

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Андрей Драгомирович Хлудков
Должность: директор
Дата подписания: 17.06.2026 18:10:41
Уникальный программный ключ:
880f7c07c583b07b775f6604c39281b15e9512

Федеральное государственное бюджетное образовательное учреждение
высшего образования

**РОССИЙСКАЯ АКАДЕМИЯ НАРОДНОГО ХОЗЯЙСТВА и
ГОСУДАРСТВЕННОЙ СЛУЖБЫ при ПРЕЗИДЕНТЕ РОССИЙСКОЙ
ФЕДЕРАЦИИ**

СЕВЕРО-ЗАПАДНЫЙ ИНСТИТУТ УПРАВЛЕНИЯ

Факультет среднего профессионального образования

УТВЕРЖДЕНА
решением цикловой (методической)
комиссии общепрофессиональных
дисциплин и профессиональных
модулей специальностей 09.02.07
Информатика и вычислительная
техника
Протокол от 31.10.2025 № 2

РАБОЧАЯ ПРОГРАММА ПРАКТИКИ

УП.03.01 Учебная практика

Специальность – 09.02.11 Разработка и управление программным обеспечением

Профиль – на базе основного общего образования

Квалификация – программист

Форма обучения – очная

Год набора – 2026

Санкт-Петербург 2025 год

Автор-составитель: Вилков Владислав Евгеньевич, преподаватель 1 категории ФСПО СЗИУ РАНХиГС.

СОДЕРЖАНИЕ

1. Общие положения	4
1.1. Область применения программы	4
1.2. Место практики в структуре основной профессиональной образовательной программы	4
1.3. Цели и задачи учебной практики	4
1.4. Планируемые результаты обучения по учебной практике	4
2. Структура и содержание учебной практики.....	11
2.1. Объем учебной практики и виды работ	11
2.2. Тематический план и содержание учебной практики.....	11
2.3. Регламент распределения видов работ по учебной практике с ДОТ	14
3. Материалы текущего контроля успеваемости и промежуточной аттестации обучающихся	15
3.1. Формы и методы текущего контроля успеваемости обучающихся и промежуточной аттестации.....	15
3.2. Оценочные средства текущего контроля успеваемости обучающихся	16
3.3. Оценочные средства промежуточной аттестации обучающихся	19
4. Методические указания для обучающихся по освоению учебной практики....	20
5. Учебная литература и ресурсы информационно-телекоммуникационной сети «Интернет»	21
6. Материально-техническая база, информационные технологии, программное обеспечение и информационные справочные системы	22

1 Общие положения

1.1 Область применения программы

Рабочая программа учебной практики УП.03.01 является частью основной профессиональной образовательной программы в соответствии с ФГОС по специальности СПО 09.02.11 «Разработка и управление программным обеспечением».

1.2 Место практики в структуре основной профессиональной образовательной программы

Учебная практика УП.03.01 входит в состав профессионального модуля ПМ.03 «Разработка приложений для мобильных платформ» и представляет собой неотъемлемый элемент профессиональной подготовки обучающихся. Она базируется на освоении дисциплин «Проектирование и разработка пользовательского интерфейса», «Разработка приложений для мобильных платформ», и «Технологии безопасности мобильных платформ» и реализуется на 4 курсе обучения в 7 семестре.

1.3 Цели и задачи учебной практики

Цель учебной практики «УП.03.01 Учебная практика»: формирование у студентов систематизированных знаний и практических навыков в области разработки и тестирования мобильных приложений для платформ Android, Aurora OS и РЕД ОС М, включая: проектирование пользовательских интерфейсов (UI) с учётом принципов UX и требований к доступности; разработку архитектуры приложения с применением современных паттернов (MVC, MVVM, MVI) и компонентного подхода; создание функциональных модулей и интеграцию с внешними API и сервисами (карты, геолокация, Bluetooth, NFC, push-уведомления); реализацию механизмов информационной безопасности (аутентификация, безопасное хранение данных, защита от атак); отладку и комплексное тестирование UI и бизнес-логики — от формирования тестовых сценариев и настройки тестовых платформ до выполнения UI-тестов (Espresso, UI Automator), профилирования производительности (Layout Inspector, Systrace) и проведения A/B-тестов; подготовку приложений к публикации в маркетплейсах (Google Play, RuStore, Aurora Store) с настройкой CI/CD-пайплайнов (например, на GitHub Actions); формирование навыков технической документации: создание документации для UI-компонентов и API, документирование кода и архитектурных решений, оформление отчётности о выполненных этапах тестирования и разработки в соответствии с установленными регламентами.

1.4 Планируемые результаты обучения по учебной практике

Перечень профессиональных компетенций

Код и наименование компетенции	Умения	Знания	Навыки
ПК 3.1. Разрабатывать модули программного обеспечения для мобильных платформ	<ul style="list-style-type: none"> – разрабатывать программный код; – отлаживать приложения на различных устройствах; – работать с системами контроля версий; – использовать паттерны проектирования; – осуществлять тестирование кода; – производить рефакторинг; – интегрировать приложения с облачными сервисами 	<ul style="list-style-type: none"> – основы языков программирования; – принципы ООП и функционального программирования; – архитектуры мобильных приложений (MVC, MVVM, VIPER); – принципы работы основных мобильных ОС (iOS, Android); – жизненный цикл мобильного приложения; – методы оптимизации производительности; – основы работы с графическим интерфейсом и анимацией; – основы безопасности в мобильной разработке; – основы работы с сетью и API; – принципы работы с базами данных на мобильных платформах; – платформы по кроссплатформенной разработке, таких как Flutter, React Native или MAUI. 	<ul style="list-style-type: none"> – разработки модулей программного обеспечения для мобильных платформ; – разработки многопоточных приложений; – оптимизации производительности приложений; – работы с интеграцией сторонних библиотек
ПК 3.2. Проектировать и разрабатывать пользовательский	<ul style="list-style-type: none"> – создавать интуитивно понятные и легко навигируемые интерфейсы; 	<ul style="list-style-type: none"> – принципы дизайна пользовательского интерфейса (UI) и 	<ul style="list-style-type: none"> – создания пользовательских интерфейсов с использованием

<p>интерфейс и пользовательский опыт</p>	<ul style="list-style-type: none"> – использовать анимацию и переходы для улучшения пользовательского опыта; – оптимизировать интерфейс для работы на разных экранах и устройствах; – интегрировать элементы пользовательского интерфейса с серверной частью или базой данных приложения; – анализировать пользовательские данные и обратную связь для улучшения UX; – разрабатывать макеты и прототипы приложений; – владеть инструментами дизайна интерфейса; – глубоко понимать принципы дизайна пользовательского интерфейса и пользовательского опыта; – проводить пользовательские исследования, включая создание опросов, интервью с пользователями и анализ данных; – работать с прототипированием и созданием макетов пользовательского интерфейса; – работать в команде и эффективно взаимодействовать с разработчиками и 	<p>пользовательского опыта (UX);</p> <ul style="list-style-type: none"> – основы графического дизайна и типографики; – гайдлайны и стандарты для создания интерфейсов на платформах iOS и Android; – принципы адаптивного дизайна ; – основы работы с векторной и растровой графикой; – процесс проектирования интерфейса от идеи до реализации; – основные принципы дизайна пользовательского интерфейса, таких как иерархия информации, цветовая гамма, типографика и композиция; – психологию пользователей и их потребности при взаимодействии с веб-приложениями; – современные тенденции в дизайне пользовательского интерфейса и пользовательского опыта; – основные принципы разработки адаптивного и доступного пользовательского интерфейса; – основные технологии веб-разработки, такие 	<p>инструментов и библиотек, таких как UIKit (iOS) и Android XML (Android);</p> <ul style="list-style-type: none"> – разработки адаптивных и мультирезолюционных интерфейсов; – тестирования пользовательского опыта; – проведения юзабилити-тестов; – проектирование пользовательского интерфейса (UI) и пользовательского опыта (UX) для различных веб-приложений и сайтов; – разработки прототипов и макетов пользовательского интерфейса с использованием инструментов, таких как Sketch, Adobe XD или Figma; – проведения пользовательских исследований, включая сбор обратной связи от пользователей и анализ конкурентного рынка; – создания дизайн-системы и стайл-гайдов для обеспечения единообразия визуального стиля и пользовательского опыта; – тестирования и итеративное улучшение пользовательского интерфейса на основе
--	---	--	---

	менеджерами проектов.	как HTML, CSS и JavaScript.	обратной связи пользователей.
<p>ПК 3.3. Проектировать и разрабатывать базы данных для мобильных платформ</p>	<ul style="list-style-type: none"> – проектировать и оптимизировать базы данных; – выполнять CRUD (Create, Read, Update, Delete) операции; – обеспечивать синхронизацию данных между устройствами; – работать с кэшированием данных; – обрабатывать конфликты данных в распределенных системах; – работать с многозадачностью и потоками данных; – владеть языком SQL для работы с базами данных; – глубоко понимать принципы работы с базами данных в программном обеспечении для мобильных платформ; – создавать и оптимизировать структуру баз данных для хранения и обработки данных в мобильных приложениях; – работать с ORM (Object-Relational Mapping) инструментами для более удобного взаимодействия с базами данных; – обеспечивать безопасность и защиту данных при работе с базами 	<ul style="list-style-type: none"> – основы реляционных баз данных; – основы NoSQL и графовых баз данных; – принципы работы с транзакциями; – основы безопасности и шифрования данных; – принципы работы с миграциями баз данных; – основы работы с асинхронными операциями; – основные принципы работы с базами данных в программном обеспечении для мобильных платформ; – различные типы баз данных, таких как реляционные, NoSQL и графовые базы данных; – современные тенденции в разработке мобильных приложений с использованием баз данных; – основные принципы проектирования баз данных для эффективного хранения и обработки данных в мобильных приложениях; – основные технологии разработки мобильных приложений, таких 	<ul style="list-style-type: none"> – работы с SQLite и другими СУБД для мобильных платформ; – разработки эффективных схем баз данных; – работы с NoSQL и графовыми базами данных; – работы с ORM (Object-Relational Mapping) инструментами; – работы с асинхронным доступом к данным; – разработки функций и возможностей для работы с базами данных в программном обеспечении для мобильных платформ; – создания интерфейсов для работы с базами данных, включая CRUD операции (создание, чтение, обновление, удаление данных); – интеграции баз данных в пользовательский интерфейс приложений для удобного доступа и управления данными; – оптимизации работы с базами данных для обеспечения высокой производительности и эффективного использования ресурсов устройства.

	данных в мобильных приложениях.	как Java, Kotlin, Swift или React Native, для работы с базами данных.	
ПК 3.4. Осуществлять внедрение мультимедиа в программное обеспечение для мобильных платформ	<ul style="list-style-type: none"> – работать с разными форматами изображений и аудиофайлами; – создавать графические ресурсы с высоким разрешением; – проектировать интерфейс с учетом визуальных аспектов, таких как цвета, шрифты и стили; – осуществлять анимацию интерфейсных элементов; – обрабатывать и интегрировать аудио в приложение для воспроизведения звуков и музыки; – владеть инструментами для работы с мультимедиа; – понимать принципы работы с изображениями, видео и аудио в программном обеспечении для мобильных платформ; – создавать и редактировать мультимедийные файлы с использованием различных форматов и кодеков; – работать с анимацией и эффектами для создания привлекательных визуальных элементов 	<ul style="list-style-type: none"> – основы графического дизайна и композиции; – различные форматы изображений и их применение; – основы аудиодизайна и звуковой обработки; – принципы анимации и визуальной привлекательности в мобильных приложениях; – основные принципы работы с изображениями, видео и аудио в программном обеспечении для мобильных платформ; – основные форматы и кодеки для работы с мультимедиа; – современные тенденции в дизайне и использовании мультимедиа в приложениях для мобильных устройств; – основные принципы разработки мультимедийных функций с учетом ограниченных ресурсов мобильных устройств; – основные технологии разработки мобильных приложений, таких 	<ul style="list-style-type: none"> – создания и редактирования графических элементов для приложений с использованием специализированных инструментов; – интеграции изображений и иконок в пользовательский интерфейс; – разработки и анимации пользовательских элементов и переходов; – работы с аудиофайлами и интеграции аудио в приложение; – разработки мультимедийных функций и возможностей в программном обеспечении для мобильных платформ; – создания интерфейсов для работы с изображениями, видео и аудио в приложениях для мобильных устройств; – интеграции мультимедийных элементов в пользовательский интерфейс; – оптимизации работы с мультимедиа для обеспечения высокой производительности и эффективного

	<p>в приложениях для мобильных устройств; оптимизировать мультимедийные элементы для обеспечения быстрой загрузки и плавной работы на мобильных устройствах.</p>	<p>как Java, Kotlin, Swift или React Native.</p>	<p>использования ресурсов устройства; получения медиа-данных с помощью механизмов в операционной системе</p>
<p>ПК 3.5. Выполнять тестирование и отладку программного обеспечения</p>	<ul style="list-style-type: none"> – разрабатывать и запускать тестовые сценарии для проверки функциональности программного обеспечения для мобильных платформ; – выявлять и исправлять ошибки и несоответствия в работе ПО; – проводить аппаратное и программное тестирование программного обеспечения для мобильных платформ; – использовать инструменты анализа и отладки для поиска и устранения проблем; – работать с инструментами для обнаружения и исправления ошибок; – работать с отчетами о тестировании; анализировать и устранять утечки памяти 	<ul style="list-style-type: none"> – основы тестирования программного обеспечения; – виды тестирования (функциональное, нагрузочное, UI-тестирование и др.); – принципы работы с отладчиками; – основы continuous integration и continuous delivery (CI/CD); – основы создания тестовых сценариев; – принципы и методы тестирования программного обеспечения для мобильных платформ; – особенности отладки программного обеспечения для мобильных платформ; – принципы работы эмуляторов и симуляторов; методы аппаратного и программного тестирования 	<ul style="list-style-type: none"> – создания тестовых сценариев и единиц тестирования для мобильных платформ; – отладки и анализа проблем в работе мобильных приложений; – использования инструментов и оборудования для тестирования программных компонентов мобильных платформ; работы с эмуляторами и симуляторами для программного обеспечения мобильных платформ
<p>ПК 3.6. Выполнять интеграцию разработанного приложения с</p>	<ul style="list-style-type: none"> – проектировать и реализовывать структуру запросов и ответов при работе с API; 	<ul style="list-style-type: none"> – принципы работы с RESTful API и другими протоколами; 	<ul style="list-style-type: none"> – работы с API сторонних сервисов и платформ для получения данных и функциональности;

внешними системами и платформами	<ul style="list-style-type: none"> – аутентифицировать пользователей через сторонние сервисы, такие как OAuth; – обрабатывать и адаптировать данные, получаемые от сторонних сервисов, для использования в приложении; интегрировать функциональность социальных медиа, осуществлять доступ к аппаратным компонентам устройства и управление ими. 	<ul style="list-style-type: none"> – основы OAuth и авторизации в сторонних сервисах; стандарты и протоколы взаимодействия с внешними сервисами 	<ul style="list-style-type: none"> – интеграции социальных медиа и сетей для авторизации и обмена данными; – использования сторонних библиотек и SDK для расширения функциональности приложения; взаимодействия с аппаратными компонентами устройства
ПК 3.7. Осуществлять защиту данных в мобильных приложениях	<ul style="list-style-type: none"> – разрабатывать и реализовывать меры безопасности; – реализовывать хэширование паролей, сессионные токены и двухфакторную аутентификацию; – осуществлять валидацию данных, поступающих от пользователей; – разрабатывать политику доступа и права пользователей к данным и функциональности приложения; реализовывать меры контроля доступа и аудита для отслеживания действий пользователей и обнаружения несанкционированных действий. 	<ul style="list-style-type: none"> – основные угрозы безопасности мобильных приложений; – принципы криптографии и шифрования данных; – стандарты и протоколы безопасности, такие как HTTPS, OAuth и OpenID Connect; – законодательные и регуляторные требования к защите данных, включая GDPR и HIPAA; – основные принципы безопасности информации и методов ее защиты; – стандартные криптографические алгоритмы для шифрования данных; – методы аутентификации и авторизации пользователей, таких как OAuth или JWT; 	<ul style="list-style-type: none"> – разработки безопасных методов аутентификации и авторизации пользователей; – обработки и хранения конфиденциальных данных; – отслеживания и обработки уязвимостей безопасности; – использования шифрования для защиты данных в покое и в движении; – использования шифрования данных для защиты конфиденциальной информации, такой как пароли, персональные данные пользователей и другие чувствительные данные; – реализации механизмов аутентификации и авторизации для

		<ul style="list-style-type: none"> – многоуровневые механизмы контроля доступа к данным; – методы тестирования на уязвимости безопасности и опыт применения инструментов для их обнаружения; – принципы обеспечения безопасности передачи данных по сети; законодательство и регуляции в области защиты данных и умение применять их в практической разработке мобильных приложений. 	<p>обеспечения доступа только авторизованным пользователям;</p> <ul style="list-style-type: none"> – применения механизмов хеширования для защиты паролей пользователей от несанкционированного доступа; – обеспечения безопасности передачи данных между клиентскими устройствами и серверами с использованием протоколов шифрования, таких как SSL/TLS; – разработки механизмов контроля доступа к данным, чтобы предотвратить несанкционированное чтение, изменение или удаление данных; – проектирования и реализации систем резервного копирования и восстановления данных для обеспечения их сохранности в случае сбоев или потери устройства; – тестирования приложений на уязвимости безопасности, такие как SQL-инъекции, межсайтовые сценарии и другие уязвимости, и принятие мер по их устранению; соблюдение законодательства и
--	--	--	---

			регуляций в области защиты данных
--	--	--	-----------------------------------

В результате освоения учебной практики студент должен:

иметь практический опыт	<ul style="list-style-type: none"> – Проектирования пользовательских интерфейсов (UI) для мобильных платформ (смартфоны, планшеты, смарт-часы) с учётом принципов UX и требований доступности; – Разработки UI с применением Jetpack Compose и XML, адаптации интерфейсов под разные плотности экранов (dpi, sp, dp) и устройства; – Создания дизайн-систем (компоненты, шаблоны, дизайн-токены) и прототипов в Figma с последующим переносом в код; – Реализации архитектуры UI с использованием паттернов MVC, MVVM, MVI и компонентного подхода (фрагменты, ViewModels, Compose Composables); – Интеграции UI с данными из REST API; – Проведения UI-тестирования с использованием Espresso и UI Automator, профилирования производительности через Layout Inspector и Systrace; – Настройки локализации интерфейса и мультиязычности; – Работы с инструментами сборки и конфигурации проекта (Gradle), реализации режимов работы в offline и отладки через Logcat; – Использования Kotlin Coroutines и Flow для асинхронных запросов, подключения базы данных Room и реализации репозиторной архитектуры; – Публикации приложений в маркетплейсах (Google Play, RuStore, Aurora Store), настройки CI/CD-пайплайнов (например, на GitHub Actions); – Реализации механизмов информационной безопасности: безопасного хранения данных (EncryptedSharedPreferences, Android Keystore), аутентификации (OAuth2, OpenID, биометрия), защиты от атак (MITM, инъекция, подмена Activity); – Проведения аудита безопасности с помощью MobSF, внедрения обфускации кода (ProGuard, R8) и проверки защиты от рутирования.
уметь	<ul style="list-style-type: none"> – Применять гайдлайны и принципы юзабилити для Android, Aurora и РЕД ОС М при проектировании UI; – Адаптировать интерфейсы под разные устройства и плотности экранов, настраивать accessibility (TalkBack, контраст, размер текста); – Использовать инструменты навигации (Jetpack Navigation, Deep Links) и управления состоянием (StateFlow, LiveData, remember {}); – Проводить UI-тестирование и A/B-тесты, внедрять системы сбора обратной связи (например, Firebase Feedback); – Настраивать сборку проекта через Gradle, работать с файловой системой и SharedPreferences; – Реализовывать взаимодействие с REST API и Retrofit, обрабатывать сетевые вызовы и ошибки (retry logic); – Организовывать загрузку/отправку медиафайлов, построение экранов с подгрузкой данных (Paging 3); – Собирать .apk/.aab для публикации в маркетплейсах, интегрировать API (Yandex Maps, Geo API, Bluetooth, NFC) и push-уведомления; – Обеспечивать безопасную передачу данных (HTTPS, SSL Pinning), работать с токенами (JWT) и обновлять access-токены; – Анализировать apk через MobSF, проверять root-статус устройства, интегрировать обфускацию в CI;

	– Составлять техническую документацию для UI-компонентов, API и архитектурных решений, оформлять отчётность о выполненных этапах тестирования и разработки.
знать	<ul style="list-style-type: none"> – Принципы UX/UI и гайдлайны для мобильных платформ (Android, Aurora, РЕД ОС М); – Основы мобильной типографики, работы с сетками и плотностью экранов (dpi, sp, dp); – Методы обеспечения доступности интерфейса (контрастность, TalkBack) и создания дизайн систем; – Особенности архитектур UI (MVC vs MVVM vs MVI) и компонентного подхода; – Инструменты UI тестирования (Espresso, UI Automator) и профилирования (Layout Inspector, Systrace); – Архитектуру Android и Aurora OS (Activity, Service, BroadcastReceiver, ContentProvider); – Подходы к работе с данными (Room, DAO, репозиторная архитектура) и асинхронности (Kotlin Coroutines, Flow); – Процессы публикации приложений и настройки CI/CD (GitHub Actions), особенности поддержки разных платформ и устройств; – Модели угроз и принципы защищённой архитектуры приложения (Secure Storage, пермишены, scopes); – Механизмы аутентификации (OAuth2, OpenID, биометрия) и безопасной передачи данных (HTTPS, SSL Pinning); – Инструменты анализа безопасности (MobSF, OWASP) и методы hardening (ProGuard, R8, проверка root статуса); – Регламенты оформления технической документации и отчётности о разработке и тестировании мобильных приложений.

2 Структура и содержание учебной практики

2.1 Объем учебной практики и виды работ

Виды учебной работы	Объем учебной работы, час.
Учебная нагрузка обучающихся всего, в том числе:	108 (3 недели)
практические занятия	108
промежуточная аттестация	-
Форма промежуточной аттестации	Зачет с оценкой

Учебная практика проводится концентрировано, в соответствии с графиком учебного процесса на текущий учебный год.

Учебная практика проводится преподавателями специальности «09.02.11 Разработка и управление программным обеспечением» на учебной базе ФСПО в форме практических занятий. По результатам прохождения учебной практики обучающиеся составляют отчёт о прохождении учебной практики.

Структура, содержание и виды работ учебной практики приведены в Таблице 2.2.

Таблица 2.2. Тематический план и содержание учебной практики

п/п	Наименование тем (разделов)	Содержание и виды работ	Распределение часов	Формируемые компетенции	Формы текущей аттестации
			ПР		
1	Тема 1.1. Проектирование и разработка пользовательского интерфейса	<p>Содержание:</p> <ul style="list-style-type: none"> — цели и задачи практики. Требования охраны труда. Инструктаж по технике безопасности во время прохождения практики. — введение в UX/UI: принципы юзабилити и гайдлайны Android/Aurora/РЕД ОС М; — мобильная типографика, сетки, работа с плотностью экранов (dpi, sp, dp); — цвет, контрастность и адаптивность: доступность для всех категорий пользователей; — создание дизайн-систем: компоненты, шаблоны, дизайн-токены; — поведение и анимации: Material Design Motion, Jetpack Compose Animations; — UX-паттерны под смарт-часы, планшеты и раскладные устройства. <p>Виды работ:</p> <ol style="list-style-type: none"> 1. Разработка интерфейса под смартфон (Compose / XML). 2. Адаптация интерфейса под планшет с использованием Split View. 3. Проектирование UI для «карусели» на смарт-часах (Wear OS). 4. Разработка прототипа в Figma и перенос в код (UI → Dev). 5. Работа с кастомными шрифтами, цветами, стилями. 6. Настройка accessibility (TalkBack, контраст, размер текста). 	34	ПК 3.1 ПК 3.2 ПК 3.3 ПК 3.4 ПК 3.5 ПК 3.6 ПК 3.7	ПЗ, О
2	Тема 1.2. Разработка приложений для мобильных платформ	<p>Содержание:</p> <ul style="list-style-type: none"> — архитектура UI: MVC vs MVVM vs MVI; — компонентный подход: фрагменты, ViewModels, Compose Composables; — работа с RecyclerView и LazyColumn/LazyRow; — навигация: Jetpack Navigation, Deep Links; — управление состоянием: StateFlow, LiveData, remember {}; — локализация интерфейса и мультиязычность. <p>Виды работ:</p> <ol style="list-style-type: none"> 1. Внедрение Jetpack Navigation с аргументами. 2. Построение экрана профиля с ViewModel и LiveData. 3. Использование StateFlow и collectAsState. 4. Настройка нескольких языков через resources. 5. Интеграция UI с данными из REST API. 6. Создание кастомных компонентов UI (настройка Canvas и Paint). 	38		ПЗ, О

3	Тема 1.3. Технологии безопасности мобильных платформ	<p>Содержание:</p> <ul style="list-style-type: none"> — UI-тестирование: Espresso, UI Automator; — производительность интерфейса: анализ с помощью Layout Inspector; — инструменты отладки: Debug Drawer, LeakCanary; — инкрементальная сборка и modular UI; — Design Review: как проводить и зачем; — обратная связь от пользователя: системы сбора и анализа UX. <p>Виды работ:</p> <ol style="list-style-type: none"> 1. Написание UI-тестов с Espresso. 2. Профилирование UI с Layout Inspector и Systrace. 3. Проведение A/B-тестов интерфейса. 4. Внедрение Feedback-системы (например, Firebase Feedback). 5. Отладка и устранение UI-багов. 6. Адаптация интерфейса под РЕД ОС М (особенности рендеринга). 	36		ПЗ, О
		Итого часов:	108		

2.3 Регламент распределения видов работ по учебной практике с ДОТ

Учебная практика реализуется с применением дистанционных образовательных технологий (ДОТ). Распределение видов учебной работы, форматов текущего контроля представлены в Таблице 2.3.

Таблица 2.3. — Распределение видов учебной работы и текущей аттестации

Вид учебной работы	Формат проведения
Практические занятия	Частично с применением ДОТ

Доступ к системе дистанционных образовательных программ осуществляется каждым обучающимся самостоятельно с любого устройства на портале: <https://sziu-de.ranepa.ru>, в соответствии с их индивидуальным паролем и логином к личному кабинету/ профилю.

Текущий контроль, проводимый в системе дистанционного обучения, оцениваются как в системе дистанционного обучения, так и преподавателем вне системы.

Доступ к материалам практической занятий предоставляется в течение всего семестра по мере прохождения освоения программы. Доступ к каждому виду работ и количество попыток на выполнение задания предоставляется ограниченное время согласно регламенту учебной практики, опубликованному в системе дистанционного обучения. Преподаватель оценивает выполненные обучающимися работы не позднее 14 рабочих дней после окончания срока выполнения.

3 Материалы текущего контроля успеваемости и промежуточной аттестации обучающихся

3.1 Формы и методы текущего контроля успеваемости и промежуточной аттестации обучающихся

Текущий контроль и оценка результатов учебной практики УП.01.01 осуществляется преподавателем в процессе проведения учебной практики на практических занятиях.

Для контроля успеваемости используется метод опроса (О) в форме теоретических вопросов и практический метод (ПЗ) в форме выполнения практических заданий.

Опрос – как основной вид устной проверки, может использоваться фронтальный и индивидуальный. Студент отвечает на теоретический вопрос каждой темы, количество вопросов по теме – от одного до трёх, вопрос определяет преподаватель учебной практики.

Критерии оценивания опроса

Оценки «отлично» заслуживает студент, обнаруживший глубокое знание материала, умение свободно выполнять задания, понимающий взаимосвязь основных понятий;

Оценки «хорошо» заслуживает студент, обнаруживший полное знание материала; успешно выполняющий предусмотренные задания; и допустивший незначительные ошибки: неточность фактов, стилистические ошибки;

Оценки «удовлетворительно» заслуживает студент, обнаруживший знания основного материала в объеме, необходимом для дальнейшего изучения дисциплины. Справляющийся с выполнением заданий; допустивший погрешности в ответе, но обладающий необходимыми знаниями для их устранения под руководством преподавателя;

Оценки «неудовлетворительно» заслуживает студент, обнаруживший существенные пробелы в знании основного материала; не справляющийся с выполнением заданий, допустивший серьезные погрешности в ответах, нуждающийся в повторении основных разделов курса под руководством преподавателя.

Критерии оценивания практических заданий:

«отлично» – верно и полностью выполнено задание, выбрано наиболее полное и рациональное решение задачи, верно отвечает на вопросы по теоретической части практической работы;

«хорошо» – верно и полностью выполнено задание, имеются погрешности в решении, имеются неточности в ответах на вопросы по работе;

«удовлетворительно» – задание выполнено не полностью, имеются значительные погрешности или приняты нерациональные способы решения; затрудняется в ответах на вопросы по работе;

«неудовлетворительно» – неправильно выполнено задание, не отвечает на вопросы по работе.

Формы текущего контроля

№ п/п	Название темы	Формы текущего контроля успеваемости
1	Тема 1.1. Проектирование и разработка пользовательского интерфейса	ПЗ, О
2	Тема 1.2. Разработка приложений для мобильных платформ	ПЗ, О
3	Тема 1.3. Технологии безопасности мобильных платформ	ПЗ, О

Примечание. В столбце «Форма текущего контроля успеваемости, промежуточной аттестации» перечисляются все используемые в учебном процессе по данной дисциплине формы контроля освоения материала. (ПЗ – практическое задание, О – опрос).

Формы промежуточного контроля

Форма промежуточной аттестации по учебной практике — зачёт с оценкой в 7 семестре.

Метод контроля:

письменный — отчёт по учебной практике УП.03.01.;

устный — защита отчёта.

Критерии оценивания промежуточной аттестации:

Оценка «зачтено-отлично»: письменный отчёт оформлен без ошибок: соблюдены все требования к структуре (титульный лист, введение, основная часть, заключение, список литературы, приложения), отсутствуют ошибки в оформлении и содержании. Материал глубоко раскрывает все аспекты практики, включая индивидуальное задание. Выводы логичны, обоснованы, подкреплены фактами. Используются первоисточники, приложения (схемы, таблицы, фотоматериалы) уместны и информативны. Отчёт сдан в срок. При защите студент демонстрирует превосходное владение материалом: чётко и грамотно излагает содержание отчёта, уверенно использует профессиональную терминологию. Даёт исчерпывающие ответы на все вопросы, включая дополнительные, показывает глубокое понимание темы, способность к анализу и формулировке выводов. Речь структурирована, убедительна, при необходимости сопровождается иллюстративными материалами.

Оценка «зачтено-хорошо»: письменный отчёт в целом соответствует методическим указаниям, однако допущены 1–2 незначительных недочёта в оформлении (например, мелкие ошибки в приложениях). Содержание отражает ключевые аспекты практики, но анализ некоторых вопросов проведён не в полной мере. Выводы корректны, но недостаточно развёрнуты. Используются необходимые первоисточники, однако их анализ не всегда глубок. Отчёт сдан своевременно или с незначительным опозданием, согласованным с руководителем. Во время защиты студент грамотно излагает суть отчёта, но допускает 1–2 небольшие речевые или логические ошибки. На основные вопросы отвечает правильно, однако при ответе на дополнительные вопросы испытывает небольшие затруднения. Показывает хорошее понимание темы, но не всегда способен глубоко проанализировать отдельные аспекты. Терминологию использует уверенно, но иногда допускает неточности.

Оценка «зачтено-удовлетворительно»: письменный отчёт имеет существенные недостатки в оформлении: ошибки в структуре, форматировании, оформлении приложений.

Содержание неполное: упущены отдельные важные аспекты практики, индивидуальное задание выполнено поверхностно или не выполнено. Выводы формальные, не отражают сути работы. Первоисточники использованы фрагментарно, анализ поверхностный. Отчёт сдан с опозданием, не согласованным с руководителем. При защите студент излагает содержание отчёта неполно, сбивчиво, с существенными логическими или речевыми ошибками. Ответы на вопросы неполные, содержат ошибки; на дополнительные вопросы ответить не может. Понимание темы поверхностное, анализ формальный. Терминология используется ограниченно, с ошибками. Тем не менее, базовый уровень понимания материала и выполнения заданий прослеживается.

Оценка «не зачтено-неудовлетворительно»: письменный отчёт не соответствует методическим указаниям, содержит множественные ошибки в оформлении и содержании. Ключевые аспекты практики не раскрыты, анализ отсутствует, выводы формальны или отсутствуют. Первоисточники не использованы, приложения либо отсутствуют, либо не соответствуют теме. Отчёт не сдан в срок или представлен в неполном объёме. Возможны случаи плагиата или использования недостоверных данных. Во время защиты студент не может чётко изложить содержание отчёта: речь несвязная, содержит множественные ошибки. Не отвечает на поставленные вопросы или даёт ошибочные ответы. Не демонстрирует понимания темы, не способен анализировать и делать выводы. Профессиональную терминологию не использует или использует неправильно. Материал не освоен, практика фактически не выполнена или выполнена ненадлежащим образом.

3.2 Оценочные средства текущего контроля успеваемости обучающихся

Примеры типовых заданий для учебной практики

Тема 1.1. Проектирование и разработка пользовательского интерфейса

Примеры типовых заданий:

1. Задание на разработку мобильного интерфейса:
 - создать интерфейс приложения для смартфона в Figma с учётом гайдлайнов Material Design;
 - реализовать адаптацию интерфейса под разные плотности экранов (dpi, sp, dp), подготовить макеты для экранов с плотностью 160 dpi, 320 dpi и 480 dpi;
 - перенести дизайн из Figma в код с использованием Jetpack Compose или XML;
 - предоставить скриншоты работы приложения на эмуляторах с разными разрешениями и экранами.
2. Практическое задание по адаптации интерфейса:

- взять разработанный ранее интерфейс для смартфона и адаптировать его под планшетный формат с использованием Split View;
 - обеспечить корректное отображение контента в режимах портретной и альбомной ориентации;
 - проверить работу интерфейса на эмуляторе планшета с диагональю 10 дюймов;
 - составить отчёт с описанием внесённых изменений и скриншотами.
3. Задание по проектированию UI для носимых устройств:
- спроектировать интерфейс «карусели» для смарт часов на платформе Wear OS в Figma;
 - учесть ограничения экрана смарт часов (маленький размер, круглая форма);
 - реализовать анимации перехода между элементами карусели;
 - экспортировать дизайн и интегрировать его в тестовое приложение для Wear OS;
 - продемонстрировать работу интерфейса на эмуляторе смарт-часов.
4. Практикум по работе с типографикой и цветами:
- подобрать и подключить кастомные шрифты для мобильного приложения;
 - разработать цветовую палитру с учётом требований контрастности (WCAG AA);
 - настроить стили текста и цветов в ресурсах приложения (styles.xml или Theme.kt);
 - продемонстрировать изменение шрифтов и цветов в интерфейсе при смене темы (светлая/тёмная).
5. Кейс по настройке доступности (accessibility):
- взять существующее мобильное приложение и настроить его для работы с TalkBack;
 - увеличить контрастность элементов интерфейса до уровня WCAG AA;
 - добавить контент-описание (content description) для всех интерактивных элементов;
 - протестировать приложение с включённым TalkBack и составить отчёт о найденных и исправленных проблемах.

Теоретические вопросы для устного опроса:

1. Какие основные принципы юзабилити необходимо учитывать при проектировании мобильных интерфейсов?
2. В чём разница между dp, sp и px в Android? Для чего используется каждая единица измерения?
3. Что такое дизайн-система? Какие компоненты она обычно включает?
4. Как обеспечить доступность интерфейса для пользователей с нарушениями зрения?
5. Какие анимации рекомендуется использовать в мобильных приложениях согласно Material Design Motion?
6. Каковы особенности проектирования интерфейсов для смарт-часов?
7. Что такое дизайн-токены? Как они помогают поддерживать консистентность дизайна?
8. Как адаптировать интерфейс под разные форм-факторы устройств (смартфоны, планшеты, складные устройства)?

Тема 1.2. Разработка приложений для мобильных платформ

Примеры типовых заданий:

1. Задание на внедрение навигации:
 - разработать мобильное приложение с несколькими экранами (минимум 3);
 - внедрить Jetpack Navigation с передачей аргументов между экранами;
 - настроить Deep Links для открытия конкретных экранов из внешних источников;
 - продемонстрировать навигацию и передачу данных между экранами.
2. Практикум по работе с ViewModel и LiveData:
 - создать экран профиля пользователя с использованием ViewModel и LiveData;
 - привязать данные пользователя (имя, фото, статус) к элементам интерфейса;
 - обеспечить сохранение состояния экрана при смене ориентации устройства;
 - показать обновление данных в реальном времени при изменении LiveData.
3. Задание по работе с состоянием в Compose:
 - разработать простой счётчик с использованием StateFlow и collectAsState;
 - отобразить текущее значение счётчика в интерфейсе;
 - добавить кнопки для увеличения и уменьшения значения;
 - продемонстрировать обновление интерфейса при изменении состояния.
4. Кейс по локализации интерфейса:

- взять существующее приложение и добавить поддержку минимум двух языков (например, русский и английский);
 - вынести все текстовые строки в ресурсы (strings.xml);
 - настроить переключение языка в настройках приложения;
 - проверить корректность отображения текста на обоих языках.
5. Практическое задание по интеграции с API:
- выбрать публичное REST API (например, JSONPlaceholder или OpenWeatherMap);
 - создать приложение, которое запрашивает данные из API и отображает их в RecyclerView или LazyColumn;
 - обработать возможные ошибки сети и отображения данных;
 - предоставить скриншоты успешного получения и отображения данных.

Теоретические вопросы для устного опроса:

1. В чём отличие архитектур MVC, MVVM и MVI? Приведите примеры использования каждой.
2. Что такое компонентный подход в разработке UI? Какие компоненты используются в Android?
3. Как работает RecyclerView? Для чего нужны ViewHolder и Adapter?
4. Что такое StateFlow и LiveData? В чём их различия и когда какой использовать?
5. Как реализовать локализацию интерфейса в Android приложении?
6. Что такое Deep Links? Как их настроить в приложении?
7. Как управлять состоянием в Jetpack Compose? Что такое remember {}?
8. Какие преимущества даёт использование Jetpack Navigation?

Тема 1.3. Технологии безопасности мобильных платформ

Примеры типовых заданий:

1. Задание на написание UI тестов:
 - выбрать существующее мобильное приложение;
 - написать 3–5 UI тестов с использованием Espresso для проверки основных сценариев (вход в приложение, навигация, ввод данных);
 - запустить тесты на эмуляторе и предоставить отчёт о результатах (пройденные/не пройденные тесты);
 - проанализировать результаты и предложить улучшения для повышения покрытия тестами.

2. Практикум по профилированию UI:
 - использовать Layout Inspector для анализа иерархии View в приложении;
 - выявить «тяжёлые» layouts, вызывающие задержки отрисовки;
 - применить Systrace для анализа производительности интерфейса;
 - оптимизировать проблемные участки и измерить прирост производительности.
3. Задание по проведению A/B тестов:
 - определить элемент интерфейса для A/B теста (например, цвет кнопки, расположение элемента);
 - настроить A/B тест с использованием Firebase Remote Config или аналогичного инструмента;
 - собрать данные о поведении пользователей в обеих группах (кликабельность, время взаимодействия);
 - проанализировать результаты и сделать вывод о наиболее эффективном варианте.
4. Кейс по внедрению системы обратной связи:
 - интегрировать Firebase Feedback или аналогичную систему в мобильное приложение;
 - настроить сбор отзывов пользователей и отправку их на сервер;
 - реализовать отображение формы обратной связи по нажатию кнопки в интерфейсе;
 - протестировать отправку и получение отзывов.
5. Практическое задание по отладке и устранению багов:
 - получить приложение с преднамеренно внесёнными UI багами (например, некорректное отображение элементов, утечки памяти);
 - использовать Debug Drawer и LeakCanary для диагностики проблем;
 - устранить выявленные баги и проверить корректность работы интерфейса;
 - составить отчёт о найденных проблемах и способах их решения.

Теоретические вопросы для устного опроса:

6. Что такое UI тестирование? Какие инструменты используются для UI тестирования в Android?
7. Как анализировать производительность интерфейса? Какие инструменты для этого существуют?

8. Что такое Layout Inspector? Как он помогает в отладке UI?
9. Для чего нужен LeakCanary? Как он обнаруживает утечки памяти?
10. Что такое Design Review? Как его проводить?
11. Как собирать и анализировать обратную связь от пользователей?
12. Какие особенности рендеринга интерфейса в РЕД ОС М?
13. Что такое инкрементальная сборка и modular UI? Как они влияют на производительность разработки?

Типовые вопросы для защиты Отчета по учебной практике УП.03.01

1. Какие принципы юзабилити вы учитывали при проектировании интерфейса? Приведите конкретные примеры из вашего проекта.
2. Как вы адаптировали интерфейс под разные плотности экранов (dpi, sp, dp)? Покажите макеты для разных разрешений.
3. Какие гайдлайны (Android, Aurora, РЕД ОС М) вы использовали? В чём их ключевые отличия?
4. Как вы обеспечивали доступность интерфейса (accessibility)? Какие настройки TalkBack, контрастности и размера текста вы реализовали?
5. Что такое дизайн-токены? Использовали ли вы их в своём проекте? Если да, то какие и для чего?
6. Опишите процесс переноса дизайна из Figma в код. С какими сложностями вы столкнулись?
7. Как вы проектировали интерфейс для смарт-часов? Какие ограничения учитывали?
8. Какую архитектуру (MVC, MVVM, MVI) вы использовали в проекте? Обоснуйте выбор.
9. Как вы организовали навигацию между экранами? Покажите фрагмент кода с настройкой Jetpack Navigation.
10. Как вы управляли состоянием в приложении? Использовали ли StateFlow, LiveData или remember {}? Почему?
11. Опишите процесс интеграции UI с данными из REST API. Как обрабатывали ошибки сети?
12. Как вы реализовали локализацию интерфейса? Покажите, как переключаются языки в вашем приложении.

13. Какие кастомные компоненты UI вы создали? Для чего они нужны? Приведите фрагмент кода.
14. Как вы использовали компонентный подход (фрагменты, ViewModels, Compose Composables) в своём проекте?
15. Какие UI тесты вы написали с использованием Espresso? Покажите пример теста и объясните, что он проверяет.
16. Как вы профилировали производительность интерфейса? Какие инструменты (Layout Inspector, Systrace) использовали? Какие проблемы выявили и как их решили?
17. Опишите процесс проведения A/B теста интерфейса. Какие метрики вы отслеживали? Какие выводы сделали?
18. Как вы внедрили систему обратной связи (например, Firebase Feedback)? Покажите интерфейс сбора отзывов.
19. Какие утечки памяти вы обнаружили с помощью LeakCanary? Как их устранили?
20. Что такое Design Review? Проводили ли вы его для своего проекта? Какие замечания получили?
21. С какими особенностями рендеринга интерфейса в РЕД ОС М вы столкнулись? Как их учли в разработке?
22. Какие UI тесты вы написали с использованием Espresso? Покажите пример теста и объясните, что он проверяет.
23. Как вы профилировали производительность интерфейса? Какие инструменты (Layout Inspector, Systrace) использовали? Какие проблемы выявили и как их решили?
24. Опишите процесс проведения A/B теста интерфейса. Какие метрики вы отслеживали? Какие выводы сделали?
25. Как вы внедрили систему обратной связи (например, Firebase Feedback)? Покажите интерфейс сбора отзывов.
26. Какие утечки памяти вы обнаружили с помощью LeakCanary? Как их устранили?
27. Что такое Design Review? Проводили ли вы его для своего проекта? Какие замечания получили?
28. С какими особенностями рендеринга интерфейса в РЕД ОС М вы столкнулись? Как их учли в разработке?

4 Методические указания для обучающихся по освоению дисциплины

Отчёт учебной практики УП.03.01. состоит из двух частей: основной и приложений.

Объём основной части отчёта составляет 13 – 16 страниц текста. Вторая часть отчёта о практике представляет собой приложения к отчёту (графики, схемы, диаграммы и таблицы, статистические данные, первичные документы, листинги т. п.).

Основная часть отчёта по учебной практике должна включать: титульный лист, содержание, основную часть и список использованной литературы.

Титульный лист — оформляется на специальном бланке, разработанном цикловой комиссией. Шрифт — Times New Roman, размер шрифта — 14 (пример оформления титульного листа — приложение А).

В содержании приводятся название разделов отчёта с обязательным указанием страниц.

По каждому разделу должны быть раскрыты теоретические вопросы и выполнены практические задания.

В конце отчёта приводится список использованной литературы.

Список использованной литературы должен включать источники и литературу, интернет-ресурсы, которые были использованы при выполнении учебной практики и оформлении отчёта.

К литературе относят: периодические и непериодические издания. Период издания литературы — 1–5 лет, статей — 1–2 года.

Для обеспечения наглядности в отчёте могут быть использованы графики, схемы и таблицы. Изложенный материал должен быть выстроен в логической последовательности.

4. Учебная литература и ресурсы информационно-телекоммуникационной сети

«Интернет»

Основная литература

1. Введение в разработку приложений для ОС Android : учебное пособие для СПО / Ю. В. Березовская, О. А. Юфрякова, В. Г. Вологодина [и др.]. — 2-е изд. — Саратов : Профобразование, 2024. — 427 с. — ISBN 978-5-4488-0993-4. — Текст : электронный // Электронный ресурс цифровой образовательной среды СПО PROФобразование : [сайт]. — URL: <https://profspo.ru/books/139746>

2. Семакова, А. Введение в разработку приложений для смартфонов на ОС Android : учебное пособие для СПО / А. Семакова. — 2-е изд. — Саратов : Профобразование, 2024. — 102 с. — ISBN 978-5-4488-0994-1. — Текст : электронный // Электронный ресурс

цифровой образовательной среды СПО PROФобразование : [сайт]. — URL: <https://profspo.ru/books/139747>

3. Нужный, А. М. Разработка мобильных приложений : учебное пособие для СПО / А. М. Нужный, Н. И. Гребенникова, В. В. Сафронов. — Саратов : Профобразование, 2022. — 92 с. — ISBN 978-5-4488-1494-5. — Текст : электронный // Электронный ресурс цифровой образовательной среды СПО PROФобразование : [сайт]. — URL: <https://profspo.ru/books/121301>

6 Материально-техническая база, информационные технологии, программное обеспечение и информационные справочные системы

Для реализации учебной практики необходимы:

- рабочие станции (персональные компьютеры) с характеристиками не ниже: процессор — Intel Core i5 (или аналогичный AMD), ОЗУ — 16 ГБ, SSD — не менее 256 ГБ;
- проекционное оборудование (проектор/интерактивная доска) для демонстрации материалов;
- сетевое подключение со скоростью не менее 100 Мбит/с.

Программное обеспечение

- Операционные системы: Windows 10/11, Linux (Ubuntu, CentOS, Astra, Alt)
- Офисные пакеты: Microsoft Office 365, LibreOffice
- Системы управления базами данных: PostgreSQL, MySQL/MariaDB, Microsoft SQL Server (Express-версия), MongoDB (для работы с NoSQL-данными).

Электронно-библиотечные системы (ЭБС)

1. ЭБС «BOOK.RU». — URL: <https://book.ru/>
2. ЭБС «Znanium». — URL: <https://znanium.ru/>
3. ЭБС «Айбукс». — URL: <https://ibooks.ru/>
4. ЭБС «Лань». — URL: <https://e.lanbook.com/>
5. ЭБС «Юрайт». — URL: <https://urait.ru/>
6. Электронные каталоги библиотеки СЗИУ РАНХиГС. — URL: <https://sziu-lib.ranepa.ru/>

Федеральное государственное бюджетное образовательное учреждение
высшего образования
**РОССИЙСКАЯ АКАДЕМИЯ НАРОДНОГО ХОЗЯЙСТВА и
ГОСУДАРСТВЕННОЙ СЛУЖБЫ при ПРЕЗИДЕНТЕ РОССИЙСКОЙ
ФЕДЕРАЦИИ**
СЕВЕРО-ЗАПАДНЫЙ ИНСТИТУТ УПРАВЛЕНИЯ

Факультет среднего профессионального образования

Специальность 09.02.11 Разработка и управление программным обеспечением

ОТЧЕТ
о прохождении учебной практики УП.03.01

по ПМ.03 Разработка приложений для мобильных платформ

Студента 3 курса группы _____
номер группы

Иванов Иван Иванович
(Ф.И.О. студента)

Наименование базы практики СЗИУ РАНХиГС (ФСПО)

Сроки прохождения практики: с «___» _____ 202__ г. по «___» _____ 202__ г.

Руководитель практики: _____ / П.П. Петров /
(подпись)

Обучающийся: _____ / И.И. Иванов /
(подпись)

Санкт-Петербург 202_ год