

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**



**Частное учреждение высшего образования
«Высшая школа предпринимательства (институт)»
(ЧУВО «ВШП»)**

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
Б1.В.03 «Разработка мобильных приложений»**

Направление подготовки: 09.03.02 Информационные системы и технологии

**Направленность (профиль) программы бакалавриата
«Информационные технологии»**

ОДОБРЕНО

Ученым советом ЧУВО «ВШП»

Протокол заседания

№01-02/24 от 15 мая 2025 г.

УТВЕРЖДАЮ

ЧУВО «ВШП»

» мая 2025 г.

ллабян М.Г.



Тверь, 2025

Рабочая программа учебной дисциплины **Б1.В.03 Разработка мобильных приложений**, компонента основной профессиональной образовательной программы высшего образования — программы бакалавриата по направлению подготовки **09.03.02 Информационные системы и технологии** направленность (профиль) **«Информационные технологии»**, направлена на обеспечение у обучающегося способности осуществлять профессиональную деятельность в соответствующей области и сферах профессиональной деятельности, в том числе на их практическую подготовку с учётом рабочей программы воспитания и календарного плана воспитательной работы Частном учреждении высшего образования **«Высшая школа предпринимательства (институт)»** (далее — **ЧУВО «ВШП»**).

1. ОБЛАСТЬ ПРИМЕНЕНИЯ И НОРМАТИВНЫЕ ССЫЛКИ

Настоящая рабочая программа учебной дисциплины устанавливает требования к результатам обучения студента и определяет содержание и виды учебных занятий и отчетности.

Программа предназначена для преподавателей и студентов направления подготовки 09.03.02 Информационные системы и технологии.

Программа учебной дисциплины разработана в соответствии с ФГОС ВО, утвержденного приказом Минобрнауки России от 19.09.2017 № 926 «Об утверждении федерального государственного образовательного стандарта высшего образования - бакалавриата по направлению подготовки 09.03.02 Информационные системы и технологии», основной профессиональной образовательной программой высшего образования по направлению подготовки 09.03.02 Информационные системы и технологии, направленность (профиль) Информационные технологии.

2. ЦЕЛИ И ЗАДАЧИ ДИСЦИПЛИНЫ

Целью изучения дисциплины «Разработка мобильных приложений» является формирование системы понятий, знаний, умений и навыков в области разработки приложений для мобильных устройств.

Для этого в рамках дисциплины решаются следующие задачи:

- изучение теоретических основ разработки приложений для мобильных устройств;
- формирование представлений о современных тенденциях в области информатики, связанных с использованием мобильных устройств.

3. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП ВО

Дисциплина является компонентом части, формируемой участниками образовательных отношений Блока 1 основной профессиональной образовательной программы высшего образования по направлению подготовки 09.03.02 Информационные системы и технологии, направленность (профиль) — Информационные технологии.

4. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ В РАМКАХ ПЛАНИРУЕМЫХ РЕЗУЛЬТАТОВ ОСВОЕНИЯ ОСНОВНОЙ ПРОФЕССИОНАЛЬНОЙ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Процесс освоения учебной дисциплины направлен на формирование у обучающихся следующих компетенций.

В результате освоения учебной дисциплины обучающийся должен демонстрировать следующие результаты обучения: УК-1, ПК-1, ПК-2, ПК-4.

Таблица 1. Результаты обучения

| Код компетенции | Наименование компетенции | Индекс и наименование индикатора содержания компетенции | Дескрипторы – основные признаки освоения (показатели достижения результата) |
|-----------------|--|---|--|
| УК-1 | Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач | УК-1.1 Способен осуществлять поиск и критический анализ информации | Знать: <ul style="list-style-type: none">- Основные источники информации по разработке мобильных приложений.- Методики критического анализа технической литературы и документации. Уметь: <ul style="list-style-type: none">- Искать и выбирать актуальные и надежные источники информации.- Анализировать и оценивать найденную информацию на предмет ее достоверности и полезности. Владеть: <ul style="list-style-type: none">- Навыками работы с технической документацией и справочными материалами.- Навыками эффективного поиска информации в интернете и специализированных базах данных. |

| | | | |
|------|--|--|---|
| | | УК-1.2 Способен применять системный подход для решения поставленных задач | <p>Знать:</p> <ul style="list-style-type: none"> - Основы системного подхода в программной инженерии. - Принципы анализа и декомпозиции сложных задач на составляющие компоненты. <p>Уметь:</p> <ul style="list-style-type: none"> - Применять системный подход при разработке мобильных приложений. - Декомпозировать задачи на подзадачи и распределять их по модулям. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками системного мышления и планирования разработки приложений. - Навыками интеграции различных компонентов и модулей в единое целое. |
| ПК-1 | Разработка и отладка программного кода | ПК-1.1 Способен разрабатывать программный код | <p>Знать:</p> <ul style="list-style-type: none"> - Основы синтаксиса и семантики языков программирования Java и Kotlin. - Принципы объектно-ориентированного программирования. <p>Уметь:</p> <ul style="list-style-type: none"> - Разрабатывать программный код для мобильных приложений на Android. - Использовать библиотеки и фреймворки для разработки мобильных приложений. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками эффективного написания и структурирования кода. - Навыками использования средств разработки и отладки кода (IDE, дебаггеры). |
| | | ПК-1.2 Способен отлаживать программный код | <p>Знать:</p> <ul style="list-style-type: none"> - Основные методы и инструменты отладки программного кода. - Принципы логирования и диагностики ошибок. <p>Уметь:</p> <ul style="list-style-type: none"> - Использовать отладочные инструменты для поиска и исправления ошибок в коде. - Проводить тестирование отдельных модулей и компонентов. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками логирования и анализа ошибок в приложениях. - Навыками работы с отладчиками и профайлерами для мобильных приложений. |
| ПК-2 | Проверка работоспособности и рефакторинг кода программного обеспечения | ПК-2.1 Способен проверять работоспособность программного кода | <p>Знать:</p> <ul style="list-style-type: none"> - Основы юнит-тестирования и интеграционного тестирования. - Принципы написания тестов и их автоматизации. <p>Уметь:</p> <ul style="list-style-type: none"> - Писать и выполнять юнит-тесты для проверки функциональности кода. - Проводить интеграционное тестирование для проверки взаимодействия модулей. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками использования тестовых фреймворков (JUnit, Espresso). - Навыками анализа результатов тестирования и устранения выявленных проблем. |

| | | | |
|------|---|--|---|
| | | ПК-2.2 Способен рефакторить программный код | Знать: <ul style="list-style-type: none"> - Принципы рефакторинга и его роль в поддержке качества кода. - Основные техники рефакторинга (удаление дублирующегося кода, улучшение имен переменных и функций). Уметь: <ul style="list-style-type: none"> - Применять техники рефакторинга для улучшения структуры и читаемости кода. - Оптимизировать код для повышения его производительности. Владеть: <ul style="list-style-type: none"> - Навыками анализа и оценки кода перед рефакторингом. - Навыками использования инструментов для автоматизированного рефакторинга. |
| ПК-4 | Разработка требований и проектирование программного обеспечения | ПК-4.1 Способен разрабатывать требования к программному обеспечению | Знать: <ul style="list-style-type: none"> - Основные виды и методы сбора требований к программному обеспечению. - Принципы документирования и управления требованиями. Уметь: <ul style="list-style-type: none"> - Формулировать функциональные и нефункциональные требования к мобильным приложениям. - Использовать методы анализа требований (интервью, анкеты, рабочие группы). Владеть: <ul style="list-style-type: none"> - Навыками составления и документирования требований. - Навыками работы с инструментами для управления требованиями (Jira, Confluence). |
| | | ПК-4.2 Способен проектировать программное обеспечение | Знать: <ul style="list-style-type: none"> - Принципы проектирования программного обеспечения и архитектуры приложений. - Основные шаблоны проектирования и их применение. Уметь: <ul style="list-style-type: none"> - Разрабатывать архитектуру мобильных приложений. - Применять шаблоны проектирования при разработке приложений. Владеть: <ul style="list-style-type: none"> - Навыками моделирования и документирования архитектуры приложений. - Навыками использования инструментов для проектирования программного обеспечения (UML, ER-диаграммы). |

5. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

5.1. Объем дисциплины в зачетных единицах с указанием количества академических часов

Общая трудоемкость учебной дисциплины составляет 6 зачетных единиц, 216 часов, включая все формы контактной и самостоятельной работы обучающихся.

Объем дисциплины по учебному плану составляет –
6 зачётных единицы = 216 академических часов.

Контактная работа обучающегося (студенты) с научно-педагогическим работником организации (всего) - 72 академических часов,

в том числе:

Лекционные занятия (Лек.) - 20 академических часов,

Практические занятия (Пр.) - 50 академических часов,

Консультации (Конс.) - 2 академических часа.

Самостоятельная работа обучающегося (студента):

Самостоятельная работа (СР) - 99 академический час,

Текущий контроль успеваемости

и промежуточно-заочная аттестация обучающегося (студента):

Часы на контроль - 45 академических часов.

Таблица 2. Объем дисциплины

| № п/п | Раздел/тема дисциплины | Виды учебной деятельности, включая самостоятельную работу обучающихся (студентов), и трудоёмкость (в ак. часах) | | | | Коды формируемых компетенций |
|--------------------|--|--|-----|-------|--------------------------------------|---|
| | | Виды учебных занятий по дисциплине | | | Самосто ятельна я работа СР | |
| | | Лек. | Пр. | Конс. | | |
| 5 семестр / 3курс | | | | | | |
| 1 | Тема 1. Введение в разработку мобильных приложений | 1 | 2 | - | 6 | УК-1.1, УК-1.2 |
| 2 | Тема 2. Основы Java и разработка на Android | 2 | 4 | - | 9 | ПК-1.1, ПК-1.2 |
| 3 | Тема 3. Интерфейс пользователя в Android | 1 | 4 | - | 9 | ПК-4.2 |
| 4 | Тема 4. Фрагменты и многомодульные приложения | 2 | 4 | - | 9 | ПК-4.2 |
| 5 | Тема 5. Работа с данными в Android | 2 | 4 | - | 9 | ПК-2.1, ПК-4.1 |
| 6 | Тема 6. Введение в Kotlin для Android | 2 | 4 | - | 9 | ПК-1.1, ПК-1.2 |
| 7 | Тема 7. Подготовка к зачету | | 4 | - | 12 | УК-1.1, УК-1.2, ПК-1.1, ПК-1.2, ПК-2.1, ПК-4.1, ПК-4.2 |
| 6 семестр / 3 курс | | | | | | |
| 8 | Тема 8. Работа с сетью и веб-сервисами | 2 | 2 | - | 4 | ПК-1.1, ПК-3.1 |
| 9 | Тема 9. Асинхронные операции в Android | 1 | 4 | - | 4 | ПК-2.1, ПК-2.2 |
| 10 | Тема 10. Работа с мультимедиа | 1 | 2 | - | 4 | ПК-1.1, ПК-1.2 |
| 11 | Тема 11. Расширенные возможности интерфейса пользователя | 1 | 2 | - | 4 | ПК-4.2 |
| 12 | Тема 12. Уведомления и фоновая работа | 1 | 2 | - | 4 | ПК-3.1 |

| | | | | | | |
|-----------------------------------|---|-------------------------|---|---|--------------------|---|
| 13 | Тема 13. Интеграция с внешними сервисами | 1 | 2 | - | 4 | ПК-3.1, ПК-3.2 |
| 14 | Тема 14. Тестирование и отладка мобильных приложений | 2 | 4 | - | 4 | ПК-2.1, ПК-2.2 |
| 15 | Тема 15 Публикация и распространение приложений | 1 | 2 | - | 4 | ПК-4.1, ПК-4.2 |
| 16 | Тема 16 Итоговое занятие и подготовка к экзамену | | 4 | 2 | 4 | УК-1.1, УК-1.2, ПК-1.1, ПК-1.2, ПК-2.1, ПК-2.2, ПК-3.1, ПК-3.2, ПК-4.1, ПК-4.2 |
| ИТОГО аудиторных часов/СР: | | 5-6 семестры/ 3 курс | 72 ак. часов | | 99 ак. часа | - |
| Часы на контроль | | 5-6 семестры/ 3 курс | 9 ак. час (форма промежуточной аттестации – зачет – 5 семестр) 36 ак. час (форма промежуточной аттестации – экзамен – 6 семестр) | | | |
| ВСЕГО ак. часов: | | 5-6 семестры/ 3 курс | 216 академических часов | | | |

5.2. Тематическое содержание дисциплины

* количество академических часов и виды занятий представлены в таблице № 2.

5 семестр

Тема 1: Введение в разработку мобильных приложений

Изучение истории и развития мобильных платформ, обзор современных мобильных операционных систем (Android и iOS), а также сравнение мобильных платформ с рассмотрением их особенностей и различий.

Тема 2: Основы Java и разработка на Android

Изучение основ языка программирования Java (синтаксис, типы данных, операторы, объектно-ориентированное программирование, работа с коллекциями и потоками), а затем применение этих знаний для разработки Android-приложений (установка и настройка среды разработки Android Studio, изучение структуры Android-приложения и основных компонентов, таких как Activity, Service, Broadcast Receiver, Content Provider).

Тема 3: Интерфейс пользователя в Android

Рассмотрение основных элементов интерфейса пользователя (Views и ViewGroups), работа с Layout'ами (LinearLayout, RelativeLayout, ConstraintLayout), создание и использование ресурсов (строки, изображения, стили).

Тема 4: Фрагменты и многомодульные приложения

Введение в фрагменты, изучение жизненного цикла фрагмента, использование фрагментов для создания адаптивных интерфейсов.

Тема 5: Работа с данными в Android

Использование SharedPreferences для хранения настроек, работа с SQLite (создание, чтение, обновление и удаление данных), введение в ORM для SQLite.

Тема 6: Введение в Kotlin для Android

Изучение основ Kotlin (синтаксис, типы данных, функции), совместимость Kotlin с Java, преимущества использования Kotlin для разработки Android-приложений.

Тема 7: Подготовка к зачету

Повторение пройденного материала, решение типовых задач и вопросов для подготовки к зачету.

6 семестр

Тема 8: Работа с сетью и веб-сервисами

Введение в сетевые технологии, работа с HTTP-запросами с использованием библиотеки Retrofit, обработка JSON-данных с помощью GSON и Moshi.

Тема 9: Асинхронные операции в Android

Изучение основ асинхронного программирования, использование AsyncTask, введение в RxJava (реактивное программирование).

Тема 10: Работа с мультимедиа

Воспроизведение аудио и видео, работа с камерой и галереей, запись аудио и видео.

Тема 11: Расширенные возможности интерфейса пользователя

Создание анимаций и переходов, создание пользовательских View, работа с темами и стилями.

Тема 12: Уведомления и фоновая работа

Создание и управление уведомлениями, использование сервисных компонентов, введение в WorkManager для выполнения фоновых задач.

Тема 13: Интеграция с внешними сервисами

Использование Google Play Services, интеграция с картами и геолокацией, введение в Firebase (аналитика, аутентификация, база данных в реальном времени).

Тема 14: Тестирование и отладка мобильных приложений

Изучение основ тестирования (юнит-тесты и инструментальные тесты), использование JUnit и Espresso, отладка приложений с помощью логирования и отладчика.

Тема 15: Публикация и распространение приложений

Подготовка приложения к публикации, создание подписанного APK, публикация приложения в Google Play Store.

Тема 16: Итоговое занятие и подготовка к экзамену

Повторение и обобщение пройденного материала, решение типовых задач и вопросов для подготовки к экзамену, обсуждение практических аспектов разработки и публикации мобильных приложений.

5.2.1 Содержание практических занятий

Таблица 3

| № п/п | Наименование темы (раздела) дисциплины | Содержание практического занятия |
|-----------|---|---|
| 5 семестр | | |
| 1 | Практическое занятие 1: Установка и настройка среды разработки Android Studio | Задание: Установить Android Studio, настроить эмулятор Android. Цель: Ознакомиться с интерфейсом Android Studio и основными инструментами разработки. |
| 2 | Практическое занятие 2: Основы Java и разработка на Android | Задание: Написать несколько программ на Java, используя основные синтаксические конструкции и ООП. Затем создать новое Android-приложение, изучить структуру проекта, создать и запустить Activity. |

| | | |
|------------------|--|--|
| | | Цель: Углубить знания в языке программирования Java и научиться применять их для создания базовых Android-приложений. |
| 3 | Практическое занятие 3: Разработка пользовательского интерфейса в Android | Задание: Создать интерфейс пользователя с использованием различных Layout'ов и View-элементов. Цель: Научиться создавать и настраивать интерфейс пользователя в Android-приложениях. |
| 4 | Практическое занятие 4: Работа с фрагментами | Задание: Создать приложение с использованием фрагментов, изучить их жизненный цикл. Цель: Освоить создание и использование фрагментов для разработки адаптивных интерфейсов. |
| 5 | Практическое занятие 5: Работа с данными в Android | Задание: Реализовать хранение данных с использованием SharedPreferences и SQLite. Цель: Научиться использовать механизмы хранения данных в Android. |
| 6 | Практическое занятие 6: Введение в Kotlin | Задание: Написать несколько программ на Kotlin, изучить его синтаксис и основные конструкции. Цель: Ознакомиться с языком программирования Kotlin и его применением для разработки Android-приложений. |
| 7 | Практическое занятие 7: Подготовка к зачету | Задание: Решение типовых задач и вопросов для закрепления пройденного материала. Цель: Повторение и обобщение изученного материала для успешной сдачи зачета. |
| 6 семестр | | |
| 8 | Практическое занятие 8: Работа с сетью и веб-сервисами | Задание: Создать приложение, выполняющее HTTP-запросы и обрабатывающее JSON-данные с использованием библиотеки Retrofit. Цель: Научиться работать с сетевыми запросами и веб-сервисами в Android-приложениях. |
| 9 | Практическое занятие 9: Асинхронные операции в Android | Задание: Реализовать асинхронные задачи с использованием AsyncTask и RxJava. Цель: Освоить методы асинхронного программирования в Android-приложениях. |
| 10 | Практическое занятие 10: Работа с мультимедиа | Задание: Создать приложение для воспроизведения аудио и видео, работы с камерой и галереей. Цель: Научиться работать с мультимедийным контентом в Android-приложениях. |
| 11 | Практическое занятие 11: Расширенные возможности интерфейса пользователя | Задание: Реализовать анимации и переходы, создать пользовательские View. Цель: Изучить методы создания сложных и анимированных интерфейсов в Android. |
| 12 | Практическое занятие 12: Уведомления и фоновая работа | Задание: Создать приложение с использованием уведомлений и выполнения фоновых задач с помощью WorkManager. Цель: Научиться использовать уведомления и фоновые задачи в Android-приложениях. |
| 13 | Практическое занятие 13: Интеграция с внешними сервисами | Задание: Реализовать интеграцию с Google Play Services, использование карт и геолокации, а также интеграцию с Firebase. Цель: Освоить методы интеграции мобильных приложений с внешними сервисами и API. |
| 14 | Практическое занятие 14: Тестирование и отладка мобильных приложений | Задание: Написать юнит-тесты и инструментальные тесты, отладить приложение с использованием логирования и отладчика. Цель: Научиться тестировать и отлаживать мобильные приложения для обеспечения их корректной работы. |
| 15 | Практическое занятие 15: Публикация и распространение приложений | Задание: Подготовить приложение к публикации, создать подписанный APK, опубликовать приложение в Google Play Store. Цель: Изучить процесс подготовки и публикации мобильных приложений. |
| 16 | Практическое занятие 16: Итоговое занятие и подготовка к экзамену | Задание: Решение типовых задач и вопросов для закрепления пройденного материала, обсуждение практических аспектов разработки и публикации мобильных приложений. Цель: Повторение и обобщение пройденного материала для успешной сдачи экзамена. |

5.2.2 Содержание самостоятельной работы

Таблица 4

| № п/п | Наименование темы (раздела) дисциплины | Содержание самостоятельной работы | Форма контроля |
|------------------|--|--|--|
| 5 семестр | | | |
| 1 | Самостоятельная работа 1: Изучение истории и развития мобильных платформ | Задание: Прочитать рекомендованные материалы по истории и развитию мобильных платформ, подготовить краткий обзор. Цель: Понять исторический контекст и эволюцию мобильных операционных систем. | Подготовить реферат |
| 2 | Самостоятельная работа 2: Основы Java и разработка на Android | Задание: Изучить материалы по основам языка программирования Java, выполнить дополнительные упражнения на Java, решая задачи из рекомендованных учебников. Затем самостоятельно изучить структуру типового Android-приложения и подготовить краткий отчет. Цель: Освоить основы языка программирования Java и понять, как применять эти знания для разработки Android-приложений. | Ответы на контрольные вопросы, выполнение практического задания. |
| 3 | Самостоятельная работа 3: Создание макетов интерфейса пользователя | Задание: Разработать макеты интерфейса пользователя для нескольких экранов Android-приложения. Цель: Научиться проектировать интерфейс пользователя с использованием различных Layout'ов и View-элементов. | Выполнение практического задания. |
| 4 | Самостоятельная работа 4: Работа с фрагментами и многомодульными приложениями | Задание: Изучить примеры использования фрагментов, реализовать фрагменты для своего проекта. Цель: Освоить создание и использование фрагментов для разработки адаптивных интерфейсов. | Выполнение практического задания. |
| 5 | Самостоятельная работа 5: Изучение механизмов хранения данных в Android | Задание: Прочитать материалы по SharedPreferences и SQLite, подготовить обзор и пример использования. Цель: Научиться использовать механизмы хранения данных в Android. | Выполнение практического задания. |
| 6 | Самостоятельная работа 6: Введение в Kotlin | Задание: Изучить материалы по языку программирования Kotlin, решить задачи на синтаксис и основные конструкции. Цель: Ознакомиться с языком программирования Kotlin и его применением для разработки Android-приложений. | Выполнение практического задания. |
| 7 | Самостоятельная работа 7: Подготовка к зачету | Задание: Решение типовых задач и вопросов для закрепления пройденного материала. Цель: Повторение и обобщение изученного материала для успешной сдачи зачета. | Тест. |
| 6 семестр | | | |
| 8 | Самостоятельная работа 8: Работа с сетью и веб-сервисами | Задание: Прочитать материалы по работе с HTTP-запросами и JSON-данными, выполнить упражнения с использованием библиотеки Retrofit. Цель: Научиться работать с сетевыми запросами и веб-сервисами в Android-приложениях. | Выполнение практического задания. |
| 9 | Самостоятельная работа 9: Изучение асинхронного программирования | Задание: Изучить материалы по асинхронному программированию в Android, решить задачи с использованием AsyncTask и RxJava. Цель: Освоить методы асинхронного программирования в Android-приложениях. | Выполнение практического задания. |
| 10 | Самостоятельная работа 10: Работа с мультимедийными данными | Задание: Прочитать материалы по работе с аудио и видео, выполнить упражнения по воспроизведению и записи мультимедийного контента. Цель: Научиться работать с мультимедийным контентом в Android-приложениях. | Выполнение практического задания. |
| 11 | Самостоятельная работа 11: Изучение создания сложных интерфейсов | Задание: Изучить материалы по созданию анимаций и пользовательских View, реализовать упражнения на эту тему. Цель: Освоить методы создания сложных и | Выполнение практического задания. |

| | | | |
|----|---|--|-----------------------------------|
| | | анимированных интерфейсов в Android. | |
| 12 | Самостоятельная работа 12: Уведомления и фоновая работа | Задание: Изучить материалы по созданию уведомлений и фоновых задач, выполнить упражнения с использованием WorkManager. Цель: Научиться использовать уведомления и фоновые задачи в Android-приложениях. | Выполнение практического задания. |
| 13 | Самостоятельная работа 13: Интеграция с внешними сервисами | Задание: Изучить материалы по интеграции с Google Play Services и Firebase, выполнить упражнения по интеграции карт и геолокации. Цель: Освоить методы интеграции мобильных приложений с внешними сервисами и API. | Выполнение практического задания. |
| 14 | Самостоятельная работа 14: Тестирование и отладка мобильных приложений | Задание: Прочитать материалы по тестированию и отладке приложений, выполнить упражнения по написанию юнит-тестов и инструментальных тестов. Цель: Научиться тестировать и отлаживать мобильные приложения для обеспечения их корректной работы. | Выполнение практического задания. |
| 15 | Самостоятельная работа 15: Публикация и распространение приложений | Задание: Изучить процесс подготовки и публикации мобильных приложений, подготовить свое приложение к публикации. Цель: Изучить процесс подготовки и публикации мобильных приложений. | Выполнение практического задания. |
| 16 | Самостоятельная работа 16: Итоговое занятие и подготовка к экзамену | Задание: Решение типовых задач и вопросов для закрепления пройденного материала, обсуждение практических аспектов разработки и публикации мобильных приложений. Цель: Повторение и обобщение пройденного материала для успешной сдачи экзамена. | Экзамен |

6.Оценочные материалы по дисциплине

Оценочные материалы находятся в документе «Оценочные материалы по дисциплине «Разработка мобильных приложений».

7.Методические материалы для обучающихся по освоению дисциплины (модуля)

А) Рекомендации обучающемуся (студенту) по работе с конспектом после лекции

Какими бы замечательными качествами в области методики ни обладал лектор, какое бы большое значение на занятиях ни уделял лекции слушатель, глубокое понимание материала достигается только путем самостоятельной работы над ним. Самостоятельную работу следует начинать с доработки конспекта, желательно в тот же день, пока время не стерло содержание лекции из памяти (через 10 часов после лекции в памяти остается не более 30-40 % материала). С целью доработки необходимо в первую очередь прочитать записи, восстановить текст в памяти, а также исправить описки, расшифровать не принятые ранее сокращения, заполнить пропущенные места, понять текст, вникнуть в его смысл. Далее прочитать материал по рекомендуемой литературе, разрешая в ходе чтения, возникшие ранее затруднения, вопросы, а также дополнения и исправляя свои записи. Записи должны быть наглядными, для чего следует применять различные способы выделений. В ходе доработки конспекта углубляются, расширяются и закрепляются знания, а также дополняется, исправляется и совершенствуется конспект. Подготовленный конспект и рекомендуемая литература используется при подготовке к практическому занятию. Подготовка сводится к внимательному прочтению учебного материала, к выводу с карандашом в руках всех утверждений и формул, к решению примеров, задач, к ответам на вопросы, предложенные в конце лекции преподавателем или помещенные в рекомендуемой литературе. Примеры, задачи, вопросы по теме являются материалом самоконтроля. Непременным условием глубокого усвоения учебного материала является знание основ, на которых строится изложение материала. Обычно преподаватель напоминает, какой ранее изученный материал и в какой степени требуется подготовить к очередному занятию. Эта рекомендация, как и требование систематической и серьезной работы над всем лекционным курсом, подлежит безусловному выполнению. Потери логической связи как внутри темы, так и

между ними приводит к негативным последствиям: материал учебной дисциплины перестает основательно восприниматься, а творческий труд подменяется утомленным переписыванием. Обращение к ранее изученному материалу не только помогает восстановить в памяти известные положения, выводы, но и приводит разрозненные знания в систему, углубляет и расширяет их. Каждый возврат к старому материалу позволяет найти в нем что-то новое, переосмыслить его с иных позиций, определить для него наиболее подходящее место в уже имеющейся системе знаний. Неоднократное обращение к пройденному материалу является наиболее рациональной формой приобретения и закрепления знаний. Очень полезным, но, к сожалению, еще мало используемым в практике самостоятельной работы, является предварительное ознакомление с учебным материалом. Даже краткое, беглое знакомство с материалом очередной лекции дает многое. Обучающиеся (студенты) получают общее представление о её содержании и структуре, о главных и второстепенных вопросах, о терминах и определениях. Все это облегчает работу на лекции и делает ее целеустремленной.

Б) Рекомендации обучающемуся (студенту) по подготовке к занятиям семинарского типа

Обучающийся (студент) должен чётко уяснить, что именно с лекции начинается его подготовка к лабораторному/ практическому/ семинарскому/ методическому/ клиническому практическому занятию. Вместе с тем, лекция лишь организует мыслительную деятельность, но не обеспечивает глубину усвоения программного материала. При подготовке к такому виду занятий можно выделить 2 этапа:

- 1-й - организационный,
- 2-й - закрепление и углубление теоретических знаний.

На первом этапе обучающийся (студент) планирует свою самостоятельную работу, которая включает:

- уяснение задания на самостоятельную работу;
- подбор рекомендованной литературы;
- составление плана работы, в котором определяются основные пункты предстоящей подготовки.

Составление плана дисциплинирует и повышает организованность в работе. Второй этап включает непосредственную подготовку обучающегося (студента) к занятию. Начинать надо с изучения рекомендованной литературы. Необходимо помнить, что на лекции обычно рассматривается не весь материал, а только его часть. Остальная его часть восполняется в процессе самостоятельной работы. В связи с этим работа с рекомендованной литературой обязательна. Особое внимание при этом необходимо обратить на содержание основных положений и выводов, объяснение явлений и фактов, уяснение практического приложения рассматриваемых теоретических вопросов. В процессе этой работы обучающийся (студент) должен стремиться понять и запомнить основные положения рассматриваемого материала, примеры, поясняющие его, а также разобраться в иллюстративном материале. Заканчивать подготовку следует составлением плана (перечня основных пунктов) по изучаемому материалу (вопросу). Такой план позволяет составить концентрированное, сжатое представление по изучаемым вопросам. В процессе подготовки к семинарскому занятию рекомендуется взаимное обсуждение материала, во время которого закрепляются знания, а также приобретает практика в изложении и разъяснении полученных знаний, развивается речь. При необходимости следует обращаться за консультацией к преподавателю. Идя на консультацию, необходимо хорошо продумать вопросы, которые требуют разъяснения. В начале семинарского занятия обучающиеся (студента) под руководством преподавателя более глубоко осмысливают теоретические положения по теме занятия, раскрывают и объясняют основные явления и факты. В процессе творческого обсуждения и дискуссии вырабатываются умения и навыки использовать приобретенные знания для решения практических задач.

В) Рекомендации по самостоятельной работе обучающегося (студента) над изучаемым материалом

Успешное освоение данного курса базируется на рациональном сочетании нескольких видов учебной деятельности - лекций, семинарских занятий, самостоятельной работы. При этом самостоятельную работу следует рассматривать одним из главных звеньев полноценного высшего образования, на которую отводится значительная часть учебного времени.

Самостоятельная работа студентов складывается из следующих составляющих:

- работа с основной и дополнительной литературой, с материалами интернета и конспектами лекций;
- внеаудиторная подготовка к контрольным работам, выполнение докладов, рефератов и курсовых работ;
- выполнение самостоятельных практических работ;
- подготовка к экзаменам (зачетам) непосредственно перед ними.

Для правильной организации работы необходимо учитывать порядок изучения разделов курса, находящихся в строгой логической последовательности. Поэтому хорошее усвоение одной части дисциплины является предпосылкой для успешного перехода к следующей. Задания, проблемные вопросы, предложенные для изучения дисциплины, в том числе и для самостоятельного выполнения, носят междисциплинарный характер и базируются, прежде всего, на причинно-следственных связях между компонентами окружающего нас мира. В течение семестра, необходимо подготовить рефераты (проекты) с использованием рекомендуемой основной и дополнительной литературы и сдать рефераты для проверки преподавателю. Важным составляющим в изучении данного курса является решение ситуационных задач и работа над проблемно-аналитическими заданиями, что предполагает знание соответствующей научной терминологии и т.д.

Для лучшего запоминания материала целесообразно использовать индивидуальные особенности и разные виды памяти: зрительную, слуховую, ассоциативную. Успешному запоминанию также способствует приведение ярких свидетельств и наглядных примеров. Учебный материал должен постоянно повторяться и закрепляться.

При выполнении докладов, творческих, информационных, исследовательских проектов особое внимание следует обращать на подбор источников информации и методику работы с ними.

Для успешной сдачи экзамена (зачета) рекомендуется соблюдать следующие правила:

1. Подготовка к экзамену (зачету) должна проводиться систематически, в течение всего семестра.
2. Интенсивная подготовка должна начаться не позднее, чем за месяц до экзамена.
3. Время непосредственно перед экзаменом (зачетом) лучше использовать таким образом, чтобы оставить последний день свободным для повторения курса в целом, для систематизации материала и доработки отдельных вопросов.

На экзамене высокую оценку получают студенты, использующие данные, полученные в процессе выполнения самостоятельных работ, а также использующие собственные выводы на основе изученного материала.

Учитывая значительный объем теоретического материала, студентам рекомендуется регулярное посещение и подробное конспектирование лекций.

8. ОСОБЕННОСТИ РЕАЛИЗАЦИИ ДИСЦИПЛИНЫ ДЛЯ ИНВАЛИДОВ И ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ

Обучающимся с ограниченными возможностями здоровья предоставляются специальные учебники и учебные пособия, иная учебная литература, специальные технические средства обучения коллективного и индивидуального пользования, предоставление услуг ассистента (помощника), оказывающего обучающимся необходимую техническую помощь, а также услуги сурдопереводчиков и тифлосурдопереводчиков.

а) для слабовидящих:

- на промежуточной аттестации присутствует ассистент, оказывающий студенту необходимую техническую помощь с учетом индивидуальных особенностей (он помогает занять

рабочее место, передвигаться, прочитать и оформить задание, в том числе записывая под диктовку);

- задания для выполнения, а также инструкция о порядке проведения промежуточной аттестации оформляются увеличенным шрифтом;

- задания для выполнения на промежуточной аттестации зачитываются ассистентом;

- письменные задания выполняются на бумаге, надиктовываются ассистенту;

- обеспечивается индивидуальное равномерное освещение не менее 300 люкс;

- студенту для выполнения задания при необходимости предоставляется увеличивающее устройство;

в) для глухих и слабослышащих:

- на промежуточной аттестации присутствует ассистент, оказывающий студенту необходимую техническую помощь с учетом индивидуальных особенностей (он помогает занять рабочее место, передвигаться, прочитать и оформить задание, в том числе записывая под диктовку);

- промежуточно-заочная аттестация проводится в письменной форме;

- обеспечивается наличие звукоусиливающей аппаратуры коллективного пользования, при необходимости поступающим предоставляется звукоусиливающая аппаратура индивидуального пользования;

- по желанию студента промежуточно-заочная аттестация может проводиться в письменной форме;

д) для лиц с нарушениями опорно-двигательного аппарата (тяжелыми нарушениями двигательных функций верхних конечностей или отсутствием верхних конечностей):

- письменные задания выполняются на компьютере со специализированным программным обеспечением или надиктовываются ассистенту;

- по желанию студента промежуточно-заочная аттестация проводится в устной форме.

Примечание:

а) Для обучающегося (бакалавра), осваивающего учебную дисциплину, обязательный компонент основной профессиональной образовательной программы высшего образования — программы бакалавриата по направлению подготовки **09.03.02 Информационные системы и технологии (направленность (профиль) «Информационные технологии»), форма обучения — очно-заочная), одобренной на заседании Учёного совета образовательной организации, утверждённой ректором Частного образовательного учреждения высшего образования «Высшая школа предпринимательства», по индивидуальному учебному плану (при наличии факта зачисления в образовательную организацию такого обучающегося (бакалавра)), Институт:**

- разрабатывает, согласовывает с участниками образовательных отношений и утверждает в установленном порядке согласно соответствующему локальному нормативному акту **индивидуальный учебный план** конкретного обучающегося (бакалавра) (*учебный план, обеспечивающий освоение конкретной основной образовательной программы высшего образования на основе индивидуализации её содержания с учётом особенностей и образовательных потребностей конкретного обучающегося (бакалавра)*);

- устанавливает для конкретного обучающегося (бакалавра) по индивидуальному учебному плану **одинаковые дидактические единицы** — элементы содержания учебного материала, изложенного в виде утверждённой в установленном образовательной организацией порядке согласно соответствующему локальному нормативному акту рабочей программы учебной дисциплины, обязательного компонента разработанной и реализуемой Институтом основной профессиональной образовательной программы высшего образования — программы бакалавриата по направлению подготовки **09.03.02 Информационные системы и технологии** (направленность (профиль) «Информационные технологии»), форма обучения — очно-заочная), как и для обучающегося (бакалавра), осваивающего основную образовательную программу высшего образования в учебной группе;

- определяет в индивидуальном учебном плане конкретного обучающегося (бакалавра) **объём учебной дисциплины** с указанием количества академических часов/ ЗЕТ, выделенных на

его контактную работу (групповую и (или) индивидуальную работу) с руководящими и (или) научно-педагогическими работниками, реализующими основную образовательную программу высшего образования;

- определяет в индивидуальном учебном плане конкретного обучающегося (бакалавра) количество академических часов/ ЗЕТ по учебной дисциплине, выделенных на его самостоятельную работу (*при необходимости*).

б) Для обучающегося (бакалавра) с ограниченными возможностями здоровья и инвалида, осваивающего учебную дисциплину, обязательный компонент основной профессиональной образовательной программы высшего образования — программы бакалавриата по направлению подготовки **09.03.02 Информационные системы и технологии** (направленность (профиль) «Информационные технологии»), форма обучения — очно-заочная), одобренной на заседании Учёного совета образовательной организации, утверждённой ректором Частного образовательного учреждения высшего образования «Высшая школа предпринимательства», (*при наличии факта зачисления в образовательную организацию такого обучающегося (бакалавра) с учётом конкретной (конкретных) нозологии (нозологий)*), **Институт:**

- разрабатывает, согласовывает с участниками образовательных отношений и утверждает в установленном порядке соответствующему локальному нормативному акту **индивидуальный учебный план** конкретного обучающегося (бакалавра) с ограниченными возможностями здоровья/ инвалида (*при наличии факта зачисления в образовательную организацию такого обучающегося (бакалавра) с учётом конкретной (конкретных) нозологии (нозологий)*) (учебный план, обеспечивающий освоение конкретной основной образовательной программы высшего образования на основе индивидуализации её содержания с учётом особенностей и образовательных потребностей конкретного обучающегося (бакалавра));

- устанавливает для конкретного обучающегося (бакалавра) с ограниченными возможностями здоровья содержание образования (**одинаковые дидактические единицы** — элементы содержания учебного материала, как и для обучающего (бакалавра), осваивающего основную образовательную программу высшего образования в учебной группе) и условия организации обучения, изложенного в виде утверждённой в установленном Институте порядке соответствующему локальному нормативному акту рабочей программы учебной дисциплины, обязательного компонента разработанной и реализуемой им адаптированной основной профессиональной образовательной программы высшего образования - программы бакалавриата по направлению подготовки **09.03.02 Информационные системы и технологии** (направленность (профиль) «Информационные технологии»), форма обучения — очно-заочная), а для инвалидов также в соответствии с индивидуальной программой реабилитации инвалида (для конкретного обучающегося (бакалавра) с ограниченными возможностями здоровья/ инвалида (*при наличии факта зачисления в образовательную организацию такого обучающегося (бакалавра) с учётом конкретной (конкретных) нозологии (нозологий)*);

- определяет в индивидуальном учебном плане конкретного обучающегося бакалавра) с ограниченными возможностями здоровья/ инвалида (*при наличии факта зачисления такого обучающегося (бакалавра) с учётом конкретной (конкретных) нозологии (нозологий)*) **объём учебной дисциплины** с указанием количества академических часов/ ЗЕТ, выделенных на его контактную работу (групповую и (или) индивидуальную работу) с руководящими и (или) научно-педагогическими работниками, реализующими основную образовательную программу высшего образования;

- определяет в индивидуальном учебном плане конкретного обучающегося (бакалавра) с ограниченными возможностями здоровья/ инвалида (*при наличии факта зачисления в образовательную организацию такого обучающегося (бакалавра) с учётом конкретной (конкретных) нозологии (нозологий)*) количество академических часов/ ЗЕТ по учебной дисциплине, выделенных на его самостоятельную работу (*при необходимости*).

9. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

9.1 Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины

Основная литература:

1. Коротеев М.В., Введение в Android разработку на Java [Электронный ресурс] : учебное пособие / М.В. Коротеев, А.Ю. Шаталова. - М. : КноРус, 2022. - 231 с. - ISBN 978-5-406-11058-4. - Режим доступа: <https://book.ru/book/947344>
2. Гордиенко А.П., Языки программирования и методы трансляции [Электронный ресурс] : учебник / А. П. Гордиенко. - М. : КноРус, 2024. - 374 с. - ISBN 978-5-406-12853-4. - Режим доступа: <https://book.ru/book/953498>

Дополнительная литература:

1. Виноградов, Д. В. Разработка мобильных приложений и облачные сервисы : учеб. пособие / Д. В. Виноградов ; Владим. гос. ун-т им. А. Г. и Н. Г. Столетовых. – Владимир : Изд-во ВлГУ, 2022. – 235 с. – ISBN 978-5-9984-1677-4
2. Евсеев Д.А., Web-дизайн в примерах и задачах [Электронный ресурс] : учебное пособие / Д.А. Евсеев, В.В. Трофимов. - М. : КноРус, 2022. - 263 с. - ISBN 978-5-406-09190-6. - Режим доступа: <https://book.ru/book/942676>

9.2 Используемое программное обеспечение (комплект лицензионного и свободно распространяемого программного обеспечения, в том числе отечественного производства в соответствии с п.4.3.2. ФГОС ВО 09.03.02):

1. Microsoft Windows 11 Pro или аналогичная ОС, включая дистрибутивы Linux, например Debian, Ubuntu, OpenSuse, в том числе отечественного производства, например ОС Astra Linux Common Edition (Разработчик: АО «НПО РусБИТех»), ОС «РОСА» (Разработчик: «НТИЦ ИТ РОСА»).
2. Microsoft Office 365 или аналогичный офисный пакет, например OpenOffice, LibreOffice, ONLYOFFICE, в том числе отечественного производства, например МойОфис (Разработчик: ООО «НОВЫЕ ОБЛАЧНЫЕ ТЕХНОЛОГИИ»).
3. Adobe Reader или аналогичный просмотрщик PDF, например Okular, Foxit Reader, в том числе отечественного производства, например Окуляр ГОСТ (Разработчик: ООО «Лаборатория 50»).
4. Google Chrome или аналогичный веб-браузер, например Microsoft Edge, Mozilla Firefox, в том числе отечественного производства, например Яндекс.Браузер (Разработчик: ООО «ЯНДЕКС»).
5. Microsoft Visual Studio Code или аналогичная IDE, например Sublime Text, Eclipse, в том числе отечественного производства
6. PyCharm / IntelliJ IDEA / CLion / WebStorm либо аналогичная IDE полного стека, в том числе отечественного производства
7. MySQL CE 8.0 / MySQL Workbench или аналогичные СУБД, например MS SQL, PostgreSQL, в том числе отечественного производства
8. Android Studio или аналогичная IDE для разработки мобильных приложений, в том числе отечественного производства
9. Figma или аналогичное ПО для подготовки макетов, например Penpot, Lunacy, в том числе отечественного производства.

9.3 Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины (модуля) (в соответствии с п.4.3.4. ФГОС ВО 09.03.02)

1. Электронно-библиотечная система BOOK.RU [Электронный ресурс]. - Режим доступа: <https://book.ru/>

9.4 Базы данных, информационно-справочные и поисковые системы (в соответствии с п.4.3.4. ФГОС ВО 09.03.02)

1. КонсультантПлюс: справочно-поисковая система [Электронный ресурс]. - <http://www.consultant.ru>
2. Мировая цифровая библиотека: <http://wdl.org/ru>
3. Научная электронная библиотека «Scopus»: <https://www.scopus.com>
4. Научная электронная библиотека ScienceDirect: <http://www.sciencedirect.com>
5. Научная электронная библиотека «eLIBRARY»: <https://elibrary.ru>
6. Портал «Гуманитарное образование» <http://www.humanities.edu.ru>
7. Федеральный портал «Российское образование» <http://www.edu.ru>
8. Федеральное хранилище «Единая коллекция цифровых образовательных ресурсов» <http://school-collection.edu.ru>
9. Поисковые системы Yandex, Rambler и др.
10. Электронная библиотека Российской Государственной Библиотеки (РГБ): <http://elibrary.rsl.ru>
11. Электронно-библиотечная система <http://www.sciteclibrary.ru>

10. Материально-техническое обеспечение дисциплины

Помещения для проведения всех видов работы, предусмотренных учебным планом, укомплектованы необходимым оборудованием и техническими средствами обучения.

| Наименование помещений для проведения всех видов учебной деятельности, предусмотренной учебным планом, в том числе помещения для самостоятельной работы, с указанием перечня основного оборудования, учебно-наглядных пособий и используемого программного обеспечения | Адрес (местоположение) помещений для проведения всех видов учебной деятельности, предусмотренной учебным планом (в случае реализации образовательной программы в сетевой форме дополнительно указывается наименование организации, с которой заключен договор) |
|--|--|
| <p>Специализированная многофункциональная учебная аудитория для проведения учебных занятий лекционного типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, в том числе, для организации практической подготовки обучающихся, с перечнем основного оборудования:</p> <ul style="list-style-type: none"> - Столы для обучающихся; - Стулья для обучающихся; - Стол педагогического работника; - Стул педагогического работника; - Компьютеры с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду лицензиата; - Маркерная или меловая доска; - Проектор. | <p>170001, Россия, город Тверь, улица Спартак, дом 26а</p> |
| <p>Специализированная многофункциональная учебная аудитория для проведения учебных занятий семинарского типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, в том числе, для организации практической</p> | <p>170001, Россия, город Тверь, улица Спартак, дом 26а</p> |

| | |
|--|---|
| <p>подготовки обучающихся, с перечнем основного оборудования:</p> <ul style="list-style-type: none"> - Столы для обучающихся; - Стулья для обучающихся; - Стол педагогического работника; - Стул педагогического работника; - Компьютеры с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду лицензиата; - Маркерная или меловая доска; - Проектор. | |
| <p>Помещение для самостоятельной работы обучающихся с перечнем основного оборудования:</p> <ul style="list-style-type: none"> - Столы для обучающихся; - Стулья для обучающихся; - Стол педагогического работника; - Стул педагогического работника; - Компьютеры с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду лицензиата; - Маркерная или меловая доска; - Проектор. | <p>170001, Россия, город Тверь, улица Спартака, дом 26а</p> |
| <p>Помещение для практических занятий на персональных компьютерах:</p> <ul style="list-style-type: none"> - Столы для обучающихся; - Стулья для обучающихся; - Стол педагогического работника; - Стул педагогического работника; - Компьютеры с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду лицензиата; - Ноутбуки с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду лицензиата; - Маркерная или меловая доска; - Проектор. | <p>170001, Россия, город Тверь, улица Спартака, дом 26а</p> |

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**



**Частное учреждение высшего образования
«Высшая школа предпринимательства (институт)»
(ЧУВО «ВШП»)**

**ОЦЕНОЧНЫЕ МАТЕРИАЛЫ
по дисциплине
Б1.В.03 «Разработка мобильных приложений»**

**Направление подготовки: 09.03.02 Информационные системы и технологии
Направленность (профиль) программы бакалавриата
«Информационные технологии»**

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ В РАМКАХ ПЛАНИРУЕМЫХ РЕЗУЛЬТАТОВ ОСВОЕНИЯ ОСНОВНОЙ ПРОФЕССИОНАЛЬНОЙ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Процесс освоения учебной дисциплины направлен на формирование у обучающихся следующих компетенций.

В результате освоения учебной дисциплины обучающийся должен демонстрировать следующие результаты обучения: УК-1, ПК-1, ПК-2, ПК-4.

| Код компетенции | Наименование компетенции | Индекс и наименование индикатора содержания компетенции | Дескрипторы – основные признаки освоения (показатели достижения результата) |
|-----------------|--|--|--|
| УК-1 | Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач | УК-1.1 Способен осуществлять поиск и критический анализ информации | Знать: <ul style="list-style-type: none"> - Основные источники информации по разработке мобильных приложений. - Методики критического анализа технической литературы и документации. Уметь: <ul style="list-style-type: none"> - Искать и выбирать актуальные и надежные источники информации. - Анализировать и оценивать найденную информацию на предмет ее достоверности и полезности. Владеть: <ul style="list-style-type: none"> - Навыками работы с технической документацией и справочными материалами. - Навыками эффективного поиска информации в интернете и специализированных базах данных. |
| | | УК-1.2 Способен применять системный подход для решения поставленных задач | Знать: <ul style="list-style-type: none"> - Основы системного подхода в программной инженерии. - Принципы анализа и декомпозиции сложных задач на составляющие компоненты. Уметь: <ul style="list-style-type: none"> - Применять системный подход при разработке мобильных приложений. - Декомпозировать задачи на подзадачи и распределять их по модулям. Владеть: <ul style="list-style-type: none"> - Навыками системного мышления и планирования разработки приложений. - Навыками интеграции различных компонентов и модулей в единое целое. |
| ПК-1 | Разработка и отладка программного кода | ПК-1.1 Способен разрабатывать программный код | Знать: <ul style="list-style-type: none"> - Основы синтаксиса и семантики языков программирования Java и Kotlin. - Принципы объектно-ориентированного программирования. Уметь: <ul style="list-style-type: none"> - Разрабатывать программный код для мобильных приложений на Android. - Использовать библиотеки и фреймворки для разработки мобильных приложений. Владеть: |

| | | | |
|------|--|--|--|
| | | | <ul style="list-style-type: none"> - Навыками эффективного написания и структурирования кода. - Навыками использования средств разработки и отладки кода (IDE, дебаггеры). |
| | | ПК-1.2 Способен отлаживать программный код | <p>Знать:</p> <ul style="list-style-type: none"> - Основные методы и инструменты отладки программного кода. - Принципы логирования и диагностики ошибок. <p>Уметь:</p> <ul style="list-style-type: none"> - Использовать отладочные инструменты для поиска и исправления ошибок в коде. - Проводить тестирование отдельных модулей и компонентов. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками логирования и анализа ошибок в приложениях. - Навыками работы с отладчиками и профайлерами для мобильных приложений. |
| ПК-2 | Проверка работоспособности и рефакторинг кода программного обеспечения | ПК-2.1 Способен проверять работоспособность программного кода | <p>Знать:</p> <ul style="list-style-type: none"> - Основы юнит-тестирования и интеграционного тестирования. - Принципы написания тестов и их автоматизации. <p>Уметь:</p> <ul style="list-style-type: none"> - Писать и выполнять юнит-тесты для проверки функциональности кода. - Проводить интеграционное тестирование для проверки взаимодействия модулей. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками использования тестовых фреймворков (JUnit, Espresso). - Навыками анализа результатов тестирования и устранения выявленных проблем. |
| | | ПК-2.2 Способен рефакторить программный код | <p>Знать:</p> <ul style="list-style-type: none"> - Принципы рефакторинга и его роль в поддержке качества кода. - Основные техники рефакторинга (удаление дублирующегося кода, улучшение имен переменных и функций). <p>Уметь:</p> <ul style="list-style-type: none"> - Применять техники рефакторинга для улучшения структуры и читаемости кода. - Оптимизировать код для повышения его производительности. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками анализа и оценки кода перед рефакторингом. - Навыками использования инструментов для автоматизированного рефакторинга. |

| | | | |
|------|---|--|--|
| ПК-4 | Разработка требований и проектирование программного обеспечения | ПК-4.1 Способен разрабатывать требования к программному обеспечению | <p>Знать:</p> <ul style="list-style-type: none"> - Основные виды и методы сбора требований к программному обеспечению. - Принципы документирования и управления требованиями. <p>Уметь:</p> <ul style="list-style-type: none"> - Формулировать функциональные и нефункциональные требования к мобильным приложениям. - Использовать методы анализа требований (интервью, анкеты, рабочие группы). <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками составления и документирования требований. - Навыками работы с инструментами для управления требованиями (Jira, Confluence). |
| | | ПК-4.2 Способен проектировать программное обеспечение | <p>Знать:</p> <ul style="list-style-type: none"> - Принципы проектирования программного обеспечения и архитектуры приложений. - Основные шаблоны проектирования и их применение. <p>Уметь:</p> <ul style="list-style-type: none"> - Разрабатывать архитектуру мобильных приложений. - Применять шаблоны проектирования при разработке приложений. <p>Владеть:</p> <ul style="list-style-type: none"> - Навыками моделирования и документирования архитектуры приложений. - Навыками использования инструментов для проектирования программного обеспечения (UML, ER-диаграммы). |

КРИТЕРИИ ОЦЕНИВАНИЯ КОМПЕТЕНЦИЙ

(признак, на основании которого, проводится оценка по выбранному показателю)

| Показатель оценивания компетенций | Результат обучения | Критерии оценивания компетенций |
|--|--------------------|---|
| Высокий уровень (отлично) | Знать | Обучающийся продемонстрировал: глубокие исчерпывающие знания и понимание учебного материала; содержательные, полные, правильные и конкретные ответы на все вопросы, включая дополнительные; свободное владение основной и дополнительной литературой, рекомендованной учебной программой дисциплины. |
| | Уметь | Обучающийся продемонстрировал: понимание учебного материала; умение свободно решать практические задания (ситуационные задачи), которые следует выполнить или описание результата, который нужно получить и др.; логически последовательные, содержательные, полные, правильные и конкретные ответы (решения) на все поставленные задания (вопросы), включая дополнительные; свободное владение основной и дополнительной литературой, рекомендованной учебной программой дисциплины. |
| | Владеть | Обучающийся продемонстрировал: понимание учебного материала; умение свободно решать комплексные практические задания (решения задач по нестандартным ситуациям); логически последовательные, полные, правильные и конкретные ответы в ходе защиты задания, включая дополнительные уточняющие вопросы (задания); свободное владение основной и дополнительной литературой, рекомендованной учебной программой дисциплины. |
| Средний уровень (хорошо) | Знать | Обучающийся продемонстрировал: твердые и достаточно полные знания учебного материала; правильное понимание сущности и взаимосвязи рассматриваемых процессов и явлений; последовательные, правильные, конкретные ответы на поставленные вопросы при свободном устранении замечаний по отдельным вопросам; достаточное владение литературой, рекомендованной учебной программой дисциплины |
| | Уметь | Обучающийся продемонстрировал: понимание учебного материала; логически последовательные, правильные и конкретные ответы (решения) на основные задания (вопросы), включая дополнительные; устранение замечаний по отдельным элементам задания (вопроса); владение основной и дополнительной литературой, рекомендованной учебной программой дисциплины |
| | Владеть | Обучающийся продемонстрировал: понимание учебного материала; продемонстрировал логически последовательные, достаточно полные, правильные ответы, включая дополнительные; самостоятельно устранил замечания по отдельным элементам задания (вопроса); владение основной и дополнительной литературой, рекомендованной учебной программой дисциплины |
| Достаточный уровень (удовлетворительно) | Знать | Обучающийся продемонстрировал: твердые знания и понимание основного учебного материала; правильные, без грубых ошибок, ответы на поставленные вопросы при устранении неточностей и несущественных ошибок в освещении отдельных положений при наводящих вопросах преподавателя; недостаточно полное владение литературой, рекомендованной учебной программой дисциплины |
| | Уметь | Обучающийся продемонстрировал: понимание основного учебного материала; правильные, без грубых ошибок, ответы (решения) на основные задания (вопросы), включая дополнительные; устранение, при наводящих вопросах преподавателя, замечаний по отдельным элементам задания |

| | | |
|--|----------------|--|
| | | (вопроса); недостаточное полное владение литературой, рекомендованной учебной программой дисциплины |
| | Владеть | Обучающийся понимание основного учебного материала; без грубых ошибок дал ответы на поставленные вопросы при устранении неточностей и ошибок в решениях в ходе защиты задания (проекта, портфолио) при наводящих вопросах преподавателя; недостаточно полное владение литературой, рекомендованной учебной программой дисциплины |

ОПИСАНИЕ ШКАЛ ОЦЕНИВАНИЯ

При проведении промежуточной аттестации в ЧУВО «ВШП» используются традиционные формы аттестации:

| Форма промежуточной аттестации | Шкала оценивания |
|--------------------------------|---|
| ЗАЧЕТ | «зачтено», «незачтено» |
| ЭКЗАМЕН | «отлично», «хорошо», «удовлетворительно», «неудовлетворительно» |

КРИТЕРИИ И ПРОЦЕДУРЫ ОЦЕНИВАНИЯ КОМПЕТЕНЦИЙ НА РАЗЛИЧНЫХ ЭТАПАХ ИХ ФОРМИРОВАНИЯ

Для оценивания результатов обучения в виде **ЗНАНИЙ** используются следующие процедуры и технологии:

- тестирование.

Для оценивания результатов обучения в виде **УМЕНИЙ и ВЛАДЕНИЙ** используются следующие процедуры и технологии:

- устный или письменный ответ на вопрос.
- практические задания, включающие одну или несколько задач (вопросов) в виде краткой формулировки действий (комплекса действий), которые следует выполнить, или описать результат, который нужно получить.

Критерии оценивания результата обучения по дисциплине (модулю)

| Результат обучения по дисциплине (модулю) | ШКАЛА ОЦЕНИВАНИЯ | | | | Процедуры оценивания |
|---|---|---|---|---|----------------------|
| | «отлично» | «хорошо» | «удовлетворительно» | «неудовлетворительно» | |
| <u>УК-1</u> <u>ПК-1</u> <u>ПК-2</u> <u>ПК-4</u> Знать: | Обучаемый продемонстрировал: глубокие исчерпывающие знания и понимание учебного материала; содержательные, полные, правильные и конкретные ответы на все вопросы, включая дополнительные; свободное владение основной и дополнительной литературой, рекомендованной учебной | Обучаемый продемонстрировал: твердые и достаточно полные знания учебного материала; правильное понимание сущности и взаимосвязи рассматриваемых процессов и явлений; последовательные, правильные, конкретные ответы на поставленные вопросы при свободном устранении | Обучаемый продемонстрировал: твердые знания и понимание основного учебного материала; правильные, без грубых ошибок, ответы на поставленные вопросы при устранении неточностей и несущественных ошибок в освещении отдельных положений при наводящих вопросах преподавателя; недостаточно полное владение | Обучаемый продемонстрировал: неправильные ответы на основные вопросы; грубые ошибки в ответах; непонимание сущности излагаемых вопросов; неуверенные и неточные ответы на дополнительные вопросы; не владеет основной литературой, рекомендованной учебной программой дисциплины. | Тестовые задания |

| | | | | | |
|---|--|---|--|--|------------------------------|
| | программой дисциплины. | замечаний по отдельным вопросам; достаточное владение литературой. | литературой, рекомендованной учебной программой дисциплины. | | |
| <u>УК-1</u> <u>ПК-1</u> <u>ПК-2</u> <u>ПК-4</u> <u>Уметь:</u> | Обучаемый продемонстрировал: понимание учебного материала, содержательные, полные, правильные и конкретные ответы на все поставленные вопросы, включая дополнительные; свободное владение основной и дополнительной литературой, рекомендованной учебной программой дисциплины | Обучаемый продемонстрировал: понимание учебного материала; логически последовательные, правильные и конкретные ответы на основные задания/вопросы, включая дополнительные; устранение замечаний по отдельным элементам задания; владение основной и дополнительной литературой, рекомендованной учебной программой дисциплины | Обучаемый продемонстрировал: понимание основного учебного материала; без грубых ошибок, ответы на основные вопросы, включая дополнительные, устранение, при наводящих вопросах преподавателя, замечаний по отдельным элементам задания; недостаточное полное владение литературой, рекомендованной учебной программой дисциплины | Обучаемый продемонстрировал: непонимание основного учебного материала; не дал правильные ответы на основные вопросы, включая дополнительные; не устранил, при наводящих вопросах преподавателя, замечания и грубые ошибки по вопросу; не владеет основной учебной литературой, рекомендованной учебной программой дисциплины | Вопросы Практические задания |
| <u>УК-1</u> <u>ПК-1</u> <u>ПК-2</u> <u>ПК-4</u> <u>Владеть:</u> | Обучаемый продемонстрировал: понимание учебного материала; правильные и конкретные ответы, включая дополнительные и уточняющие вопросы; свободное владение основной и дополнительной литературой, рекомендованной учебной программой дисциплины | Обучаемый продемонстрировал: понимание учебного материала; продемонстрировал логически последовательные, достаточно полные, верные ответы; самостоятельно устранил замечания по отдельным элементам; владение основной и дополнительной литературой, рекомендованной учебной программой дисциплины | Обучаемый продемонстрировал: понимание основного учебного материала; без грубых ошибок дал ответы на поставленные вопросы, в том числе при наводящих вопросах преподавателя; недостаточно полное владение литературой, рекомендованной учебной программой дисциплины | Обучаемый продемонстрировал: непонимание основного учебного материала; дал неправильные ответы на поставленные вопросы; не владеет основной учебной литературой, рекомендованной учебной программой дисциплины | Вопросы Практические задания |

1. Оценочные материалы для самостоятельной работы обучающихся (студентов)

1.1 Реферат

5 семестр

Реферат позволит студентам углубиться в различные аспекты разработки мобильных приложений, а также понять их важность и применение в современных информационных системах. Реферат покрывает компетенции УК-1.1, УК-1.2, ПК-1.1, ПК-1.2, ПК-4.1, ПК-4.2.

Примерная тематика рефератов:

- 1. История и эволюция мобильных платформ**
 - Исследование развития мобильных операционных систем, таких как Android и iOS, их влияние на рынок и пользователей.
- 2. Основы языка программирования Java в контексте мобильных приложений**
 - Анализ ключевых аспектов языка Java и его применения в разработке мобильных приложений для Android.
- 3. Проектирование пользовательского интерфейса в Android**
 - Изучение принципов и лучших практик проектирования пользовательского интерфейса для Android-приложений.
- 4. Фрагменты и адаптивные интерфейсы в Android**
 - Исследование использования фрагментов для создания адаптивных интерфейсов в мобильных приложениях.
- 5. Технологии хранения данных в мобильных приложениях**
 - Обзор различных методов хранения данных в Android, таких как SharedPreferences и SQLite, и их применение.
- 6. Использование языка Kotlin для разработки Android-приложений**
 - Анализ преимуществ использования Kotlin для разработки мобильных приложений и сравнение с Java.

1.2 Реферат

6 семестр

Реферат позволит студентам углубиться в различные аспекты разработки мобильных приложений, а также понять их важность и применение в современных информационных системах. Реферат покрывает компетенции УК-1.1, УК-1.2, ПК-1.1, ПК-1.2, ПК-4.1, ПК-4.2.

Примерная тематика рефератов:

- 1. Сетевые технологии и веб-сервисы в мобильных приложениях**
 - Изучение интеграции сетевых запросов и веб-сервисов в Android-приложениях с использованием Retrofit.
- 2. Асинхронное программирование в Android**
 - Обзор методов и инструментов асинхронного программирования, таких как AsyncTask и RxJava, и их применение в мобильных приложениях.
- 3. Работа с мультимедиа в Android-приложениях**
 - Исследование возможностей и методов работы с аудио и видео в мобильных приложениях.
- 4. Интеграция мобильных приложений с внешними сервисами**
 - Изучение методов интеграции приложений с Google Play Services, картами, геолокацией и Firebase.
- 5. Тестирование и отладка мобильных приложений**
 - Обзор методов тестирования и отладки мобильных приложений, таких как юнит-тестирование и инструментальные тесты.
- 6. Процесс публикации и распространения мобильных приложений**

- Исследование этапов подготовки и публикации мобильных приложений в Google Play Store.

Цель написания рефератов: Углубить понимание и критическое осмысление различных аспектов разработки мобильных приложений, развивая аналитические и научные навыки студентов.

Структура реферата:

Введение

- Краткое описание темы и целей реферата.
- Актуальность темы.

Основная часть

- Теоретические основы темы.
- История и эволюция (если применимо).
- Применение в современной разработке мобильных приложений.
- Примеры и кейсы.
- Проблемы и вызовы.
- Перспективы и тенденции развития.

Заключение

- Выводы по результатам исследования.
- Значение изученных аспектов для разработки мобильных приложений.

Список использованных источников

- Перечень использованной литературы и интернет-ресурсов.

Критерии оценивания:

1. **Структура и логика изложения (20%)**
 - Четкая структура работы (введение, основная часть, заключение).
 - Логичность и последовательность изложения материала.
2. **Содержание (40%)**
 - Полнота раскрытия темы.
 - Описание основных этапов развития разработки мобильных приложений.
 - Анализ современных тенденций.
 - Примеры применения мобильной разработки в информационных системах.
3. **Аналитическая часть (20%)**
 - Глубина анализа роли разработки мобильных приложений в информационных системах.
 - Наличие собственных выводов и оценок.
4. **Оформление (10%)**
 - Соответствие требованиям к оформлению рефератов (шрифт, отступы, заголовки и т.д.).
 - Корректное оформление ссылок и списка литературы.
5. **Язык и стиль (10%)**
 - Грамотность и точность изложения.
 - Научный стиль текста.

Требования к объему:

Объем реферата должен составлять 10-15 страниц печатного текста (шрифт Times New Roman, размер 12, интервал 1.5, поля 2 см со всех сторон).

2. Оценочные материалы для оценки текущей аттестации обучающихся (студентов)

2.1 Тестовые задания для текущего контроля успеваемости в виде ЗНАНИЙ

В тестовом задании вопросы, которые имеют закрытый характер.

Правильные ответы выделены знаком +.

5 семестр

1. Какое из следующих утверждений правильно описывает Android? (УК-1.1)

- Операционная система для серверов.
- Операционная система для мобильных устройств. +
- Язык программирования.
- Графический интерфейс пользователя.

2. Как называется инструмент для разработки Android-приложений? (УК-1.1)

- Eclipse.
- Visual Studio.
- Xcode.
- Android Studio. +

3. Какие из следующих компонентов не являются основными в Android-приложении? (ПК-4.2)

- Activity.
- Service.
- Broadcast Receiver.
- API Gateway. +

4. Какое ключевое слово используется для определения класса в Java? (ПК-1.1)

- define.
- class. +
- struct.
- module.

5. Какое ключевое слово используется для наследования в Java? (ПК-1.1)

- inherits.
- extends. +
- implements.
- inheritsFrom.

6. Какой метод используется для запуска Activity в Android? (ПК-1.1)

- startActivity. +
- beginActivity.
- launchActivity.
- openActivity.

7. Какой из следующих Layout используется для размещения элементов в виде сетки? (ПК-4.2)

- LinearLayout.
- RelativeLayout.
- GridLayout. +
- ConstraintLayout.

8. Какое из следующих утверждений правильно описывает View в Android? (ПК-4.2)

- Элемент интерфейса пользователя. +
- Функция для обработки данных.
- Класс для работы с базой данных.
- Метод для взаимодействия с сетью.

9. Какое ключевое слово используется для объявления функции в Kotlin? (ПК-1.1)

- func.
- def.
- function.

- fun. +
- 10. Какой из следующих типов данных является неизменяемым в Kotlin? (ПК-1.1)**
- var.
 - val. +
 - let.
 - const.
- 11. Какое из следующих утверждений верно для SQLite в Android? (ПК-4.2)**
- Это облачная база данных.
 - Это реляционная база данных. +
 - Это графовая база данных.
 - Это NoSQL база данных.
- 12. Какой метод используется для получения данных из SharedPreferences? (ПК-1.1)**
- getString. +
 - fetchString.
 - retrieveString.
 - acquireString.
- 13. Какой метод используется для создания фрагмента в Android? (ПК-4.2)**
- createFragment.
 - newFragment.
 - instantiateFragment.
 - newInstance. +
- 14. Какое из следующих утверждений верно для жизненного цикла фрагмента? (ПК-4.2)**
- Фрагмент существует только во время старта Activity.
 - Фрагмент может быть добавлен и удален во время выполнения Activity. +
 - Фрагмент не может взаимодействовать с Activity.
 - Фрагмент всегда должен быть статическим.
- 15. Какое из следующих утверждений верно для использования камеры в Android? (ПК-1.2)**
- Камеру можно использовать только с библиотекой OpenCV.
 - Камеру можно использовать через Intent для запуска встроенного приложения камеры. +
 - Камеру можно использовать только через API CameraX.
 - Камера недоступна для использования в приложениях.
- 16. Какой элемент используется для создания анимаций в Android? (ПК-4.2)**
- Animator. +
 - Mover.
 - Transformer.
 - AnimatorSet.
- 17. Какое из следующих утверждений верно для создания уведомлений в Android? (ПК-1.2)**
- Уведомления можно создавать только через Firebase.
 - Уведомления можно создавать с помощью NotificationManager. +
 - Уведомления создаются с помощью ActivityManager.
 - Уведомления недоступны для использования в Android-приложениях.
- 18. Какой компонент используется для выполнения фоновых задач в Android? (ПК-1.2)**
- BroadcastReceiver.
 - Service. +
 - ContentProvider.
 - Activity.

Критерии оценки результатов теста

1. **"Неудовлетворительно" (0-39%)**
 - Студент ответил правильно на менее 40% вопросов.
 - Значительные пробелы в знаниях по большинству тем.
 - Неправильное понимание ключевых понятий и принципов.

- Неспособность применить теоретические знания на практике.
- 2. **"Удовлетворительно" (40-59%)**
 - Студент ответил правильно на 40-59% вопросов.
 - Основные понятия и принципы поняты частично, есть ошибки в ответах.
 - Знания по большинству тем на базовом уровне, недостаточная глубина понимания.
 - Частичная способность применять теоретические знания на практике, нужны дополнительные разъяснения.
- 3. **"Хорошо" (60-79%)**
 - Студент ответил правильно на 60-79% вопросов.
 - Хорошее понимание ключевых понятий и принципов, незначительные ошибки.
 - Знания по всем темам на достаточном уровне, однако есть некоторые пробелы.
 - Способность применять теоретические знания на практике, но требуется улучшение точности и уверенности.
- 4. **"Отлично" (80-100%)**
 - Студент ответил правильно на 80-100% вопросов.
 - Полное и правильное понимание всех ключевых понятий и принципов.
 - Глубокие знания по всем темам, минимальные или отсутствующие ошибки.
 - Высокий уровень способности применять теоретические знания на практике, демонстрация уверенности и точности в ответах.

2.2 Тестовые задания для текущего контроля успеваемости в виде ЗНАНИЙ

В тестовом задании вопросы, которые имеют закрытый характер.

Правильные ответы выделены знаком +.

6 семестр

1. **Какой библиотекой используется для выполнения HTTP-запросов в Android? (ПК-2.1)**
 - Gson.
 - Retrofit. +
 - OkHttp.
 - Volley.
2. **Какой класс используется для выполнения асинхронных задач в Android? (ПК-2.1)**
 - Thread.
 - AsyncTask. +
 - Service.
 - Handler.
3. **Какое из следующих утверждений верно для использования Google Play Services? (ПК-1.2)**
 - Google Play Services предоставляет только аналитические функции.
 - Google Play Services предоставляет широкий спектр API, включая карты и геолокацию. +
 - Google Play Services используется только для аутентификации.
 - Google Play Services предназначен только для управления уведомлениями.
4. **Какой инструмент используется для тестирования мобильных приложений на Android? (ПК-2.1)**
 - Espresso. +
 - Mocha.
 - Selenium.
 - JUnit.
5. **Какое ключевое слово используется для обработки исключений в Java? (ПК-2.2)**
 - handle.
 - catch. +

- excerpt.
- capture.

6. Какой метод используется для публикации приложения в Google Play Store? (ПК-4.1)

- Выбор меню "Publish" в Android Studio.
- Создание подписанного APK и загрузка его через Google Play Console. +
- Использование команды adb publish.
- Отправка APK по электронной почте в Google.

7. Какое из следующих утверждений верно для процесса отладки приложения в Android Studio? (ПК-1.2)

- Отладка возможна только на реальном устройстве.
- Отладка возможна как на эмуляторе, так и на реальном устройстве. +
- Отладка возможна только на эмуляторе.
- Отладка недоступна в Android Studio.

8. Какой элемент используется для создания пользовательских View в Android? (ПК-4.2)

- ViewGroup.
- CustomView. +
- LayoutInflater.
- ViewStub.

9. Какое из следующих утверждений верно для интеграции Firebase в Android-приложения? (ПК-4.1)

- Firebase используется только для аутентификации.
- Firebase предоставляет широкий спектр сервисов, включая аналитику и базы данных в реальном времени. +
- Firebase доступен только для платных пользователей.
- Firebase используется только для хранения файлов.

10. Какое ключевое слово используется для создания корутин в Kotlin? (ПК-2.1)

- coroutine.
- launch. +
- async.
- start.

11. Какое из следующих утверждений верно для использования WorkManager в Android? (ПК-1.2)

- WorkManager используется для управления UI.
- WorkManager используется для выполнения долгосрочных фоновых задач. +
- WorkManager используется для обработки событий.
- WorkManager используется для управления ресурсами приложения.

12. Какое из следующих утверждений верно для создания адаптивных интерфейсов в Android? (ПК-4.2)

- Адаптивные интерфейсы могут быть созданы только с помощью ConstraintLayout.
- Адаптивные интерфейсы могут быть созданы с использованием различных Layout'ов и фрагментов. +
- Адаптивные интерфейсы не поддерживаются в Android.
- Адаптивные интерфейсы требуют использования сторонних библиотек.

13. Какое из следующих утверждений верно для публикации приложений в Google Play Store? (ПК-4.1)

- Публикация приложений возможна только для компаний.
- Публикация приложений возможна для всех разработчиков с учетной записью Google Play Console. +
- Публикация приложений возможна только для бесплатных приложений.
- Публикация приложений требует использования сторонних сервисов.

14. Какое из следующих утверждений верно для использования Espresso для тестирования? (ПК-2.1)

- Espresso используется для написания юнит-тестов.
- Espresso используется для написания инструментальных тестов. +
- Espresso используется для тестирования производительности.
- Espresso используется для написания UI-тестов только на C++.

15. Какой метод используется для воспроизведения видео в Android? (ПК-1.2)

- playVideo.
- startPlayback.
- setVideoURI. +
- initializePlayer.

16. Какое из следующих утверждений верно для использования карты в Android? (ПК-1.2)

- Карты могут быть интегрированы только через Google Play Services. +
- Карты не могут быть интегрированы в приложения.
- Карты требуют использования сторонних библиотек.
- Карты могут быть интегрированы только через Firebase.

17. Какое из следующих утверждений верно для отладки приложения с помощью Android Studio? (ПК-1.2)

- Логирование может быть выполнено с помощью Logcat. +
- Логирование невозможно с помощью Android Studio.
- Логирование может быть выполнено только на реальном устройстве.
- Логирование доступно только в платной версии Android Studio.

Критерии оценки результатов теста

5. "Неудовлетворительно" (0-39%)

- Студент ответил правильно на менее 40% вопросов.
- Значительные пробелы в знаниях по большинству тем.
- Неправильное понимание ключевых понятий и принципов.
- Неспособность применить теоретические знания на практике.

6. "Удовлетворительно" (40-59%)

- Студент ответил правильно на 40-59% вопросов.
- Основные понятия и принципы поняты частично, есть ошибки в ответах.
- Знания по большинству тем на базовом уровне, недостаточная глубина понимания.
- Частичная способность применять теоретические знания на практике, нужны дополнительные разъяснения.

7. "Хорошо" (60-79%)

- Студент ответил правильно на 60-79% вопросов.
- Хорошее понимание ключевых понятий и принципов, незначительные ошибки.
- Знания по всем темам на достаточном уровне, однако есть некоторые пробелы.
- Способность применять теоретические знания на практике, но требуется улучшение точности и уверенности.

8. "Отлично" (80-100%)

- Студент ответил правильно на 80-100% вопросов.

- Полное и правильное понимание всех ключевых понятий и принципов.
- Глубокие знания по всем темам, минимальные или отсутствующие ошибки.
- Высокий уровень способности применять теоретические знания на практике, демонстрация уверенности и точности в ответах.

2.3 Вопросы для текущего контроля успеваемости в виде УМЕНИЙ

5 семестр

1. Что такое операционная система Android? (УК-1.1)

- Правильный ответ: Android - это операционная система для мобильных устройств, разработанная компанией Google. Она предназначена для управления аппаратными ресурсами и выполнения приложений на смартфонах, планшетах и других мобильных устройствах.

2. Какой инструмент используется для разработки Android-приложений? (УК-1.1)

- Правильный ответ: Android Studio - это официальная интегрированная среда разработки (IDE) от Google для создания Android-приложений. Она предоставляет инструменты для разработки, тестирования и отладки приложений.

3. Какие основные компоненты входят в Android-приложение? (ПК-4.2)

- Правильный ответ: Основными компонентами Android-приложения являются Activity, Service, Broadcast Receiver и Content Provider. Эти компоненты выполняют разные роли в жизненном цикле приложения и обеспечивают взаимодействие с пользователем и системными ресурсами.

4. Какое ключевое слово используется для определения класса в Java? (ПК-1.1)

- Правильный ответ: Ключевое слово `class` используется для определения класса в языке программирования Java. Классы являются основными строительными блоками объектно-ориентированного программирования.

5. Какое ключевое слово используется для наследования в Java? (ПК-1.1)

- Правильный ответ: Ключевое слово `extends` используется для наследования в Java. Оно позволяет создать новый класс на основе существующего, наследуя его методы и свойства.

6. Какой метод используется для запуска Activity в Android? (ПК-1.1)

- Правильный ответ: Метод `startActivity` используется для запуска новой Activity в Android. Он принимает `Intent`, который описывает операцию, которую нужно выполнить.

7. Какие виды Layout'ов используются для размещения элементов в Android? (ПК-4.2)

- Правильный ответ: В Android используются различные виды Layout'ов для размещения элементов интерфейса: `LinearLayout`, `RelativeLayout`, `GridLayout` и `ConstraintLayout`. Каждый из них имеет свои особенности и применимость в разных ситуациях.

8. Что такое View в Android? (ПК-4.2)

- Правильный ответ: View в Android - это базовый элемент интерфейса пользователя, который представляет собой прямоугольную область на экране. Все элементы интерфейса, такие как кнопки, текстовые поля и изображения, являются наследниками класса View.

9. Какое ключевое слово используется для объявления функции в Kotlin? (ПК-1.1)

- Правильный ответ: Ключевое слово `fun` используется для объявления функции в Kotlin. Это ключевое слово указывает, что за ним следует определение функции.

10. Какой тип данных является неизменяемым в Kotlin? (ПК-1.1)

- Правильный ответ: В Kotlin тип данных, объявленный с помощью ключевого слова `val`, является неизменяемым. Это означает, что значение переменной нельзя изменить после ее инициализации.

11. Что такое SQLite в контексте Android? (ПК-4.2)

- Правильный ответ: SQLite - это встроенная реляционная база данных в Android, которая позволяет приложениям хранить и управлять структурированными данными. Она предоставляет полный набор возможностей для работы с базами данных, включая создание, чтение, обновление и удаление данных.
- 12. Какой метод используется для получения данных из SharedPreferences? (ПК-1.1)**
- Правильный ответ: Метод getString используется для получения строковых данных из SharedPreferences. SharedPreferences - это механизм для сохранения простых данных в виде пар "ключ-значение".
- 13. Какой метод используется для создания фрагмента в Android? (ПК-4.2)**
- Правильный ответ: Метод newInstance часто используется для создания новых экземпляров фрагментов в Android. Этот метод позволяет передавать параметры в фрагмент при его создании.
- 14. Что представляет собой жизненный цикл фрагмента? (ПК-4.2)**
- Правильный ответ: Жизненный цикл фрагмента включает в себя последовательность состояний от создания до уничтожения, таких как onCreate, onCreateView, onActivityCreated, onStart, onResume, onPause, onStop и onDestroy. Эти методы позволяют управлять поведением фрагмента на различных этапах его существования.
- 15. Как можно использовать камеру в Android-приложениях? (ПК-1.2)**
- Правильный ответ: Камеру можно использовать в Android-приложениях через Intent для запуска встроенного приложения камеры. Это позволяет делать снимки или записывать видео без необходимости напрямую работать с аппаратным обеспечением камеры.
- 16. Какой элемент используется для создания анимаций в Android? (ПК-4.2)**
- Правильный ответ: Класс Animator используется для создания анимаций в Android. С его помощью можно анимировать свойства объектов, такие как положение, размер и прозрачность.
- 17. Как можно создать уведомления в Android? (ПК-1.2)**
- Правильный ответ: Уведомления в Android создаются с помощью NotificationManager. Этот класс позволяет приложениям отправлять уведомления, которые отображаются пользователю в статусной строке.
- 18. Какой компонент используется для выполнения фоновых задач в Android? (ПК-1.2)**
- Правильный ответ: Компонент Service используется для выполнения фоновых задач в Android. Service позволяет выполнять длительные операции в фоновом режиме, не взаимодействуя напрямую с пользователем.

2.4 Вопросы для текущего контроля успеваемости в виде УМЕНИЙ

6 семестр

- 1. Какой библиотекой используется для выполнения HTTP-запросов в Android? (ПК-2.1)**
 - Правильный ответ: Для выполнения HTTP-запросов в Android часто используется библиотека Retrofit. Она упрощает процесс взаимодействия с веб-сервисами, предоставляя удобный интерфейс для выполнения запросов и обработки ответов.
- 2. Какой класс используется для выполнения асинхронных задач в Android? (ПК-2.1)**
 - Правильный ответ: Класс AsyncTask используется для выполнения асинхронных задач в Android. Он позволяет выполнять длительные операции в фоновом режиме и обновлять пользовательский интерфейс по завершении задачи.
- 3. Что такое Google Play Services? (ПК-1.2)**
 - Правильный ответ: Google Play Services - это набор API от Google, который позволяет разработчикам интегрировать свои приложения с различными сервисами, такими как карты, геолокация, аутентификация и аналитика.
- 4. Какой инструмент используется для тестирования мобильных приложений на Android? (ПК-2.1)**

- Правильный ответ: Инструмент Espresso используется для тестирования мобильных приложений на Android. Он позволяет автоматизировать тестирование пользовательского интерфейса, обеспечивая высокое качество и стабильность приложений.
5. **Какое ключевое слово используется для обработки исключений в Java? (ПК-2.2)**
- Правильный ответ: Ключевое слово catch используется для обработки исключений в Java. Оно позволяет перехватывать и обрабатывать ошибки, которые возникают во время выполнения программы.
6. **Какой метод используется для публикации приложения в Google Play Store? (ПК-4.1)**
- Правильный ответ: Для публикации приложения в Google Play Store необходимо создать подписанный APK и загрузить его через Google Play Console. Этот процесс включает подготовку приложения, его тестирование и подачу на рассмотрение.
7. **Как выполняется отладка приложения в Android Studio? (ПК-1.2)**
- Правильный ответ: Отладка приложения в Android Studio возможна как на эмуляторе, так и на реальном устройстве. Для этого используются инструменты, такие как Logcat и отладчик, которые помогают выявлять и исправлять ошибки в коде.
8. **Какой элемент используется для создания пользовательских View в Android? (ПК-4.2)**
- Правильный ответ: Для создания пользовательских View в Android используется класс CustomView. Этот класс позволяет разработчикам создавать уникальные элементы интерфейса, которые не включены в стандартный набор виджетов.
9. **Что предоставляет Firebase для Android-приложений? (ПК-4.1)**
- Правильный ответ: Firebase предоставляет широкий спектр сервисов для Android-приложений, включая аналитику, аутентификацию, базы данных в реальном времени и облачное хранение файлов. Эти сервисы помогают разработчикам создавать более функциональные и масштабируемые приложения.
10. **Какое ключевое слово используется для создания корутин в Kotlin? (ПК-2.1)**
- Правильный ответ: Ключевое слово launch используется для создания корутин в Kotlin. Корутину можно рассматривать как легковесную потоковую задачу, которая выполняется асинхронно и позволяет писать асинхронный код более компактно и понятно.
11. **Для чего используется WorkManager в Android? (ПК-1.2)**
- Правильный ответ: WorkManager используется для выполнения долгосрочных фоновых задач в Android. Он обеспечивает надежное выполнение задач, даже если приложение закрыто или устройство перезагружено.
12. **Что такое адаптивные интерфейсы в Android? (ПК-4.2)**
- Правильный ответ: Адаптивные интерфейсы в Android - это интерфейсы, которые могут изменяться в зависимости от устройства и его экрана. Они создаются с использованием различных Layout'ов и фрагментов, чтобы обеспечить оптимальное отображение на разных экранах.
13. **Кто может публиковать приложения в Google Play Store? (ПК-4.1)**
- Правильный ответ: Публиковать приложения в Google Play Store могут все разработчики, имеющие учетную запись Google Play Console. Этот процесс требует оплаты регистрационного взноса и соблюдения правил публикации.
14. **Для чего используется Espresso в тестировании? (ПК-2.1)**
- Правильный ответ: Espresso используется для написания инструментальных тестов в Android-приложениях. Эти тесты помогают автоматизировать проверку пользовательского интерфейса, обеспечивая высокое качество и стабильность приложений.
15. **Какой метод используется для воспроизведения видео в Android? (ПК-1.2)**

- Правильный ответ: Метод `setVideoURI` используется для воспроизведения видео в Android. Он позволяет установить URI видеофайла, который будет воспроизводиться в компоненте `VideoView`.

16. Как можно интегрировать карты в Android-приложения? (ПК-1.2)

- Правильный ответ: Карты могут быть интегрированы в Android-приложения через Google Play Services. Это позволяет использовать API карт для отображения карт и работы с геолокацией в приложениях.

17. Как выполняется логирование в Android Studio? (ПК-1.2)

- Правильный ответ: Логирование в Android Studio выполняется с помощью инструмента Logcat. Этот инструмент позволяет выводить отладочные сообщения, ошибки и другую информацию для анализа работы приложения.

Критерии оценки ответов на вопросы

- **"Отлично" (5 баллов)**
 - **Критерии:**
 - Полное и точное объяснение вопроса.
 - Ответ включает все ключевые аспекты и детали.
 - Примеры, если требуются, приведены и правильно объяснены.
 - Ответ демонстрирует глубокое понимание темы.
- **"Хорошо" (4 балла)**
 - **Критерии:**
 - Корректное объяснение вопроса.
 - Ответ охватывает основные аспекты, но может отсутствовать незначительная деталь или пример.
 - Демонстрируется хорошее, но не полное понимание темы.
- **"Удовлетворительно" (3 балла)**
 - **Критерии:**
 - Общее представление о вопросе.
 - Ответ включает основные аспекты, но содержит неточности или пропуски.
 - Примеры, если требуются, могут отсутствовать или быть неверно объяснены.
 - Демонстрируется базовое понимание темы.
- **"Неудовлетворительно" (2 балла)**
 - **Критерии:**
 - Некорректное или неполное объяснение вопроса.
 - Отсутствие ключевых аспектов и деталей.
 - Примеры, если требуются, отсутствуют или приведены неверные.
 - Ответ демонстрирует недостаточное понимание темы.

2.5 Задачи на соответствие понятий для текущего контроля успеваемости в виде ВЛАДЕНИЙ

Правильные ответы расположены в таблицах друг напротив друга, во время тестирования предполагается что порядок данных в рамках каждого столбца будет случайным.

Задача 1. Соотнесите основные понятия Java с их определениями

(УК-1.1, ПК-1.1)

| Понятие | Определение |
|-----------|---|
| А - Класс | 1 - Шаблон для создания объектов, объединяющий данные и методы. |

| | |
|----------------|--|
| В - Объект | 2 - Экземпляр класса, представляющий конкретную сущность. |
| С - Метод | 3 - Функция или процедура, определенная внутри класса. |
| D - Переменная | 4 - Место в памяти для хранения значения определенного типа. |

Правильный ответ: A-1, B-2, C-3, D-4

Задача 2. Соотнесите ключевые элементы Android Studio с их функциями
(ПК-1.1, ПК-1.2)

| Понятие | Определение |
|-------------------|--|
| A - Gradle | 1 - Система сборки, используемая для автоматизации компиляции и сборки проектов. |
| B - AVD Manager | 2 - Инструмент для управления виртуальными устройствами. |
| C - Logcat | 3 - Инструмент для вывода отладочных сообщений и логирования. |
| D - Layout Editor | 4 - Инструмент для визуального создания и редактирования интерфейсов. |

Правильный ответ: A-1, B-2, C-3, D-4

Задача 3. Соотнесите компоненты Android с их функциями
(ПК-1.1, ПК-4.2)

| Понятие | Определение |
|-----------------------|---|
| A - SharedPreferences | 1 - Механизм для хранения простых данных в виде пар "ключ-значение". |
| B - SQLite Database | 2 - Встроенная реляционная база данных для хранения структурированных данных. |
| C - Content Provider | 3 - Компонент, обеспечивающий доступ к данным других приложений. |
| D - Fragment | 4 - Модуль интерфейса, который может быть переиспользован в разных частях приложения. |

Правильный ответ: A-1, B-2, C-3, D-4

Задача 4. Соотнесите элементы пользовательского интерфейса Android с их описаниями
(ПК-4.2, ПК-1.2)

| Понятие | Определение |
|------------------|--|
| A - TextView | 1 - Элемент, предназначенный для отображения текста. |
| B - Button | 2 - Кнопка, которую можно нажать для выполнения действия. |
| C - ImageView | 3 - Элемент, предназначенный для отображения изображений. |
| D - RecyclerView | 4 - Контейнер для отображения большого количества данных с прокруткой. |

Правильный ответ: A-1, B-2, C-3, D-4

Задача 5. Соотнесите методы классов с их назначением в Android

(ПК-1.1, ПК-2.1)

| Понятие | Определение |
|-----------------|---|
| A - onCreate() | 1 - Метод, вызываемый при создании Activity или фрагмента. |
| B - onStart() | 2 - Метод, вызываемый перед тем, как Activity становится видимой пользователю. |
| C - onResume() | 3 - Метод, вызываемый, когда Activity начинает взаимодействовать с пользователем. |
| D - onDestroy() | 4 - Метод, вызываемый перед уничтожением Activity или фрагмента. |

Правильный ответ: A-1, B-2, C-3, D-4

Задача 6. Соотнесите основные понятия Android с их определениями

(ПК-1.1, ПК-4.2)

| Понятие | Определение |
|-----------------------|--|
| A - Intent | 1 - Объект, используемый для передачи данных между компонентами. |
| B - Activity | 2 - Компонент, представляющий один экран с пользовательским интерфейсом. |
| C - BroadcastReceiver | 3 - Компонент, реагирующий на широковещательные сообщения. |
| D - Service | 4 - Компонент, выполняющий длительные операции в фоновом режиме. |

Правильный ответ: A-1, B-2, C-3, D-4

Критерии оценки выполнения задач на соответствие понятий

- **Правильность соответствий:**
 - **Отлично (5):** Все соответствия выполнены правильно.
 - **Хорошо (4):** 1 ошибка в соответствиях.
 - **Удовлетворительно (3):** 2 ошибки в соответствиях.
 - **Неудовлетворительно (2):** 3 и более ошибок в соответствиях.

3. Оценочные материалы для проведения промежуточной аттестации обучающихся (студентов)

3.1 Вопросы для проведения промежуточной аттестации в форме ЗАЧЕТА 5 семестр

1. Объясните архитектуру компонентов Android-приложения и их взаимодействие. (УК-1.1)

- Пример правильного ответа: Архитектура Android-приложения включает основные компоненты: Activity, Service, Broadcast Receiver и Content Provider. Activity представляет собой экран с пользовательским интерфейсом, Service выполняет длительные операции в фоновом режиме, Broadcast Receiver реагирует на системные широковещательные сообщения, а Content Provider управляет доступом к данным приложения. Эти компоненты взаимодействуют через Intent, который позволяет запускать компоненты и передавать данные между ними.

2. Каковы основные отличия между LinearLayout, RelativeLayout и ConstraintLayout в Android? Приведите примеры использования каждого из них. (ПК-4.2)

- Пример правильного ответа: LinearLayout выстраивает элементы в строку или столбец, RelativeLayout позволяет располагать элементы относительно друг друга, а ConstraintLayout предлагает гибкий способ размещения элементов с использованием ограничений. Например, LinearLayout подходит для простых списков, RelativeLayout — для сложных форм с элементами, зависящими друг от друга, а ConstraintLayout — для адаптивных интерфейсов с многоуровневыми зависимостями.

3. Опишите процесс создания и использования фрагментов в Android. Какие преимущества дает использование фрагментов? (ПК-4.2)

- Пример правильного ответа: Фрагменты создаются путем наследования класса Fragment и переопределения его методов жизненного цикла, таких как onCreateView и onActivityCreated. Фрагменты используются для создания многоэкранных интерфейсов в одном Activity и позволяют переиспользовать UI-компоненты. Преимущества включают гибкость в адаптации интерфейса под различные размеры экранов и возможность управления отдельными частями интерфейса независимо друг от друга.

4. Объясните, как работать с данными в Android, используя SQLite. Приведите пример создания таблицы и выполнения CRUD-операций. (ПК-4.2)

- Пример правильного ответа: SQLite — встроенная реляционная база данных в Android. Для работы с ней создается подкласс SQLiteOpenHelper, в котором определяются методы onCreate и onUpgrade. CRUD-операции включают добавление данных, чтение данных, обновление данных и удаление данных. Пример создания таблицы: создание таблицы "users" с полями "id", "name" и "age".

5. Какие ключевые особенности языка программирования Kotlin делают его предпочтительным для разработки Android-приложений? Приведите примеры. (ПК-1.1)

- Пример правильного ответа: Kotlin — современный язык программирования, полностью совместимый с Java, обладающий более лаконичным синтаксисом и дополнительными функциями, такими как null-безопасность, расширения функций и корутины для асинхронного программирования. Пример: использование ключевого слова "val" для неизменяемых переменных и оператора "?" для безопасного вызова методов.

6. Какое значение имеет механизм SharedPreferences в Android и как он используется для сохранения данных? Приведите пример использования. (ПК-1.1)

- Пример правильного ответа: SharedPreferences используется для сохранения простых данных в виде пар "ключ-значение". Он часто используется для хранения настроек пользователя. Пример: сохранение и получение имени пользователя.

7. Опишите процесс установки и настройки Android Studio для разработки Android-приложений. Какие основные настройки необходимо выполнить? (УК-1.2)

- Пример правильного ответа: Установка Android Studio начинается с загрузки установочного файла с официального сайта и его запуска. После установки необходимо настроить SDK и создать виртуальное устройство (AVD) для тестирования приложений. Важно также установить необходимые плагины и настроить проект для поддержки конкретных версий Android.

8. Какие преимущества и недостатки использования Java для разработки Android-приложений по сравнению с Kotlin? (ПК-1.1)

- Пример правильного ответа: Java является традиционным языком для разработки Android-приложений и имеет обширную базу библиотек и документации. Однако Kotlin предлагает более лаконичный синтаксис, встроенную поддержку null-безопасности и современные функции, что делает его более предпочтительным для новых проектов. Недостатки Java включают более многословный код и отсутствие некоторых современных функций.

9. Какова роль методов жизненного цикла Activity в Android-приложениях? Опишите основные методы и их использование. (ПК-4.2)

- Пример правильного ответа: Методы жизненного цикла Activity, такие как onCreate, onStart, onResume, onPause, onStop и onDestroy, позволяют управлять состоянием Activity и выполнять определенные действия на каждом этапе. Например, onCreate используется для инициализации, onResume — для подготовки Activity к взаимодействию с пользователем, а onDestroy — для освобождения ресурсов перед уничтожением Activity.

10. В чем заключается процесс тестирования и отладки Android-приложений в Android Studio? (ПК-1.2)

- Пример правильного ответа: Тестирование и отладка Android-приложений в Android Studio включают использование Logcat для просмотра логов, отладчика для пошагового выполнения кода и проверки значений переменных, а также написание и выполнение юнит-тестов и инструментальных тестов с помощью JUnit и Espresso. Эти инструменты помогают выявлять и исправлять ошибки, обеспечивая высокое качество кода.

11. Как можно реализовать инкапсуляцию и использование интерфейсов в Android-приложениях? Приведите примеры. (ПК-2.1)

- Пример правильного ответа: Инкапсуляция достигается за счет сокрытия данных и предоставления публичных методов для доступа к этим данным. Интерфейсы используются для определения контрактов между компонентами. Например, интерфейс может определять методы, которые должен реализовать класс, взаимодействующий с другим компонентом приложения. Это позволяет гибко изменять реализацию, не затрагивая код, использующий интерфейс.

12. Объясните принципы и методы рефакторинга кода в Android-приложениях. Каковы основные цели рефакторинга? (ПК-2.2)

- Пример правильного ответа: Рефакторинг кода включает улучшение структуры кода без изменения его внешнего поведения. Основные методы включают упрощение сложных методов, удаление дублирующегося кода и улучшение читаемости. Основные цели рефакторинга — повышение качества кода, улучшение его поддержки и расширяемости. Примеры методов рефакторинга: извлечение методов, переименование переменных и классов, а также разделение крупных классов на более мелкие.

13. Какие преимущества и недостатки имеет использование фрагментов для создания многоэкранных интерфейсов в Android? (ПК-4.2)

- Пример правильного ответа: Преимущества фрагментов включают возможность создания адаптивных интерфейсов, переиспользование компонентов и управление несколькими экранами в одном Activity. Недостатки могут включать сложность управления состоянием и необходимость дополнительного кода для обработки взаимодействий между фрагментами. Фрагменты также позволяют легко адаптировать интерфейс под различные устройства, такие как планшеты и смартфоны.

14. Каковы основные шаги при проектировании пользовательского интерфейса для Android-приложения? (УК-2.1)

- Пример правильного ответа: Основные шаги при проектировании пользовательского интерфейса включают определение целевой аудитории, анализ пользовательских сценариев, создание макетов и прототипов, выбор подходящих Layout'ов и компонентов, а также тестирование интерфейса на разных устройствах. Важно учитывать принципы удобства использования и доступности, чтобы обеспечить положительный пользовательский опыт.

15. Объясните процесс интеграции и использования библиотек в Android-приложениях. Какие инструменты и подходы используются для управления зависимостями? (УК-2.2)

- Пример правильного ответа: Интеграция библиотек в Android-приложениях осуществляется с помощью системы сборки Gradle. Управление зависимостями включает добавление библиотек в файл build.gradle, настройку версий и конфигураций. Инструменты Gradle позволяют автоматически загружать и обновлять библиотеки, обеспечивая согласованность версий и упрощая процесс разработки. Примеры популярных библиотек: Retrofit для работы с сетевыми запросами, Picasso для загрузки изображений и Room для работы с базами данных.

16. Каковы основные методы и инструменты тестирования Android-приложений? Приведите примеры написания юнит-тестов и инструментальных тестов. (ПК-2.1)

- Пример правильного ответа: Основные методы тестирования Android-приложений включают юнит-тестирование, интеграционное тестирование и инструментальное тестирование. Юнит-тесты проверяют отдельные компоненты приложения, а инструментальные тесты проверяют взаимодействие компонентов и пользовательский интерфейс. Примеры инструментов: JUnit для юнит-тестов и Espresso для инструментальных тестов. Юнит-тесты позволяют быстро выявлять ошибки на ранних этапах разработки, а инструментальные тесты обеспечивают высокое качество конечного продукта.

Критерии оценивания для зачета (зачет/незачет)

1. Понимание основных понятий и терминов

- **Зачет:** Студент правильно использует основные понятия и термины, объясняет их в контексте.
- **Незачет:** Студент не знает или неправильно использует основные понятия и термины, не может объяснить их.

2. Точность и полнота ответов на теоретические вопросы

- **Зачет:** Ответы на большинство вопросов полные и точные, основные аспекты тем раскрыты.
- **Незачет:** Ответы неполные, содержат значительные ошибки или отсутствуют важные аспекты тем.

3. Способность к применению знаний на практике

- **Зачет:** Студент правильно решает практические задачи, выполняет SQL-запросы без ошибок.
- **Незачет:** Студент допускает ошибки при решении практических задач, не может правильно выполнить SQL-запросы.

4. Качество выполнения практических заданий

- **Зачет:** Практические задания выполнены корректно, все необходимые элементы присутствуют.
- **Незачет:** Практические задания выполнены некорректно, содержат ошибки или неполные.

Окончательное решение по зачету:

- **Зачет:** Студент демонстрирует понимание основных понятий, корректно отвечает на теоретические вопросы и выполняет практические задания.

- **Незачет:** Студент не демонстрирует понимания основных понятий, отвечает с ошибками на теоретические вопросы и/или не справляется с практическими заданиями.

3.2 Вопросы для проведения промежуточной аттестации в форме ЭКЗАМЕНА

6 семестр

1. Объясните архитектуру компонентов Android-приложения и их взаимодействие. (УК-1.1)

- Ответ: Архитектура Android-приложения включает основные компоненты: Activity, Service, Broadcast Receiver и Content Provider. Activity представляет собой экран с пользовательским интерфейсом, Service выполняет длительные операции в фоновом режиме, Broadcast Receiver реагирует на системные широковещательные сообщения, а Content Provider управляет доступом к данным приложения. Эти компоненты взаимодействуют через Intent, который позволяет запускать компоненты и передавать данные между ними.

2. Какие преимущества и недостатки использования Java для разработки Android-приложений по сравнению с Kotlin? (ПК-1.1)

- Ответ: Преимущества Java включают широкую известность, обширную документацию и поддержку, что облегчает обучение и разработку. Недостатки — более многословный синтаксис и отсутствие современных функций, таких как null-безопасность. Kotlin, напротив, предлагает лаконичный синтаксис, встроенную поддержку null-безопасности и корутины для асинхронного программирования, что упрощает разработку и улучшает читаемость кода.

3. Опишите процесс создания и использования фрагментов в Android. Какие преимущества дает использование фрагментов? (ПК-4.2)

- Ответ: Фрагменты создаются путем наследования класса Fragment и переопределения его методов жизненного цикла, таких как onCreateView и onActivityCreated. Фрагменты используются для создания многоэкранных интерфейсов в одном Activity и позволяют переиспользовать UI-компоненты. Преимущества включают гибкость в адаптации интерфейса под различные размеры экранов и возможность управления отдельными частями интерфейса независимо друг от друга.

4. Объясните основные отличия между LinearLayout, RelativeLayout и ConstraintLayout в Android и приведите примеры их использования. (ПК-4.2)

- Ответ: LinearLayout выстраивает элементы в строку или столбец, RelativeLayout позволяет располагать элементы относительно друг друга, а ConstraintLayout предлагает гибкий способ размещения элементов с использованием ограничений. Например, LinearLayout подходит для простых списков, RelativeLayout — для сложных форм с элементами, зависящими друг от друга, а ConstraintLayout — для адаптивных интерфейсов с многоуровневыми зависимостями.

5. Какое значение имеет механизм SharedPreferences в Android и как он используется для сохранения данных? (ПК-1.1)

- Ответ: SharedPreferences используется для сохранения простых данных в виде пар "ключ-значение". Он часто используется для хранения настроек пользователя. Пример использования: сохранение и получение имени пользователя.

6. Объясните процесс установки и настройки Android Studio для разработки Android-приложений. Какие основные настройки необходимо выполнить? (УК-1.2)

- Ответ: Установка Android Studio начинается с загрузки установочного файла с официального сайта и его запуска. После установки необходимо настроить SDK и создать виртуальное устройство (AVD) для тестирования приложений. Важно также установить необходимые плагины и настроить проект для поддержки конкретных версий Android.

7. Как можно реализовать инкапсуляцию и использование интерфейсов в Android-приложениях? Приведите примеры. (ПК-2.1)

- Ответ: Инкапсуляция достигается за счет сокрытия данных и предоставления публичных методов для доступа к этим данным. Интерфейсы используются для определения

контрактов между компонентами. Например, интерфейс может определять методы, которые должен реализовать класс, взаимодействующий с другим компонентом приложения. Это позволяет гибко изменять реализацию, не затрагивая код, использующий интерфейс.

8. Объясните принципы и методы рефакторинга кода в Android-приложениях. Каковы основные цели рефакторинга? (ПК-2.2)

- Ответ: Рефакторинг кода включает улучшение структуры кода без изменения его внешнего поведения. Основные методы включают упрощение сложных методов, удаление дублирующегося кода и улучшение читаемости. Основные цели рефакторинга — повышение качества кода, улучшение его поддержки и расширяемости.

9. Каковы основные шаги при проектировании пользовательского интерфейса для Android-приложения? (УК-2.1)

- Ответ: Основные шаги при проектировании пользовательского интерфейса включают определение целевой аудитории, анализ пользовательских сценариев, создание макетов и прототипов, выбор подходящих Layout'ов и компонентов, а также тестирование интерфейса на разных устройствах. Важно учитывать принципы удобства использования и доступности, чтобы обеспечить положительный пользовательский опыт.

10. Объясните процесс интеграции и использования библиотек в Android-приложениях. Какие инструменты и подходы используются для управления зависимостями? (УК-2.2)

- Ответ: Интеграция библиотек в Android-приложениях осуществляется с помощью системы сборки Gradle. Управление зависимостями включает добавление библиотек в файл build.gradle, настройку версий и конфигураций. Инструменты Gradle позволяют автоматически загружать и обновлять библиотеки, обеспечивая согласованность версий и упрощая процесс разработки. Примеры популярных библиотек: Retrofit для работы с сетевыми запросами, Picasso для загрузки изображений и Room для работы с базами данных.

11. Какие преимущества и недостатки имеет использование фрагментов для создания многоэкранных интерфейсов в Android? (ПК-4.2)

- Ответ: Преимущества фрагментов включают возможность создания адаптивных интерфейсов, переиспользование компонентов и управление несколькими экранами в одном Activity. Недостатки могут включать сложность управления состоянием и необходимость дополнительного кода для обработки взаимодействий между фрагментами. Фрагменты также позволяют легко адаптировать интерфейс под различные устройства, такие как планшеты и смартфоны.

12. Каковы основные методы и инструменты тестирования Android-приложений? Приведите примеры написания юнит-тестов и инструментальных тестов. (ПК-2.1)

- Ответ: Основные методы тестирования Android-приложений включают юнит-тестирование, интеграционное тестирование и инструментальное тестирование. Юнит-тесты проверяют отдельные компоненты приложения, а инструментальные тесты проверяют взаимодействие компонентов и пользовательский интерфейс. Примеры инструментов: JUnit для юнит-тестов и Espresso для инструментальных тестов. Юнит-тесты позволяют быстро выявлять ошибки на ранних этапах разработки, а инструментальные тесты обеспечивают высокое качество конечного продукта.

13. Каковы основные функции и задачи сервисов в Android? Приведите примеры их использования. (ПК-1.1)

- Ответ: Сервисы в Android выполняют длительные операции в фоновом режиме, такие как воспроизведение музыки, выполнение сетевых запросов или обновление данных. Примеры использования: MusicService для воспроизведения музыки в фоне или SyncService для синхронизации данных с сервером. Сервисы могут запускаться и выполняться независимо от пользовательского интерфейса.

14. Объясните принципы работы с сетью в Android. Как осуществляется обработка HTTP-запросов? (ПК-4.1)

- Ответ: Работа с сетью в Android осуществляется через HTTP-запросы, которые отправляются с помощью библиотек, таких как Retrofit или OkHttp. Обработка запросов включает создание клиентских объектов, настройку запросов и обработку ответов. Библиотеки помогают упростить работу с сетью, управлять асинхронными операциями и обрабатывать JSON-данные.

15. Какие подходы используются для управления зависимостями в Android-приложениях? (ПК-4.2)

- Ответ: Управление зависимостями в Android-приложениях осуществляется с помощью системы сборки Gradle. Gradle позволяет указать зависимости в файле build.gradle, который автоматически загружает и обновляет библиотеки. Подходы включают управление версиями библиотек, настройку конфигураций и модульность, что позволяет лучше контролировать и обновлять зависимости в проекте.

16. Какие методы и инструменты используются для профилирования и оптимизации производительности Android-приложений? (ПК-1.2)

- Ответ: Методы профилирования включают использование инструментов, таких как Android Profiler и Systrace, для анализа производительности приложения. Эти инструменты позволяют отслеживать использование памяти, CPU, сети и других ресурсов. Оптимизация включает устранение утечек памяти, улучшение скорости отклика пользовательского интерфейса и снижение времени загрузки приложения.

17. Каковы основные этапы публикации Android-приложения в Google Play Store? (ПК-4.1)

- Ответ: Основные этапы включают создание учетной записи разработчика в Google Play Console, подготовку APK или AAB файла, создание страницы приложения с описанием и скриншотами, настройку цен и доступности, а также прохождение процесса проверки приложения. После успешной проверки приложение становится доступным для загрузки пользователями через Google Play Store.

18. Как реализовать мультимедиа-функциональность в Android-приложениях? Примеры работы с аудио и видео. (ПК-1.1)

- Ответ: Реализация мультимедиа-функциональности включает использование классов MediaPlayer и ExoPlayer для воспроизведения аудио и видео, а также классов AudioRecord и MediaRecorder для записи. Примеры включают воспроизведение музыки из локальных файлов или потоков, запись звука через микрофон и воспроизведение видео с использованием SurfaceView или TextureView.

19. Объясните, как использовать библиотеку Retrofit для работы с сетевыми запросами в Android. (ПК-4.1)

- Ответ: Retrofit — это типобезопасная HTTP-клиентская библиотека для Android и Java, разработанная Square. Она упрощает процесс взаимодействия с веб-сервисами. Retrofit использует аннотации для описания запросов и обработки ответов. Для использования Retrofit создается интерфейс, в котором описываются методы HTTP-запросов (GET, POST и т.д.) с аннотациями, а затем создается экземпляр Retrofit, который реализует этот интерфейс.

20. Каковы основные принципы работы с базами данных в Android с использованием Room? (ПК-4.2)

- Ответ: Room — это библиотека для работы с базами данных в Android, которая предоставляет абстракцию поверх SQLite. Основные компоненты Room включают Entity (таблица базы данных), DAO (Data Access Object) для выполнения запросов и Database для управления подключением к базе данных. Room упрощает работу с базами данных за счет использования аннотаций и автоматической генерации кода, обеспечивая при этом безопасность типов и проверку на этапе компиляции.

21. Объясните, как использовать WorkManager для выполнения фоновых задач в Android. (ПК-4.1)

- Ответ: **WorkManager** — это библиотека для управления фоновой работой, которая должна выполняться надежно. **WorkManager** выбирает подходящий способ выполнения задач в зависимости от состояния устройства и требований к задаче. Для использования **WorkManager** создается класс, который наследуется от **Worker** и реализует метод **doWork()**. Затем создается и запускается экземпляр **WorkRequest**, который управляет задачей.

22. Каковы основные компоненты архитектуры MVVM и их роли в разработке Android-приложений? (ПК-4.2)

- Ответ: **MVVM (Model-View-ViewModel)** — это архитектурный шаблон, который разделяет логику представления и бизнес-логику. **Model** представляет данные и бизнес-логику, **View** отвечает за отображение данных и взаимодействие с пользователем, а **ViewModel** посредничает между **Model** и **View**, управляя данными и состоянием представления. **MVVM** упрощает тестирование и поддержку кода, обеспечивая четкое разделение обязанностей.

23. Как работать с уведомлениями в Android и какие основные классы используются для их создания и управления? (ПК-1.1)

- Ответ: Для работы с уведомлениями в **Android** используется класс **NotificationCompat.Builder** для создания уведомлений, класс **NotificationManager** для их управления и **NotificationChannel** для настройки каналов уведомлений. Уведомления могут содержать текст, изображения, действия и другие элементы. Они используются для информирования пользователей о важных событиях и могут быть настроены на отображение в шторке уведомлений, всплывающих окна и т.д.

24. Объясните, как использовать библиотеку Glide для загрузки и отображения изображений в Android-приложениях. (ПК-4.1)

- Ответ: **Glide** — это библиотека для загрузки и отображения изображений в **Android**. Она поддерживает асинхронную загрузку, кэширование и обработку изображений. Для использования **Glide** создается запрос загрузки с помощью метода **Glide.with(context).load(url).into(imageView)**. Библиотека автоматически обрабатывает загрузку изображений из сети, кэширование и отображение в **ImageView**, обеспечивая высокую производительность и минимальные затраты ресурсов.

Критерии оценки ответов на экзамене

- **"Отлично" (5 баллов)**
 - Критерии:
 - Полное и точное объяснение вопроса.
 - Ответ включает все ключевые аспекты и детали.
 - Примеры, если требуются, приведены и правильно объяснены.
 - Ответ демонстрирует глубокое понимание темы.
- **"Хорошо" (4 балла)**
 - Критерии:
 - Корректное объяснение вопроса.
 - Ответ охватывает основные аспекты, но может отсутствовать незначительная деталь или пример.
 - Демонстрируется хорошее, но не полное понимание темы.
- **"Удовлетворительно" (3 балла)**
 - Критерии:
 - Общее представление о вопросе.
 - Ответ включает основные аспекты, но содержит неточности или пропуски.
 - Примеры, если требуются, могут отсутствовать или быть неверно объяснены.
 - Демонстрируется базовое понимание темы.
- **"Неудовлетворительно" (2 балла)**
 - Критерии:

- Некорректное или неполное объяснение вопроса.
- Отсутствие ключевых аспектов и деталей.
- Примеры, если требуются, отсутствуют или приведены неверные.
- Ответ демонстрирует недостаточное понимание темы.