2	Основы Dart и алгоритмизация	ПК-2	результат проверяется через практические задания, зачет, проектную работу или итоговую защиту
3	Объектно-ориентированное и асинхронное программирование на Dart	ПК-2, ПК-4	результат проверяется через практические задания, зачет, проектную работу или итоговую защиту
4	Git, командная разработка и инструменты мобильного разработчика	ПК-9	результат проверяется через практические задания, зачет, проектную работу или итоговую защиту
5	Основы SQL и моделирование данных мобильных приложений	ПК-5	результат проверяется через практические задания, зачет, проектную работу или итоговую защиту
6	Flutter: основы интерфейса, виджеты и адаптивная верстка	ПК-3	результат проверяется через практические задания, зачет, проектную работу или итоговую защиту
7	Навигация, состояние и архитектура Flutter-приложения	ПК-1, ПК-4	результат проверяется через практические задания, зачет, проектную работу или итоговую защиту

№ модуля	Модуль	Компетенции	Формы подтверждения
8	Работа с API, JSON, HTTP и backend-интеграциями	ПК-5	результат проверяется через практические задания, зачет, проектную работу или итоговую защиту
9	Локальное хранение данных и offline-first подход	ПК-5	результат проверяется через практические задания, зачет, проектную работу или итоговую защиту
10	Платформенные возможности и нативные интеграции	ПК-6	результат проверяется через практические задания, зачет, проектную работу или итоговую защиту
11	Тестирование, отладка, качество и безопасность мобильных приложений	ПК-7	результат проверяется через практические задания, зачет, проектную работу или итоговую защиту
12	Сборка, публикация, аналитика и сопровождение приложений	ПК-8	результат проверяется через практические задания, зачет, проектную работу или итоговую защиту
13	Проектная практика и консультации	ПК-1–ПК-10	результат проверяется через практические задания, зачет, проектную работу или итоговую защиту
14	Итоговая аттестация	ПК-1–ПК-10	результат проверяется через практические задания, зачет, проектную работу или итоговую защиту

Приложение 2. Оценочные материалы промежуточной аттестации

Промежуточная аттестация проводится в форме зачета по каждому содержательному модулю. Примеры заданий являются ориентировочными и могут уточняться преподавателем с сохранением проверяемых результатов.

Таблица 24 – Примеры заданий промежуточной аттестации

Модуль	Пример задания	Критерии зачета
Основы Dart и алгоритмизация	реализовать консольную программу обработки списка задач, товаров или расходов с фильтрацией, поиском и отчетом	код запускается, корректно обрабатывает ввод, использует функции и коллекции, содержит понятные проверки ошибок
ООП и асинхронность на Dart	создать модель предметной области, сервисный слой и асинхронную загрузку тестовых данных	выделены классы, модели и сервисы, применены Future или Stream, ошибки обработаны явно
Git и инструменты	оформить репозиторий, ветку задачи, README, коммиты и результат проверки analyzer	история изменений логична, README позволяет запустить проект, исправлены замечания analyzer
SQL и моделирование данных	спроектировать схему данных мобильного приложения и написать набор SQL-запросов	таблицы связаны корректно, запросы решают задачу, ограничения и ключи обоснованы
Flutter UI	создать несколько экранов приложения с формой, списком, валидацией, темой и адаптивностью	интерфейс запускается, состояния отображаются корректно, экран проверен на разных размерах
Навигация и состояние	реализовать навигацию, передачу параметров и управление состоянием выбранным подходом	сценарий пользователя проходит без потери состояния, код разделен на слои
API-интеграции	подключить внешний API или mock API, отобразить данные и обработать загрузку, ошибку и пустой результат	сетевой слой отделен, JSON-модели корректны, ошибки сети обрабатываются

Модуль	Пример задания	Критерии зачета
Локальное хранение	реализовать хранение настроек, избранного, задач или кеша данных	данные сохраняются между запусками, миграции или ограничения описаны
Платформенные возможности	подключить одну функцию устройства: геолокацию, камеру, файл, уведомление или deep link	разрешения запрашиваются корректно, отказ пользователя обрабатывается
Тестирование и качество	подготовить unit-тесты, widget-тест или интеграционный сценарий для ключевой логики	тесты запускаются, проверяют значимый сценарий, результаты описаны в README
Сборка и публикация	подготовить release build, окружения, версию, changelog и инструкцию релизной проверки	сборка воспроизводима, конфигурация описана, секреты не размещены в репозитории

Критические ошибки промежуточной аттестации:

- приложение или задание не запускается по инструкции;
- отсутствует значимая часть обязательной функциональности;
- решение полностью расходится с темой модуля;
- отсутствует репозиторий или невозможно проверить историю разработки;
- слушатель не может объяснить собственный код и принятые решения;
- в репозитории размещены секреты, токены или иные чувствительные данные.

Приложение 3. Оценочные материалы итоговой аттестации

Паспорт итогового проекта. Итоговый проект должен быть оформлен как самостоятельное мобильное приложение. Паспорт проекта включает название, цель, целевую аудиторию, пользовательские сценарии, стек технологий, архитектуру, источники данных, инструкцию запуска, перечень тестов, скриншоты, ограничения и план развития.

Таблица 25 – Паспорт итогового проекта

Раздел паспорта	Содержание
Название и цель проекта	кратко указаны назначение приложения, пользовательская проблема и ожидаемый результат
Пользовательские сценарии	описаны основные действия пользователя и критерии успешного прохождения сценариев
Архитектура	показано разделение интерфейса, состояния, бизнес-логики, данных и инфраструктурных интеграций
Данные и API	описаны модели, локальное хранение, сетевые запросы, авторизация и обработка ошибок
Тестирование	перечислены виды проверок, тестовые данные, автоматизированные тесты или ручные сценарии
Запуск и демонстрация	представлены инструкция запуска, скриншоты, демо-сценарий и сведения о проверенном окружении

Таблица 26 – Примерные вопросы к итоговой защите

Блок вопросов	Примерное содержание
Dart	типы данных, null safety, коллекции, функции, классы, асинхронность, Stream
Flutter UI	дерево виджетов, layout, формы, списки, темы, адаптивность, доступность
Архитектура	слои приложения, состояние, repository, service, dependency injection, обработка ошибок
Данные	SQL, локальное хранилище, JSON, REST API, кеширование, синхронизация
Платформа	разрешения, геолокация, камера, файлы, уведомления, deep links, платформенные каналы

Блок вопросов	Примерное содержание
Качество	unit-тесты, widget-тесты, интеграционные проверки, DevTools, профилирование, безопасность
Релиз	build modes, flavors, подпись сборок, CI/CD, аналитика, мониторинг и поддержка

Форма оценочного листа итоговой защиты. Оценочный лист содержит сведения о слушателе, теме проекта, ссылке на репозиторий, дате защиты, составе комиссии, баллах по критериям, замечаниях, итоговой оценке и решении комиссии.

Таблица 27 – Форма оценочного листа итоговой защиты

Критерий	Макс.	Балл	Комментарий комиссии
Функциональность и пользовательские сценарии	25		
Качество интерфейса Flutter	15		
Архитектура и качество кода	20		
Работа с данными и интеграциями	15		
Тестирование, надежность и безопасность	10		
Документация, репозиторий и запуск	10		
Защита и ответы на вопросы	5		

Приложение 5. Чек-лист итогового мобильного проекта.

Таблица 28 – Чек-лист итогового мобильного проекта

Блок проверки	Критерии готовности
Репозиторий	есть история изменений, понятная структура, README, инструкция запуска, сведения о зависимостях и проверенном окружении
Запуск	приложение запускается на эмуляторе или устройстве, указаны версии Flutter и Dart, отсутствуют критические ошибки сборки
Интерфейс	экраны связаны навигацией, формы валидируются, списки и состояния отображаются корректно
Данные	локальное хранение или API работают по заявленному сценарию, ошибки и пустые состояния обработаны
Архитектура	код разделен на компоненты, модели, сервисы, слой данных и интерфейс, логика не сосредоточена в одном файле
Качество	есть тесты или проверочные сценарии, применен analyzer, исправлены критические предупреждения
Безопасность	секреты не опубликованы в репозитории, чувствительные данные не хранятся открыто, разрешения запрашиваются корректно
Демонстрация	подготовлены скриншоты или видео, сценарий показа и ответы на ожидаемые вопросы комиссии

Приложение 4. Перечень локальных нормативных актов, на которые должна опираться реализация программы

Реализация Программы осуществляется с учетом локальных нормативных актов организации, включая:

- положение о разработке, утверждении и обновлении дополнительных профессиональных программ;
- правила приема, зачисления, перевода, восстановления и отчисления слушателей;
- порядок оказания платных образовательных услуг и заключения договоров об образовании;
- положение о применении электронного обучения и дистанционных образовательных технологий;
- положение о текущем контроле, промежуточной и итоговой аттестации;
- порядок пересдачи, апелляции и рассмотрения спорных результатов;
- порядок выдачи, учета и хранения документов о квалификации;
- порядок передачи сведений о документах о квалификации в ФИС ФРДО;
- политика обработки и защиты персональных данных;
- регламент хранения учебных работ, репозиторий, оценочных материалов и результатов аттестации.

Приложение 5. Лист актуализации программы

Приложение 10. Лист актуализации программы. Лист актуализации используется для фиксации изменений Программы «Мобильный разработчик», связанных с обновлением законодательства, профессиональных стандартов, квалификационных требований, учебных материалов, оценочных средств, цифровой образовательной среды и проектной практики.

Таблица 29 – Лист актуализации программы

№	Дата	Основание изменения	Содержание изменения
1	01.05.2026	Первичное утверждение	Программа введена в действие приказом организации.
2			
3			

Приложение 6. Требования к репозиторию, README и демонстрации

Приложение 8. Требования к репозиторию, README и демонстрации. Репозиторий является основным доказательством выполнения проектной работы. Он должен позволять проверить самостоятельность, ход разработки, структуру решения и возможность запуска приложения.

Обязательная структура репозитория:

- исходный код Flutter-приложения в стандартной структуре проекта;
- файл README с описанием назначения приложения, стека, установки, запуска и демонстрационных сценариев;
- файл с указанием версии Flutter и Dart, если это требуется для воспроизводимого запуска;
- папка или раздел с тестами, если тесты предусмотрены проектом;
- скриншоты, видеодемонстрация или описание экранов;
- описание API, тестовых данных, локального хранения и ограничений;
- отсутствие секретов, приватных токенов, персональных данных и ключей доступа в открытом коде.

README итогового проекта должен содержать:

- название проекта и краткое описание решаемой задачи;
- целевую аудиторию и основные пользовательские сценарии;
- используемые технологии и библиотеки;
- инструкцию установки зависимостей и запуска приложения;
- описание структуры проекта и архитектурного подхода;
- описание работы с данными, API, локальным хранилищем и состоянием;
- инструкцию запуска тестов;
- скриншоты ключевых экранов;
- перечень реализованных функций, ограничений и направлений развития.

Демонстрация на защите должна включать:

- запуск приложения и показ ключевых экранов;
- прохождение основного пользовательского сценария;
- демонстрацию обработки ошибки, пустого результата или отсутствия сети;

- показ структуры репозитория и README;
- краткое объяснение архитектуры, состояния и данных;
- ответы на вопросы по коду, инструментам и решениям.

Приложение 9. Рубрикатор типовых ошибок мобильных проектов. Рубрикатор применяется для обратной связи слушателю и фиксации причин снижения оценки. Ошибка считается критической, если она препятствует проверке результата, запуску приложения, прохождению основного сценария или подтверждению самостоятельности.

Таблица 30 – Рубрикатор типовых ошибок мобильных проектов

Группа ошибок	Примеры	Рекомендации по исправлению
Запуск и окружение	отсутствует инструкция запуска, зависимости не устанавливаются, версия Flutter не указана, проект не собирается	зафиксировать окружение, обновить README, проверить чистый запуск, приложить команды установки и запуска
Интерфейс и UX	кнопки не ведут к ожидаемому действию, формы не валидируются, пустые состояния отсутствуют, текст обрезается	проверить пользовательские сценарии, добавить валидацию, адаптивность, сообщения об ошибках и состояния загрузки
Архитектура	вся логика находится в одном файле, UI смешан с сетевыми запросами, отсутствует разделение моделей и сервисов	выделить модели, сервисы, слой данных, состояние и виджеты; описать выбранный подход в README
Данные и API	JSON разбирается вручную без проверок, ошибки API игнорируются, токены размещены в коде, нет обработки отсутствия сети	выделить сетевой слой, использовать модели, обрабатывать статусы и исключения, вынести секреты из репозитория
Локальное хранение	данные теряются после перезапуска, миграции не описаны, чувствительные данные хранятся открыто	выбрать подходящее хранилище, добавить проверку сохранения, описать миграции и использовать защищенное хранение при необходимости

Группа ошибок	Примеры	Рекомендации по исправлению
Платформенные функции	разрешения запрашиваются без объяснения, отказ пользователя ломает сценарий, функция работает только на одном устройстве	добавить обработку отказа, проверить ограничения платформ, описать проверенное окружение и fallback-сценарий
Тестирование	тесты отсутствуют, не запускаются или проверяют незначимый код, ручные сценарии не описаны	добавить unit-тесты бизнес-логики, widget-тест ключевого экрана или подробный ручной сценарий проверки
Документация	README не объясняет назначение проекта, нет скриншотов, нет описания данных и ограничений	дополнить описание проекта, инструкции, экранов, API, хранилища, тестов и планов развития
Самостоятельность	слушатель не объясняет код, история коммитов отсутствует, проект выглядит как готовый шаблон без доработки	провести дополнительную защиту, запросить доработку в реальном времени, проверить понимание архитектуры и решений

Приложение 7. Дорожная карта подготовки выпускного проекта

Приложение 6. Детализированные карты проектных работ. Проектные работы используются для постепенного формирования практического портфолио слушателя. Каждая работа должна иметь понятную цель, пользовательский сценарий, набор экранов или функций, критерии приемки, инструкцию запуска и краткое описание ограничений. При проверке учитываются техническая корректность, удобство использования, структура кода, обработка ошибок и качество оформления результата.

Таблица 31 – Детализированные карты проектных работ

№	Проект	Обязательные функции	Усложнение	Результат проверки
1	Мобильный менеджер задач	создание, редактирование, удаление, фильтрация и закрытие задач; сохранение данных между запусками	категории, сроки, приоритеты, поиск, темная тема	приложение запускается, сценарии задач проходят без ошибок, README содержит инструкцию и скриншоты
2	Трекер привычек и целей	список привычек, ежедневные отметки, календарь выполнения, статистика прогресса	напоминания, серии, графики, экспорт результата	данные корректно сохраняются, интерфейс отображает прогресс и пустые состояния
3	Каталог товаров с избранным	карточки товаров, поиск, фильтры, избранное, подробная карточка	корзина, категории, сортировка, локальный кеш	пользователь может найти товар, открыть карточку и сохранить позицию в избранное
4	Приложение прогноза погоды	запрос города, получение данных из API, экран текущей погоды, прогноз и ошибка сети	геолокация, кеш последнего прогноза, смена единиц измерения	приложение показывает реальные или тестовые данные и корректно обрабатывает отсутствие сети

№	Проект	Обязательные функции	Усложнение	Результат проверки
5	Приложение личных финансов	добавление доходов и расходов, категории, период, итоговая сумма, локальное хранение	диаграммы, бюджет, повторяющиеся операции, экспорт	расчеты корректны, данные сохраняются, пользователь видит сводку по периоду
6	Приложение бронирования столиков	выбор даты, времени, количества гостей, контактных данных и статуса заявки	административный экран, ограничения времени, подтверждение брони	сценарий бронирования проходит от выбора параметров до подтверждения заявки
7	Приложение записи на услуги	список услуг и специалистов, выбор слота, подтверждение записи, история записей	авторизация, отмена записи, уведомления	данные записи отображаются корректно, конфликтные состояния обработаны
8	Учебная CRM для малого бизнеса	клиенты, заказы, статусы, заметки, поиск и карточка клиента	фильтры, воронка статусов, синхронизация с API	приложение позволяет вести список клиентов и отслеживать работу по заказам
9	Приложение доставки с картой	список заказов, карточка заказа, статус, адрес, маршрут или карта	геолокация курьера, push-уведомления, offline-кеш	пользователь видит заказы и может изменить статус доставки по сценарию
10	Мобильный интернет-магазин	каталог, карточка товара, корзина, оформление заказа, авторизация	промокоды, профиль, история заказов, избранное	е-commerce сценарий проходит от выбора товара до оформления заказа
11	Новостное приложение	лента из API, подробная статья, поиск, сохранение избранного	категории, offline-чтение, бесконечная прокрутка	данные загружаются, ошибки API обработаны, избранное сохраняется

№	Проект	Обязательные функции	Усложнение	Результат проверки
12	Фитнес-трекер тренировок	упражнения, тренировки, подходы, прогресс и история занятий	таймер, напоминания, статистика, шаблоны тренировок	пользователь может создать тренировку, отметить выполнение и увидеть прогресс
13	Приложение для заметок с синхронизацией	заметки, теги, поиск, локальный кеш и синхронизация с backend-сервисом	markdown, вложения, разрешение конфликтов	заметки сохраняются локально, синхронизация описана и проверена на учебном сценарии
14	Приложение с авторизацией и личным кабинетом	регистрация, вход, профиль, защищенные экраны, выход из аккаунта	роли, обновление токена, восстановление пароля	приложение корректно разделяет публичные и защищенные сценарии
15	Продакшн-версия Flutter-приложения	архитектура, локальные данные, API, тесты, сборка, README и демонстрация	CI/CD, аналитика, crash reporting, релизная конфигурация	проект готов к итоговой защите и воспроизводимо запускается по инструкции

Приложение 7. Карта проверки по модулям. Карта проверки используется преподавателем для единообразной оценки результатов. Для каждого модуля проверяются обязательная часть, качество оформления, объяснение решений и способность слушателя исправить замечания.

Таблица 32 – Карта проверки результатов по модулям

№	Модуль	Проверяемый артефакт	Критерии положительного результата
1	Введение в профессию и организация обучения	репозиторий, карта обучения, доступы, установленная среда	среда подготовлена, репозиторий создан, слушатель понимает правила обучения и критерии итогового проекта
2	Основы Dart и алгоритмизация	набор программ и мини-проект на Dart	код запускается, задачи решены, функции и коллекции применены корректно, ошибки ввода обработаны
3	ООП и асинхронность на Dart	модели, сервисы, асинхронная загрузка данных	классы выделены осмысленно, асинхронный код работает предсказуемо, ошибки не скрываются
4	Git и инструменты разработчика	репозиторий, ветки, README, результаты analyzer	история изменений логична, код отформатирован, README позволяет проверить работу
5	SQL и данные мобильного приложения	схема данных, SQL-запросы, описание локального или серверного хранения	данные нормализованы на достаточном уровне, запросы решают задачу, ограничения описаны
6	Flutter-интерфейсы	приложение с несколькими экранами, формами, списками и темой	интерфейс адаптивен, ошибки ввода видны пользователю, состояния отображаются корректно
7	Навигация, состояние и архитектура	проект с навигацией, выбранным состоянием и разделением слоев	сценарии проходят без потери состояния, код разделен на понятные компоненты

№	Модуль	Проверяемый артефакт	Критерии положительного результата
8	API и backend-интеграции	сетевой слой, модели JSON, экран данных из API	загрузка, ошибка, пустое состояние и успешный результат обработки
9	Локальное хранение и offline-first	локальное хранилище, кеш, миграции или синхронизация	данные сохраняются между запусками, сценарий offline-поведения описан и проверен
10	Платформенные возможности	интеграция с камерой, геолокацией, файлами, уведомлениями или deep links	разрешения запрашиваются корректно, отказ пользователя обработан, ограничения платформ описаны
11	Тестирование и качество	тесты, логи, результаты проверки, исправления по code review	тесты запускаются, критические ошибки устранены, код проверен analyzer
12	Сборка и публикация	релизная конфигурация, инструкция сборки, changelog	сборка воспроизводима, секреты не опубликованы, версия и окружения описаны
13	Проектная практика	итоговый проект до защиты	требования зафиксированы, основная функциональность реализована, документация подготовлена
14	Итоговая аттестация	выпускной проект и защита	проект представлен, приложение запускается, слушатель объясняет решения и отвечает на вопросы

Приложение 11. Дорожная карта подготовки выпускного проекта. Дорожная карта используется как методический ориентир для поэтапной подготовки итогового аттестационного проекта по квалификации «Мобильный разработчик».

1. Выбор темы и согласование прикладной задачи с преподавателем или куратором.
2. Формулирование цели проекта, целевой аудитории, пользовательских сценариев и критериев приемки.
3. Подготовка структуры проекта, репозитория, базовой документации и плана реализации.
4. Реализация минимально работоспособной версии и фиксация результата в репозитории.
5. Расширение функциональности, обработка ошибок, подготовка данных, интерфейса или демонстрационного сценария в соответствии со спецификой направления.
6. Проверка качества, тестирование, рефакторинг, оформление инструкций по запуску и использованию.
7. Предзащита или консультационная проверка, устранение замечаний, подготовка презентации и демонстрации.
8. Итоговая защита проекта с ответами на вопросы комиссии и фиксацией результата итоговой аттестации.

Приложение 8. Контрольный лист соответствия требованиям программы

Приложение 12. Контрольный лист соответствия требованиям программы. Контрольный лист применяется для внутренней проверки полноты Программы и готовности ее реализации.

Таблица 33 – Контрольный лист соответствия требованиям программы

№	Проверяемый элемент	Статус	Комментарий
1	Указаны вид программы, квалификация, трудоемкость, форма обучения, язык и срок освоения.	выполнено	
2	Представлены учебный план, календарный учебный график и рабочие программы модулей.	выполнено	
3	Описаны планируемые результаты, профессиональные компетенции и связь с квалификационными ориентирами.	выполнено	
4	Определены организационно-педагогические условия, ЭИОС, ЭО и ДОТ, кадровые и методические условия.	выполнено	
5	Описаны текущий контроль, промежуточная аттестация, итоговая аттестация, передача и апелляция.	выполнено	

№	Проверяемый элемент	Статус	Комментарий
6	Определены документы по итогам обучения и перечень локальных актов, обеспечивающих реализацию.	выполнено	