

Специализированная магистратура

Разработка приложений для Android



Специализированная магистратура Разработка приложений для Android

- » Формат: онлайн
- » Продолжительность: 12 месяцев
- » Учебное заведение: TECH Технологический университет
- » Расписание: по своему усмотрению
- » Экзамены: онлайн

Веб-доступ: www.techtitute.com/ru/information-technology/professional-master-degree/master-android-application-development

Оглавление

01

Презентация

стр. 4

02

Цели

стр. 8

03

Компетенции

стр. 14

04

Руководство курса

стр. 18

05

Структура и содержание

стр. 26

06

Методология

стр. 38

07

Квалификация

стр. 46

01

Презентация

Сегодня в мире насчитывается более 3000 миллионов устройств на базе Android, и, как следствие, сектор разработки мобильных приложений переживает головокружительный рост. Отчеты рынка труда подтверждают растущий спрос на специалистов в области разработки мобильных приложений, оценивая полный опыт в жизненном цикле разработки, распространения и монетизации приложений. Нелегко найти профессионалов, знающих язык программирования Android, архитектуры и пользовательские интерфейсы. Именно поэтому данная программа была разработана для того, чтобы предоставить заинтересованным лицам все знания, которые помогут им продвинуться по карьерной лестнице, получив специализированный профиль. В полностью онлайн-режиме и получить квалификацию за 12 месяцев.



“

Специализируйтесь на ключевых аспектах клиентского опыта для разработки приложений на Android, чтобы они лучше выходили на рынок”

В рамках программы "Разработка приложений для Android" студенты смогут специализироваться на одной из стандартных технологий с наибольшим профессиональным спросом в наши дни. Они получают необходимые знания для самостоятельной разработки приложений для устройств на этой операционной системе от проектирования до производства. А также получают специализированные знания о жизненном цикле приложений для Android, чтобы работать более эффективно. Для этого TECH предоставил команду высококвалифицированных профессионалов, обладающих необходимым опытом, чтобы поделиться реальными случаями, которые ведут к более глубокому и эффективному обучению. Профессионалы в области программирования, разработки приложений и новых технологий, которые будут сопровождать специалиста в процессе обучения. С надежной платформы, с интерактивным контентом, распределенным в видео, графиках, примерах, основанных на проблемах, аудиовизуальных материалах и с большим сообществом для обсуждения поднятых вопросов.

Таким образом, студент может иметь отношение к таким областям, как компьютерные или веб-технологии или программное обеспечение, развитие бизнеса, маркетинг и продажи, консультирование, мультиплатформенные мобильные технологии, бизнес, автомобильный мир, домашняя автоматизация, Интернет вещей (IoT), банковское дело и дроны. Вы сможете применить свои знания на благо любого из этих секторов. Кроме того, TECH предлагает лучшую методику обучения, основанную на технологии *Relearning* и доступную полностью в режиме онлайн, что избавляет от необходимости посещать учебные классы или типичных ограничений, связанных с заранее составленным расписанием. Таким образом, за 12 месяцев студенты приобретут необходимую базу для выполнения консультационных задач по всем аспектам, связанным с мобильными технологиями, или для развития собственного бизнеса, основанного на разработке приложений.

Кроме того, профессионал получит доступ к 10 эксклюзивным и дополнительным мастер-классам, которые проводит всемирно известный эксперт в области разработки приложений для Android. Таким образом, TECH предлагает студентам уникальный опыт обучения, обеспечивая их необходимыми инструментами для успешного решения любых задач. Благодаря комплексному мультимедийному подходу студент получает гарантию наилучших академических результатов, обеспечивая полную готовность к успешной работе в постоянно развивающейся сфере.

Данная **Специализированная магистратура в области разработки приложений для Android** содержит самую полную и современную образовательную программу на рынке. Основными особенностями обучения являются:

- ♦ Разбор практических случаев, представленных экспертами в области параллельных и распределенных вычислений
- ♦ Наглядное, схематичное и исключительно практическое содержание курса предоставляет практическую информацию по тем дисциплинам, которые необходимы для осуществления профессиональной деятельности
- ♦ Практические упражнения для самооценки, контроля и улучшения успеваемости
- ♦ Особое внимание уделяется инновационным методологиям
- ♦ Теоретические занятия, вопросы эксперту и самостоятельные работы
- ♦ Учебные материалы курса доступны с любого стационарного или мобильного устройства с выходом в интернет



Погрузитесь в уникальный опыт вместе с TECH! У вас будет возможность посетить 10 уникальных и инновационных мастер-классов, разработанных всемирно известным специалистом в области разработки приложений для Android"

“

Вы получите необходимые знания для самостоятельной разработки приложений для Android-устройств от дизайна до производства”

В преподавательский состав программы входят профессионалы отрасли, признанные специалисты из ведущих сообществ и престижных университетов, которые привносят в обучение опыт своей работы.

Мультимедийное содержание программы, разработанное с использованием новейших образовательных технологий, позволит специалисту проходить обучение с учетом контекста и ситуации, т.е. в симулированной среде, обеспечивающей иммерсивный учебный процесс, запрограммированный на обучение в реальных ситуациях.

Структура этой программы основана на проблемно-ориентированном обучении, с помощью которого специалисты должны пытаться решить различные ситуации из профессиональной практики, возникающие на протяжении учебной программы. В этом специалисту будет помогать инновационная система интерактивных видеоматериалов, созданная признанными и опытными экспертами.

Воспользуйтесь всеми преимуществами передовой технологической среды, предлагаемой TECH. Это 100% онлайн-программа, где вы будете общаться с ведущими экспертами.

Предоставляйте эффективные решения своим работодателям или собственной компании.

С управлением основами, связанными с разработкой приложений на системах Android.



Android Studio
Version 3.3

Android Studio project

02

Цели

Это значительный процесс роста, который приводит к безостановочному развитию новых технологий и устройств. Эти достижения требуют постоянной подготовки специалистов в области информационных технологий и инженерного дела. Рынок не стоит на месте, и чтобы не отставать от потребностей, необходимо, чтобы профессионалы находились на переднем крае концепций и стилей, которыми они владеют. Эта программа по разработке приложений для Android обучает всем инструментам и основам, необходимым для развития в этой карьере и предоставления эффективных решений в профессиональной среде.



“

Ставьте перед собой собственные задачи в экосистеме технологий Android. Начните делать шаги к успешной карьере разработчика приложений”



Общие цели

- ♦ Определить структурные элементы системы Android
- ♦ Проанализировать различия между различными *фреймворками* разработки, их сильные и слабые стороны
- ♦ Выяснить, как создавать стили и темы, включая виртуальную реальность с помощью *компоновки*
- ♦ Анализировать различные формы управления потоком кода в Kotlin
- ♦ Развивать передовые навыки и лучшие практики разработки приложений на языке программирования Kotlin
- ♦ Разработать методику оптимального управления данными на устройстве
- ♦ Проанализировать сценарии использования Android-устройств на рынке
- ♦ Освоить элементы отзывчивого дизайна и преодолеть вытекающие из него проблемы
- ♦ Составить различные этапы цикла непрерывной интеграции, ориентированного на разработку на Android



Станьте экспертом в области разработки приложений на Android Запишитесь сейчас на эту программу и получите диплом в 100% дистанционном режиме через 12 месяцев"





Конкретные цели

Модуль 1. Язык программирования Android

- ♦ Изучить ядро Linux и виртуальную машину, лежащую в основе Android
- ♦ Проанализировать встроенные библиотеки системы
- ♦ Определить преимущества Android перед другими платформами
- ♦ Выявить элементы приложений для Android
- ♦ Представить версии Android и их усовершенствования
- ♦ Оценить рынок приложений для Android
- ♦ Обосновать будущую эволюцию Android

Модуль 2. Фреймворки, используемые при разработке приложений для Android

- ♦ Проанализировать основной *фреймворк* для Android
- ♦ Разработать другие *фреймворки*, используемые для разработки Android-приложений
- ♦ Внедрить библиотеки с помощью *Graddl*
- ♦ Создать *фреймворки* для подключения к программному интерфейсу приложения
- ♦ Сформировать специализированные знания об архитектурах, таких как MVP
- ♦ Понять плюсы и минусы MVP и MVVM

Модуль 3. Интерфейсы и компоновка в процессе разработки приложений для Android

- ♦ Представить модель жизненного цикла представлений в Android
- ♦ Исследовать наиболее важные атрибуты визуального дизайна (*компоновка*)
- ♦ Анализировать доступные дизайны макетов
- ♦ Генерировать дизайн *компоновки* многократного использования
- ♦ Определить, как использовать альтернативные ресурсы
- ♦ Обозначить различия в использовании этих компонентов по сравнению с другими системами программирования
- ♦ Установить потенциал и использование файла AndroidManifest.xml

Модуль 4. Язык программирования приложений для Android Kotlin

- ♦ Развить язык программирования Kotlin
- ♦ Рассмотреть особенности и отличительные возможности языка Kotlin
- ♦ Изучить базовую модель выполнения программы на языке Kotlin
- ♦ Анализировать синтаксис языка и структуру программы на Kotlin
- ♦ Уточнить модель типов и переменных в Kotlin
- ♦ Определить различные формы управления потоком кода в Kotlin
- ♦ Обозначить модель классов, коллекций и объектов в Kotlin
- ♦ Получить специализированные знания о том, как использовать наследования в Kotlin
- ♦ Конкретизировать модель обработки исключений и нулевых типов в Kotlin

Модуль 5. Язык программирования приложений для Android. Расширенные возможности Kotlin в области общности, функционального программирования и параллелизма

- ♦ Изучить ковариантную и контравариантную модель общности в Kotlin
- ♦ Проанализировать модель функционального программирования Kotlin с помощью Lambdas
- ♦ Конкретизировать функции высшего порядка в Kotlin
- ♦ Разработать расширение и сопутствующие объекты в Kotlin
- ♦ Изучить использование паттерна *делегирования* в Kotlin
- ♦ Составить аннотации и отражение в Kotlin
- ♦ Углубиться в модель *тестирования* в Kotlin
- ♦ Определить различные модели асинхронного программирования, доступные в Kotlin
- ♦ Выявить модели корутинов в Kotlin
- ♦ Компилировать различные библиотеки и утилиты экосистемы Kotlin

Модуль 6. Управление данными на устройствах Android

- ♦ Анализировать различные методы управления данными на устройствах Android
- ♦ Предложить методы оптимального использования данных на устройстве
- ♦ Определить инструменты, необходимые для оптимизации данных
- ♦ Изучить особенности JSON и XML для управления данными в Android
- ♦ Оценить общую проблематику распределенных систем, применимую к миру приложений для мобильных устройств
- ♦ Определить использование библиотеки Room в качестве абстракции для использования SQLite на Android, ее преимущества и недостатки
- ♦ Обозначить разрешения безопасности, необходимые для управления данными в любом из методов, используемых в Android

Модуль 7. Инструменты для Android-устройств

- ♦ Составить подборку самых передовых инструментов для ежедневного использования
- ♦ Оценить инструменты управления Android-устройствами
- ♦ Продемонстрировать преимущества использования системы Android в дронах
- ♦ Указать функциональные возможности инструментов CRM
- ♦ Продемонстрировать преимущества использования Android-устройств в банковских платформах
- ♦ Проанализировать возможности IoT и платформ Android
- ♦ Изучить инструменты для повышения эффективности процессов на Android

Модуль 8. Адаптивный дизайн на Android

- ♦ Анализировать основные элементы дизайна
- ♦ Определить методику дизайна визуальных элементов и методику дизайна экрана
- ♦ Решить различные проблемы, возникающие при разработке для нескольких устройств
- ♦ Овладеть инструментами для создания лучших и более эффективных ресурсов для разработки на нескольких устройствах
- ♦ Изучить альтернативные *фреймворки* для нативной адаптивной разработки
- ♦ Сформировать собственную методику разработки приложений с использованием лучших практик для отображения на нескольких устройствах с самого начала проекта

Модуль 9. Маркетинг в приложениях для Android

- ♦ Проанализировать новые методики, ориентированные на пользователя
- ♦ Определить, как искусственный интеллект вывел CX на новый уровень
- ♦ Установить важность доступности и мобильности
- ♦ Разработать современные методы анализа пользовательских сеансов и поведенческих факторов
- ♦ Определить цели микроперсонализации на протяжении пользовательского путешествия
- ♦ Составить новые собственные стратегии для меняющейся и динамичной среды
- ♦ Предложить техники создания прототипов

Модуль 10. Жизненный цикл приложения для Android. Облако, Playstore и версионирование

- ♦ Определить преимущества использования автоматизированной модели распространения релизов
- ♦ Установить различия между непрерывной интеграцией, непрерывной доставкой и непрерывным распространением
- ♦ Определить основные особенности DevOps
- ♦ Произвести оценку некоторых ключевых инструментов для внедрения конвейеров CI/CD
- ♦ Определить ключевые факторы для разработки приложений, готовых к поддержке процессов CI/CD
- ♦ Изучить контейнерные технологии как фундаментальную основу практики CI/CD
- ♦ Определить практики, сценарии использования, технологии и инструменты версионной экосистемы CI/CD

03 Компетенции

Нелегко найти специалистов, знающих язык программирования Android, архитектуры и пользовательские интерфейсы в зависимости от типа устройства, на котором выполняется приложение. Получив степень Специализированной магистратуры в области разработки приложений для Android, студенты смогут специализироваться в одной из стандартных технологий с наибольшей профессиональной проекцией в наши дни. Вы получите необходимые знания для самостоятельной разработки приложений для Android-устройств от дизайна до производства. Вы приобретете специализированные знания о жизненном цикле приложений для Android, чтобы более эффективно работать в качестве разработчика приложений для мобильных устройств.



“

Пройдя эту Специализированную магистратуру, вы сможете продвинуться по карьерной лестнице в сфере бизнеса или фриланса. Разрабатывая приложения, которые добиваются успеха на рынке”



Общие профессиональные навыки

- ♦ Разрабатывать мобильные приложения для устройств с операционной системой Android
- ♦ Освоить структурные элементы системы Android
- ♦ Анализировать различные *фреймворки* для использования в многочисленных архитектурах
- ♦ Определять свойства интерфейса с точки зрения практичности
- ♦ Развить специальные знания о языке программирования Kotlin и контексте его появления
- ♦ Управлять *поставщиком контента* для обмена данными и его безопасностью
- ♦ Изучить новые методологии, ориентированные на пользователя
- ♦ Анализировать элементы отзывчивого дизайна
- ♦ Изучить новые методологии, ориентированные на пользователя
- ♦ Настроить автоматизированные конвейеры на экосистему Android

“

Сочетать все эти компетенции в вашем профессиональном профиле повысит ценность ваших предложений и сделает вас более востребованным на рынке труда”





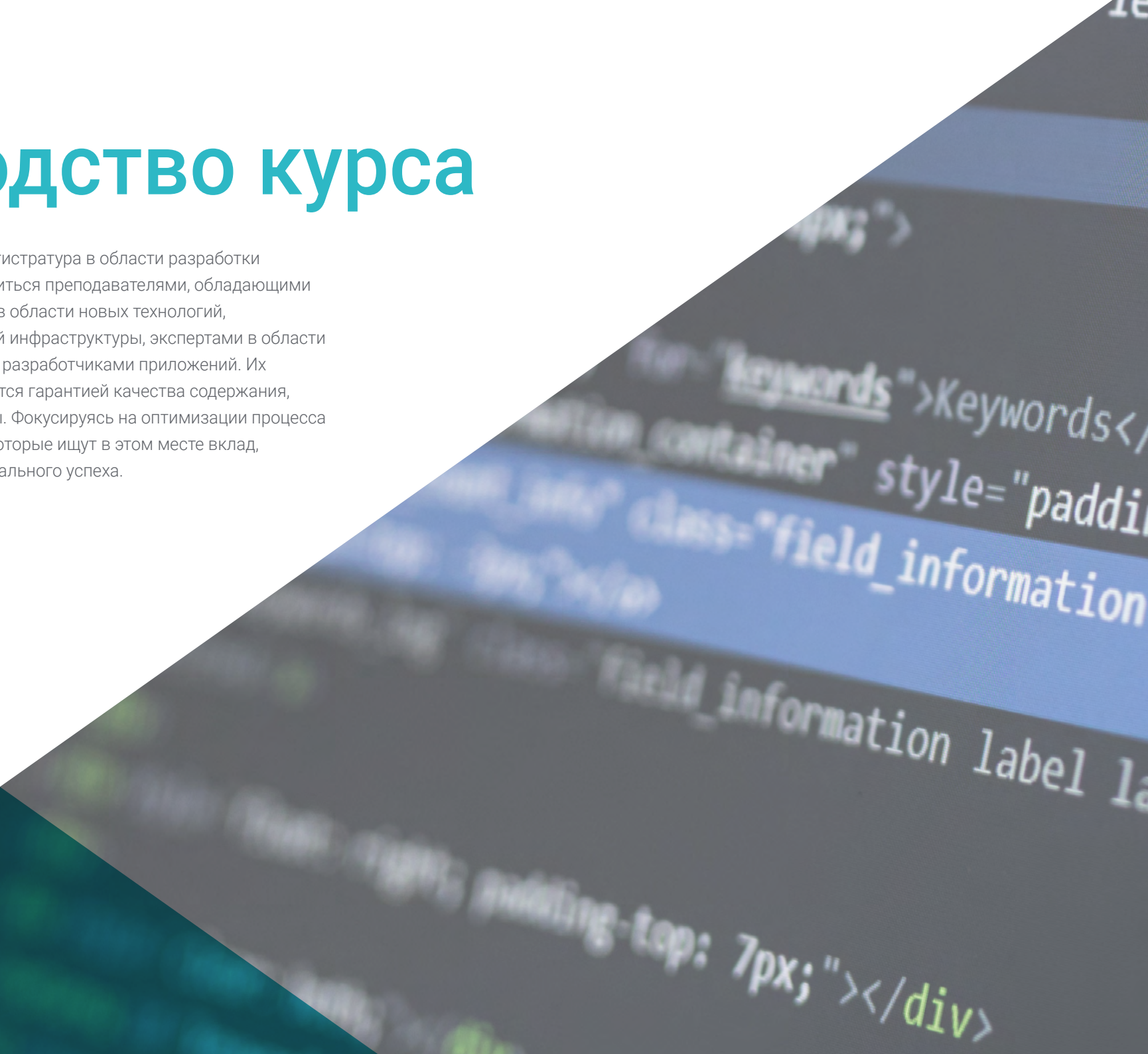
Профессиональные навыки

- ♦ Использовать альтернативные ресурсы, анализируя доступную *компановку*
- ♦ Изучить различные формы управления потоком кода в Kotlin и их возможности
- ♦ Повысить производительность приложений за счет дифференцированных возможностей языка Kotlin
- ♦ Разработать расширение и сопутствующие объекты в Kotlin
- ♦ Использовать базы данных или сетевые сервисы для создания файлов в различных форматах
- ♦ Применить библиотеку Room в качестве абстракции для использования SQLite на Android
- ♦ Оценить инструменты управления устройствами Android, анализируя возможности IoT и платформы Android
- ♦ Применять адаптивный дизайн для обеспечения более приятного пользовательского опыта
- ♦ Сочетать методы прототипирования и *вайрфрейма* с новыми революционными технологиями, такими как искусственный интеллект и интернет вещей

04

Руководство курса

Данная Специализированная магистратура в области разработки приложений для Android руководиться преподавателями, обладающими обширными знаниями и опытом в области новых технологий, архитектуры решений и цифровой инфраструктуры, экспертами в области программирования для Android и разработчиками приложений. Их обширные учебные планы являются гарантией качества содержания, отобранного для этой программы. Фокусируясь на оптимизации процесса обучения для профессионалов, которые ищут в этом месте вклад, необходимый для их профессионального успеха.



“

Профессиональные эксперты
в области разработки приложений
и технологической области будут
сопровождать вас на протяжении
всего процесса обучения,
предоставляя вам лучший материал”

Приглашенный руководитель международного уровня

Колин Ли — успешный разработчик мобильных приложений, специализирующийся на **нативном коде для Android**, чье влияние распространяется на весь мир. Эксперт является авторитетом в **районе Twin Cities** и в **обращении с Kotlin**. Один из его последних вкладов — демонстрация на живом коде, как **быстро создать браузер**, используя вышеупомнутый язык программирования и компоненты браузера Mozilla с открытым исходным кодом для Android.

Кроме того, его приложения связаны с крупными мировыми компаниями, например, ему было поручено создать **цифровые решения для Pearson**, одного из крупнейших мировых издательств. Также он создал **видеорегистратор для Android** для стартапа Flipgrid, позже приобретенного Microsoft.

Он также создал успешную VPN для Android для крупного **консалтингового клиента**. В тоже время, он является создателем инструмента управления грузоперевозками, внедренного транснациональной компанией **Amazon** для облегчения работы ее контрактных дальнбойщиков. Участвовал в создании **мобильных версий браузера Firefox** для компании Mozilla.

Сегодня он работает в качестве подрядчика, **включая анализ кода и проверку безопасности**. Его влияние на разработку мобильных приложений и многолетний опыт делают его ведущей фигурой на мировой технологической арене.



Д-р Ли, Колин

- Старший инженер Android в компании Meetup, Миннеаполис, США
- Директор в ColinTheShots LLC
- Инженер-программист Android в компании Specto Inc.
- Старший инженер Android для Mozilla
- Инженер-разработчик программного обеспечения для Amazon
- Инженер по мобильным приложениям для Flipgrid
- Специалист по настройке программного обеспечения в компании Pearson VUE
- Степень бакалавра в Университете Флориды

“

*Благодаря TECH вы
сможете учиться у лучших
мировых профессионалов”*

Руководство



Г-н Олая Бональ, Мартин

- ♦ Старший менеджер практики *блокчейн* в компании EY
- ♦ Технический специалист по *блокчейн*-клиентам в IBM
- ♦ Директор по архитектуре компании Blocknitive
- ♦ Координатор группы по нереляционным распределенным базам данных в WedoIT (дочерняя компания IBM)
- ♦ Архитектор инфраструктуры в Bankia
- ♦ Руководитель отдела верстки в компании T-Systems
- ♦ Координатор отдела Bing Data España SL

Преподаватели

Г-жа Мартинес Серрато, Йесика

- ♦ Специалист в области образования, бизнеса и маркетинга
- ♦ Менеджер по техническому обучению в Securitas Seguridad España
- ♦ *Менеджер продукции* в области электронной безопасности в Securitas Seguridad España
- ♦ Аналитик бизнес-аналитики в Ricopia Technologies
- ♦ Специалист по информатике и ответственная за компьютерные классы организмов технической подготовки в Университете Алькала-де-Энарес
- ♦ Сотрудник Ассоциации ASALUMA
- ♦ Степень бакалавра в области инженерии электронных коммуникаций в Политехнической школе Университета Алькала-де-Энарес

Г-н Грильо Эрнандес, Хосе Энрике

- ♦ Разработчик приложений и технологический аналитик
- ♦ Старший разработчик мобильных приложений в компании Globant
- ♦ Разработчик Android в Plexus Tech
- ♦ Старший Android-разработчик в RoadStr
- ♦ Старший разработчик мобильных приложений в Avantgarde IT-Information Technology Services
- ♦ Руководитель проекта в Smartdessa
- ♦ Разработчик в Educatablet
- ♦ Технологический аналитик в Corporate Mobile Solutions
- ♦ Степень магистра в области системной инженерии, полученная в Университете Симона Боливара

Г-н Ногера Родригес, Пабло

- ♦ Инженер-программист, занимающийся *блокчейн*-решениями в компании EY
- ♦ Разработчик мобильных приложений в Bitnovo
- ♦ Разработчик нативных приложений для iOS в компании Umani and Stef
- ♦ Внештатный программист, создатель приложения Aviaze в сотрудничестве с компанией Starman Aviation

Г-н Герреро Диас-Пинтадо, Артуро

- ♦ Директор по работе с клиентами в IBM
- ♦ Технический инженер по предпродажной подготовке в рамках портфеля *Watson Customer Engagement*
- ♦ Сетевой инженер НИОКР в компании Telefónica
- ♦ Степень бакалавра в области телекоммуникационной техники Университета Алькала и Датского технического университета

Г-н Сентено Мартин-Ромеро, Альфонсо

- ♦ Консультант по кибербезопасности в компании Deloitte
- ♦ Стажер в отделе коммерческого планирования и бизнес-аналитики в Telefónica Tech
- ♦ Стажер в B2B Market Intelligence
- ♦ Стажер в отделе финансовых деривативов и анализа акций в Ahorro Corporación Financiera
- ♦ Двойная степень бакалавра в области делового администрирования и управления туризмом в Папском университете Комильяс
- ♦ Буткемп по кибербезопасности в Ironhack Technology Study Centre

Г-н Арранс Торрес, Эктор

- ♦ Руководитель проекта по *разработке программного обеспечения* в компании Ezenit
- ♦ Член операционной группы в компании *Cink Venturing*
- ♦ Преподаватель курсов, связанных с разработкой цифровых проектов
- ♦ Выпускник факультета компьютерной инженерии Мадридского университета Комплутенсе
- ♦ Степень магистра делового администрирования в области менеджмента и управления бизнесом в университете The Power MBA

Г-жа Порталатин Ромеро, Изабель

- ♦ IT-инженер
- ♦ Ответственная за предложения в области информационных технологий для различных государственных и частных организаций
- ♦ Онлайн-преподаватель по различным программам профессионального обучения
- ♦ Технический инженер в области компьютерного менеджмента, Политехническая школа информационных технологий Университета Эстремадуры

Г-н Наталь Родригес, Даниэль

- ♦ Компьютерный инженер, специализирующийся на Appian
- ♦ Разработчик Appian в Vass, консалтинговой компании, специализирующейся на IT-решениях
- ♦ Разработчик платформ в телекоммуникационной компании Telefónica
- ♦ Java-программист в компании Babel
- ♦ Степень бакалавра в области компьютерной инженерии в Высшей технической школе инженеров-информатиков
- ♦ Информатик в Мадридском политехническом университете

Д-р Перальта Мартин-Паломино, Артуро

- ♦ CEO и CTO Prometheus Global Solutions
- ♦ CTO в Korporate Technologies
- ♦ CTO в AI Shephers GmbH
- ♦ Консультант и советник в области стратегического бизнеса в Alliance Medical
- ♦ Руководитель в области проектирования и разработки в компании DocPath
- ♦ Руководитель в области компьютерной инженерии в Университете Кастилии-ла-Манча
- ♦ Степень доктора в области экономики, бизнеса и финансов Университета Камило Хосе Села
- ♦ Степень доктора в области психологии Университета Кастилии-ла-Манча
- ♦ Степень магистра Executive MBA Университета Изабель I
- ♦ Степень магистра в области управления коммерцией и маркетингом Университета Изабель I
- ♦ Степень магистра в области больших данных по программе Hadoop
- ♦ Степень магистра в области передовых информационных технологий Университета Кастилии-Ла-Манча
- ♦ Член: Исследовательская группа SMILE

Г-н Госало Фернандес, Хуан Луис

- ♦ Директор по блокчейн DevOps в Аластрии
- ♦ Директор по технологиям уровня обслуживания в Santander, Испания
- ♦ Руководитель направления разработки мобильных приложений Tinkerlink в компании Cronos Telecom
- ♦ Директор по технологиям управления ИТ-услугами в Barclays Bank Spain
- ♦ Степень бакалавра в области высшей компьютерной инженерии в UNED
- ♦ Специализация в области *глубокого обучения* в DeepLearning.ai





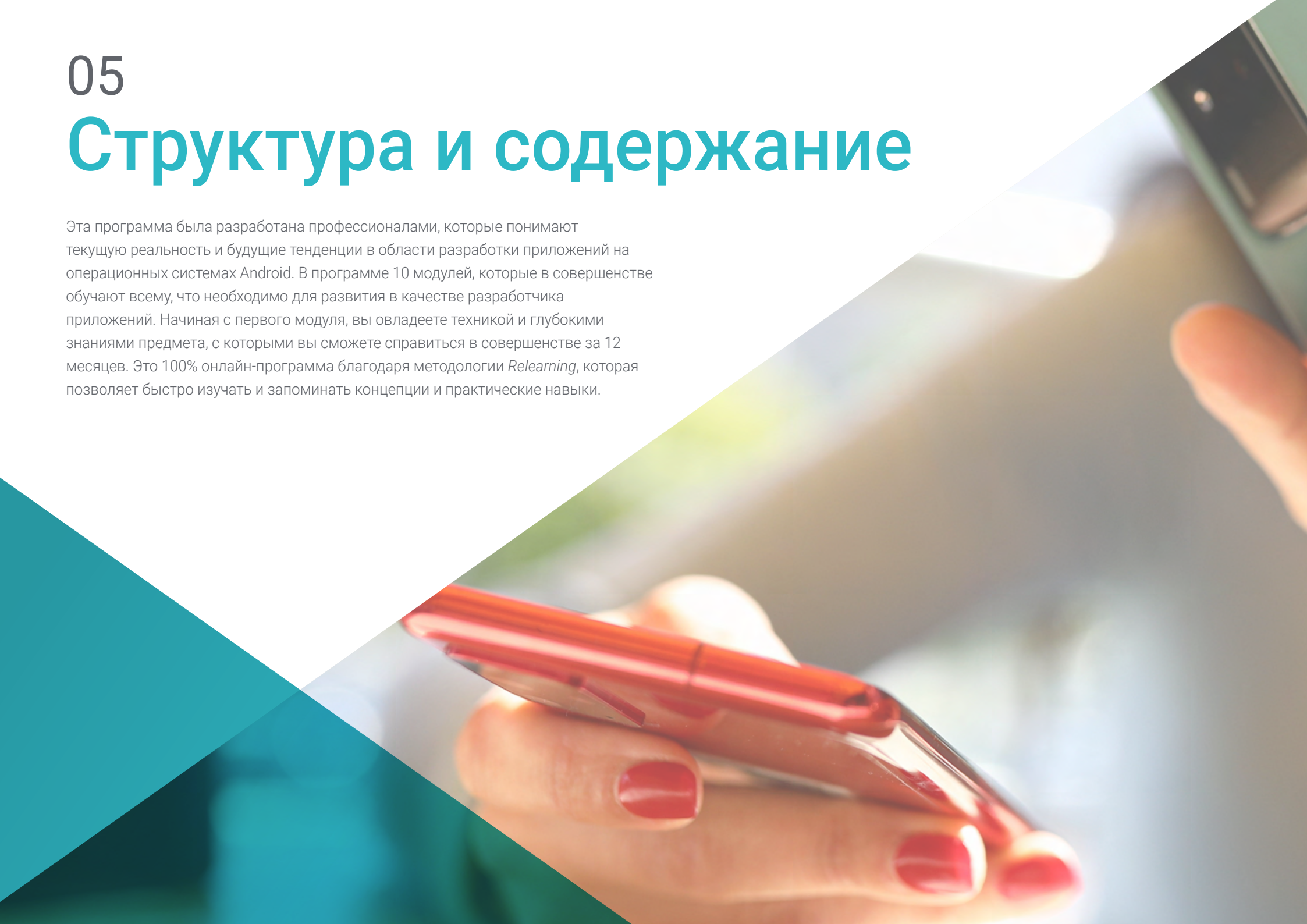
“

Воспользуйтесь возможностью
узнать о последних
достижениях в этой области,
чтобы применить их в своей
повседневной практике”

05

Структура и содержание

Эта программа была разработана профессионалами, которые понимают текущую реальность и будущие тенденции в области разработки приложений на операционных системах Android. В программе 10 модулей, которые в совершенстве обучают всему, что необходимо для развития в качестве разработчика приложений. Начиная с первого модуля, вы овладеете техникой и глубокими знаниями предмета, с которыми вы сможете справиться в совершенстве за 12 месяцев. Это 100% онлайн-программа благодаря методологии *Relearning*, которая позволяет быстро изучать и запоминать концепции и практические навыки.



“

Динамичные материалы с
практическими упражнениями,
видеороликами и интерактивными
форматами обеспечит быстрое и
легко усваиваемое обучение”

Модуль 1. Язык программирования Android

- 1.1. Платформа Android
 - 1.1.1. Платформа Android
 - 1.1.2. Операционная система Android
 - 1.1.3. *Альянс открытых стандартов* в разработке Android
- 1.2. Архитектура Android
 - 1.2.1. Архитектурные элементы системы Android
 - 1.2.2. Связь между элементами
 - 1.2.3. Расширяемость архитектуры Android
 - 1.2.4. Управление механическими ресурсами: батареей и памятью
 - 1.2.5. Эмуляторы Android
- 1.3. Ядро Android Linux
 - 1.3.1. Состав ядра
 - 1.3.2. Структурные элементы ядра
 - 1.3.3. Виртуальная машина Dalvik
 - 1.3.4. Виртуальная машина Android RunTime (ART)
- 1.4. Нативные библиотеки Android
 - 1.4.1. Нативные библиотеки Android
 - 1.4.2. Библиотеки совместимости (*библиотека поддержки*)
 - 1.4.3. Нативные библиотеки и расширяемость
- 1.5. Файловая система и система данных в Android
 - 1.5.1. Структура типичного приложения для Android
 - 1.5.2. Файловые системы YAFFS2 и ext4
 - 1.5.3. Использование SQLite и Room для управления данными
- 1.6. Безопасность Android
 - 1.6.1. Система разрешений
 - 1.6.2. Цифровые подписи в *пакетах приложений* для Android (APK)
 - 1.6.3. Выполнение процессов в ядре
 - 1.6.4. Потоки выполнения и события
- 1.7. Структурные компоненты стандартного приложения
 - 1.7.1. Представление (*View*)
 - 1.7.2. Деятельность (*Activity*)
 - 1.7.3. Фрагмент (*Fragment*)

- 1.7.4. Сервис (*Service*)
- 1.7.5. Намерение (*Intent*)
- 1.7.6. *Получатель трансляций и поставщик контента*
- 1.7.7. Управление данными и предпочтения пользователей
- 1.8. Версии Android
 - 1.8.1. Версии Android
 - 1.8.2. Установка версий Android
 - 1.8.3. Распространение дистрибутива Android
 - 1.8.4. Android vs. Apple iOS и другие мобильные системы
- 1.9. Android для транспортных средств
 - 1.9.1. Android и автомобильный мир
 - 1.9.2. Структурные элементы автомобильной системы на базе Android
 - 1.9.3. Связь между устройствами
- 1.10. Android в домашней автоматизации, носимых устройствах и интернете вещей (IoT)
 - 1.10.1. Подключенный мир
 - 1.10.2. Структурные элементы в системе домашней автоматизации Android
 - 1.10.3. Элементы Android *wearable-устройства*
 - 1.10.4. Android в интернете вещей (IoT)

Модуль 2. Фреймворки, используемые при разработке приложений для Android

- 2.1. *Фреймворки*, при разработке приложений для Android
 - 2.1.1. *Фреймворки*, используемые при разработке приложений для Android
 - 2.1.2. *Фреймворки*. Типологии
 - 2.1.3. Выбор *фреймворка* для проекта
- 2.2. Внедрение *фреймворков* в Android
 - 2.2.1. Android Core *Frameworks* для Java/Kotlin
 - 2.2.2. Jetpack *Compose*
 - 2.2.3. *Фреймворки* на других языках
- 2.3. Разрабатываемые системы управления библиотеками
 - 2.3.1. *Graddle*
 - 2.3.2. Автоматизация с помощью *Graddle*
 - 2.3.3. Инструмент разработки Maven

- 2.4. Чистый код
 - 2.4.1. Упорядоченный код
 - 2.4.2. Подготовка кода в Android-приложениях
 - 2.4.3. Закон тривиальности и расстановка приоритетов
- 2.5. Паттерны разработки под Android
 - 2.5.1. Категории паттернов
 - 2.5.2. Различия между паттернами
 - 2.5.3. Фабричный метод, наблюдатель и одиночка
- 2.6. MVP Модель, представление и ведущий
 - 2.6.1. MVC. Модель, представление и управляющий
 - 2.6.2. Модель, представление и ведущий
 - 2.6.3. Наглядный пример: Битва покемонов
- 2.7. MVVM. Модель, представление и модель представления
 - 2.7.1. MVC vs. MVVM
 - 2.7.2. Модель, представление и модель представления
 - 2.7.3. Практический пример: Битва покемонов II
- 2.8. Фреймворки и библиотеки, наиболее используемые в Android
 - 2.8.1. Библиотеки взаимодействия с API
 - 2.8.2. Библиотеки преобразования данных
 - 2.8.3. Firebase и Firebase Analytics
- 2.9. Визуальный фреймворк для Android
 - 2.9.1. Жизненный цикл Android приложения
 - 2.9.2. Проектирование XML-видов
 - 2.9.3. Расположение элементов и анимации в XML
- 2.10. Фреймворки Android в на других языках
 - 2.10.1. React Native
 - 2.10.2. Трепетание
 - 2.10.3. Ionic

Модуль 3. Интерфейсы и компоновка в процессе разработки приложений для Android

- 3.1. Жизненный цикл в Android
 - 3.1.1. Жизненный цикл Android
 - 3.1.2. Взаимосвязь процессов и действий
 - 3.1.3. Сохранение состояния приложения
 - 3.1.4. Чистая архитектура в Android
- 3.2. Представление в разработке приложений для Android (Views)
 - 3.2.1. Презентационный слой чистой архитектуры
 - 3.2.2. Представление Recycler
 - 3.2.3. Представление адаптера
- 3.3. Дизайны в разработке (компоновки) приложений для Android
 - 3.3.1. Компоновка для Android
 - 3.3.2. Constraint Layout
 - 3.3.3. Создание компоновки с помощью Android Studio Layout Editor
- 3.4. Анимация в разработке Android-приложений (Animations)
 - 3.4.1. Иконки и изображения
 - 3.4.2. Переходы
 - 3.4.3. Разница между анимацией свойств и анимацией представлений
- 3.5. Действия и намерения в разработке Android-приложений (Activity and Intentions)
 - 3.5.1. Явные и неявные намерения
 - 3.5.2. План действий
 - 3.5.3. Связь между действиями
- 3.6. Альтернативные и системные ресурсы (Material Design, Cardboard и т. д.)
 - 3.6.1. Material Design для Android
 - 3.6.2. Мультимедиа на Android
 - 3.6.3. Виртуальная реальность с Google Cardboard для Android NDK
- 3.7. Стили и темы в разработке приложений для Android
 - 3.7.1. Стили в проекте Android
 - 3.7.2. Темы проекта Android
 - 3.7.3. Повторное использование стилей и тем

- 3.8. Графика, сенсорный экран и датчики
 - 3.8.1. Работа с расширенной графикой
 - 3.8.2. Работа с сенсорным экраном и клавиатурой
 - 3.8.3. Использование датчиков устройств в Android
- 3.9. Дизайн для дополненной реальности
 - 3.9.1. Сложные интерфейсы с использованием камеры
 - 3.9.2. Датчики движения и GPS в дополненной реальности
 - 3.9.3. Воспроизведение на нестандартных дисплеях
 - 3.9.4. Распространенные ошибки и проблемы
- 3.10. Расширенная настройка пользовательского интерфейса с помощью AndroidManifest.xml
 - 3.10.1. Возможности файла манифеста Android
 - 3.10.2. Программный дизайн против декларативного
 - 3.10.3. Ключевые компоненты файла манифеста

Модуль 4. Язык программирования приложений для Android. Kotlin

- 4.1. Язык программирования Kotlin
 - 4.1.1. Язык программирования Kotlin
 - 4.1.2. Язык программирования Kotlin. Характеристики
 - 4.1.3. Как работает программа на Kotlin
- 4.2. Программирование на Kotlin
 - 4.2.1. Структура программы на Kotlin
 - 4.2.2. Зарезервированные слова и синтаксис
 - 4.2.3. Написание консоли и чтение пользовательских сообщений - Hello World
- 4.3. Типы и переменные в Kotlin
 - 4.3.1. Типы и переменные в Kotlin
 - 4.3.2. Объяснение переменных: Var vs. Val
 - 4.3.3. Операторы
 - 4.3.4. Преобразования типов
 - 4.3.5. Arrays
- 4.4. Управление потоком в Kotlin
 - 4.4.1. Управление потоком
 - 4.4.2. Условные выражения
 - 4.4.3. Петли





- 4.5. Функции Kotlin
 - 4.5.1. Функции Kotlin
 - 4.5.2. Структура функции
 - 4.5.3. Функции области видимости (*Scope Functions*)
- 4.6. Классы и объекты в Kotlin
 - 4.6.1. Классы и объекты в Kotlin
 - 4.6.2. Классы
 - 4.6.3. Предметы
 - 4.6.4. Конструкторы и инициализация свойств
 - 4.6.5. Внутренние классы и вложенные классы
 - 4.6.6. Классы данных
- 4.7. Наследственность в Kotlin
 - 4.7.1. Наследственность
 - 4.7.2. Суперклассы и подклассы
 - 4.7.3. Перезапись свойств и функций
 - 4.7.4. Наследственность vs. Другие типы отношений между классами
 - 4.7.5. Герметичные классы
 - 4.7.6. Перечисляемые
- 4.8. Абстрактные классы и интерфейсы в Kotlin
 - 4.8.1. Абстрактные классы и интерфейсы
 - 4.8.2. Абстрактные классы
 - 4.8.3. Интерфейсы
 - 4.8.4. Проверка и преобразование типов – операторы `Is`, `When`, `As`
- 4.9. Коллекции в Kotlin
 - 4.9.1. Коллекции в Kotlin
 - 4.9.2. Список
 - 4.9.3. Группа
 - 4.9.4. Карта
- 4.10. Обработка исключений и нулевых значений в Kotlin
 - 4.10.1. Обработка исключений и нулевых значений
 - 4.10.2. Нулевое значение, типы *Nullable* и *Non-Nullable*
 - 4.10.3. Исключения

Модуль 5. Язык программирования приложений для Android. Расширенные возможности Kotlin в области универсальности, функционального программирования и параллелизма

- 5.1. Общность в Kotlin
 - 5.1.1. Общность в Kotlin
 - 5.1.2. Общность в коллекциях, функциях, классах и интерфейсах
 - 5.1.3. Ковариация и контрвариантность: *Вывод или ввод*
- 5.2. Функции *Lambda* в Kotlin
 - 5.2.1. Функции *Lambdas*
 - 5.2.2. Структура функции *Lambda*
 - 5.2.3. Использование функций *Lambda*
- 5.3. Функции высшего порядка в Kotlin
 - 5.3.1. Функции высшего порядка
 - 5.3.2. Функции высшего порядка стандарта Kotlin
 - 5.3.3. Связывающие вызовы функций
- 5.4. Расширения Kotlin
 - 5.4.1. Расширения Kotlin
 - 5.4.2. Функции расширения
 - 5.4.3. Свойства расширений
 - 5.4.4. Объекты-компаньоны
- 5.5. Паттерн *делегирования* в Kotlin
 - 5.5.1. Паттерн *делегирования*
 - 5.5.2. *Делегирование* в Kotlin
 - 5.5.3. Делегированные свойства
- 5.6. Аннотации и отражение в Kotlin
 - 5.6.1. Аннотации и отражение
 - 5.6.2. Аннотации в Kotlin
 - 5.6.3. Отражение в Kotlin
- 5.7. *Тестирование* Kotlin
 - 5.7.1. *Тестирование* Kotlin
 - 5.7.2. *Фреймворки* и библиотеки для *тестирования* Kotlin
 - 5.7.3. *Kotest*

- 5.8. Асинхронное программирование в Kotlin
 - 5.8.1. Асинхронное программирование
 - 5.8.2. Техники асинхронного программирования в Kotlin
 - 5.8.3. Сравнение техник программирования
- 5.9. Корутины в Kotlin
 - 5.9.1. Корутины
 - 5.9.2. Каналы
 - 5.9.3. Контекст и *диспетчеры*
 - 5.9.4. Общее состояние и параллелизм
 - 5.9.5. Обработка исключений в корутинах
- 5.10. Экосистема Kotlin
 - 5.10.1. Экосистема Kotlin
 - 5.10.2. Библиотеки Kotlin
 - 5.10.3. Инструменты для Kotlin

Модуль 6. Управление данными на устройствах Android

- 6.1. Управление данными. Типология
 - 6.1.1. Управление данными в мобильных устройствах
 - 6.1.2. Альтернативы для управления данными на устройствах Android
 - 6.1.3. Генерация данных для работы искусственного интеллекта и аналитики использования
 - 6.1.4. Инструменты измерения производительности для оптимального управления данными
- 6.2. Управление предпочтениями пользователей
 - 6.2.1. Типы данных в файлах предпочтений
 - 6.2.2. Управление предпочтениями пользователей
 - 6.2.3. Экспорт предпочтений. Управление разрешениями
- 6.3. Система хранения файлов
 - 6.3.1. Классификация хранилищ файлов на мобильных устройствах
 - 6.3.2. Внутренняя файловая система
 - 6.3.3. Внешняя файловая система
- 6.4. JSON-файлы как хранилище данных в Android
 - 6.4.1. Неструктурированная информация в файлах JSON
 - 6.4.2. Библиотеки для управления данными в формате JSON
 - 6.4.3. Использование JSON на Android. Рекомендации и оптимизация

- 6.5. XML-файлы как хранилище данных в Android
 - 6.5.1. Формат XML в Android
 - 6.5.2. XML с помощью библиотек SAX
 - 6.5.3. XML с помощью библиотек DOM
- 6.6. Базы данных SQLite
 - 6.6.1. Реляционная база данных для управления данными
 - 6.6.2. Использование базы данных
 - 6.6.3. Методы SQLite для управления данными
- 6.7. Расширенное использование баз данных SQLite
 - 6.7.1. Восстановление после сбоев с помощью транзакций SQLite
 - 6.7.2. Использование кэширования для ускорения доступа к данным
 - 6.7.3. Базы данных на мобильных устройствах
- 6.8. Библиотека Room
 - 6.8.1. Архитектура библиотеки Room
 - 6.8.2. Библиотека Room. Функциональность
 - 6.8.3. Библиотека Room: преимущества и недостатки
- 6.9. *Контент-провайдер* для обмена информацией
 - 6.9.1. *Контент-провайдер* для обмена информацией
 - 6.9.2. *Контент-провайдер* на Android Техника использования
 - 6.9.3. Безопасность в *контент-провайдере*
- 6.10. Сбор данных в интернет-облаке
 - 6.10.1. Android и облачные системы хранения данных (Cloud)
 - 6.10.2. SOAP и REST сервисы для Android
 - 6.10.3. Проблемы распределительных систем
 - 6.10.4. Интернет как резервная копия данных приложения

Модуль 7. Инструменты для Android-устройств

- 7.1. Управление: Инструменты TO DO
 - 7.1.1. Инструменты рынка
 - 7.1.2. Инструменты рынка Сравнение функциональных возможностей
 - 7.1.3. Инструменты управления. Различия
- 7.2. MDM: управление корпоративными устройствами
 - 7.2.1. Контроль корпоративных устройств
 - 7.2.2. Анализ основных инструментов рынка
 - 7.2.3. Выбор инструмента
- 7.3. CRM: инструменты рынка
 - 7.3.1. Анализ рыночных инструментов с помощью приложения Android
 - 7.3.2. Инструменты рынка. Эффективность
 - 7.3.3. Инструменты рынка. Использование
- 7.4. Дроны на базе Android
 - 7.4.1. Приложения на Android-устройствах для управления дронами
 - 7.4.2. Автономные устройства управления
 - 7.4.3. Использование дронов на Android
- 7.5. Преимущества Android в банковских платформах
 - 7.5.1. Android в банковских платформах
 - 7.5.2. Риски и мошенничество киберпреступников
 - 7.5.3. Использование мобильных устройств
- 7.6. *Брокерская деятельность* на мобильных устройствах
 - 7.6.1. Инструменты рынка и их использование
 - 7.6.2. Сравнение инструментов
 - 7.6.3. Выбор инструмента для каждого вида использования
- 7.7. Инструменты для развлечения и обучения
 - 7.7.1. Использование
 - 7.7.2. Инструменты рынка
 - 7.7.3. Сравнение функциональных возможностей инструментов для разработки на Android

- 7.8. Android Интернет вещей IoT
 - 7.8.1. *Фреймворк и рыночные платформы*
 - 7.8.2. *Риски и соображения, связанные с Интернетом вещей на Android*
 - 7.8.3. *Использование IoT на Android*
- 7.9. Эффективность процессов
 - 7.9.1. *Анализ рынка инструментов для создания приложений*
 - 7.9.2. *Сравнение инструментов для создания Android-приложений*
 - 7.9.3. *Пример использования*
- 7.10. Самые скачиваемые приложения на сегодняшний день
 - 7.10.1. *Самые скачиваемые инструменты на сегодняшний день*
 - 7.10.2. *Группировка по типам программ*
 - 7.10.3. *Основные, второстепенные и сравнительные характеристики с iOS*

Модуль 8. Адаптивный дизайн на Android

- 8.1. *Адаптивный дизайн*
 - 8.1.1. *Адаптивный дизайн*
 - 8.1.2. *Удобство использования, доступность и UX*
 - 8.1.3. *Адаптивный дизайн. Преимущества и недостатки*
- 8.2. *Мобильные устройства vs. Планшет vs. Веб-сайт vs. Смарт-часы*
 - 8.2.1. *Разные форматы, разные размеры, разные потребности*
 - 8.2.2. *Дизайнерские проблемы*
 - 8.2.3. *Адаптивный vs. Отзывчивый*
- 8.3. *Руководства по стилю*
 - 8.3.1. *Руководства по стилю. Полезность*
 - 8.3.2. *Дизайн материалов*
 - 8.3.3. *Руководство по собственному стилю*
- 8.4. *Гибкая компоновка*
 - 8.4.1. *Гибкая компоновка*
 - 8.4.2. *Базовая компоновка*
 - 8.4.3. *Компоновка в Grid*
 - 8.4.4. *Компоновка с использованием Relative Layout*
 - 8.4.5. *Компоновка с использованием Constraint Layout*

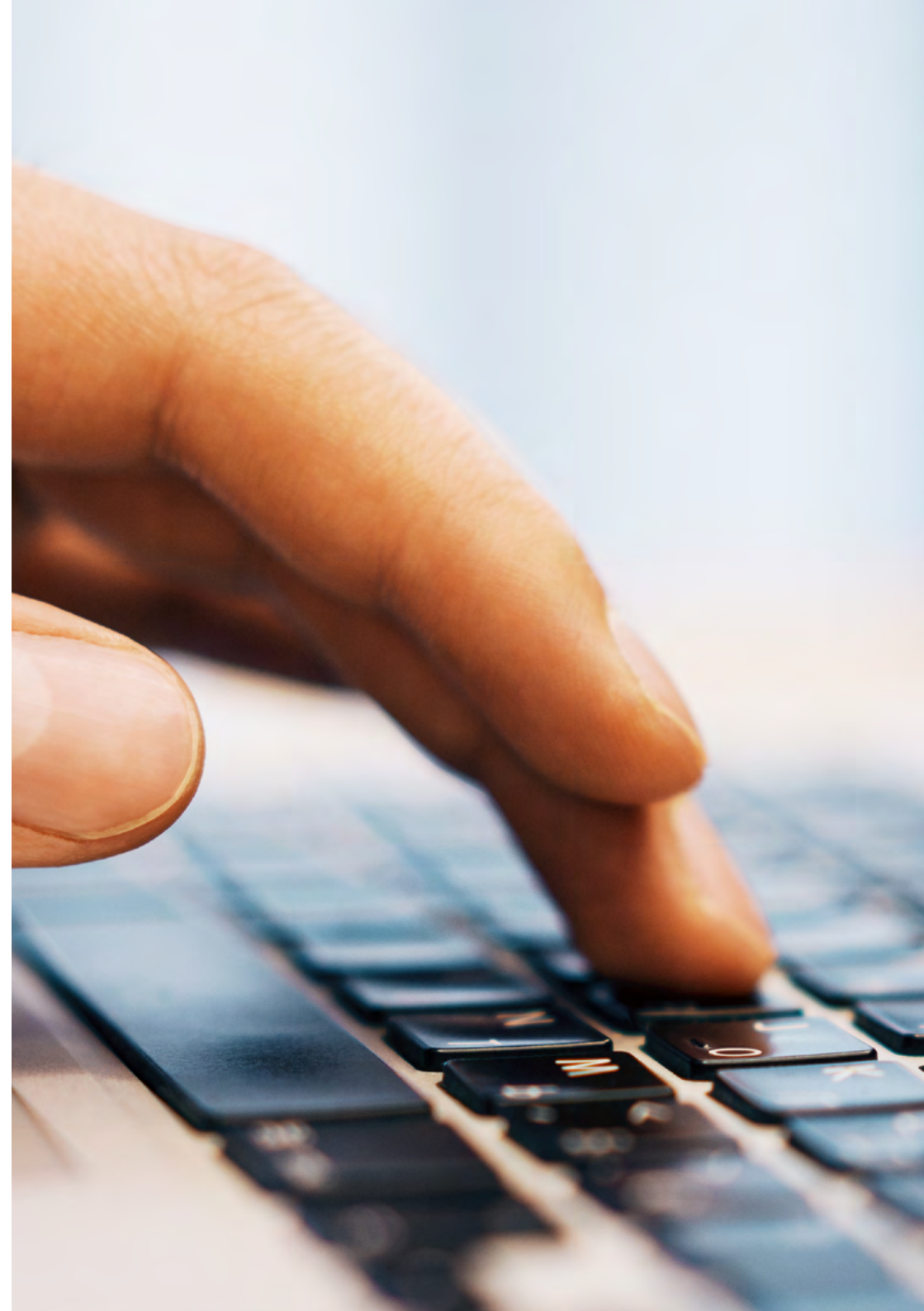




- 8.5. Гибкие ресурсы
 - 8.5.1. Гибкие ресурсы
 - 8.5.2. Изображения
 - 8.5.3. 9-Patch
 - 8.5.4. Глобальные ресурсы
- 8.6. Гибкая навигация
 - 8.6.1. Гибкая навигация
 - 8.6.2. Навигация с помощью активностей
 - 8.6.3. Навигация с помощью фрагментов
- 8.7. Внешние инструменты
 - 8.7.1. Автоматические генераторы
 - 8.7.2. Инструменты для создания прототипов
 - 8.7.3. Инструменты дизайна
- 8.8. Отладка и тесты
 - 8.8.1. *Отладочные макеты*
 - 8.8.2. Автоматические тесты
 - 8.8.3. Разработка на основе компонентов
 - 8.8.4. Хорошие практики тестирования и проб
- 8.9. Альтернативы нативному Android I. Веб-страницы
 - 8.9.1. Дизайн в WebView
 - 8.9.2. Пользовательские вкладки Chrome
 - 8.9.3. Отладка и тесты на веб-страницах
- 8.10. Альтернативы нативной версии Android II. Гибридные приложения
 - 8.10.1. *React native*
 - 8.10.2. *Flutter*
 - 8.10.3. *Ionic*
 - 8.10.4. *Cordova*

Модуль 9. Маркетинг в приложениях для Android

- 9.1. От обслуживания клиентов к опыту клиентов
 - 9.1.1. Обслуживание клиентов. Развитие современного клиента
 - 9.1.2. Пользователь с доступом к информации. Запросы и потребности
 - 9.1.3. Обратная связь как источник знаний
- 9.2. Клиентский опыт (Customer Journey)
 - 9.2.1. Путь пользователя к конверсии
 - 9.2.2. Микросегментация
 - 9.2.3. Опыт использования кросс-каналов
- 9.3. Измерение пользовательского опыта
 - 9.3.1. Веб-архитектура и мобильные устройства
 - 9.3.2. Сессионная аналитика как новый стандарт
 - 9.3.3. Состояние дел в области пользовательского опыта
- 9.4. Маркетинг Android-приложений
 - 9.4.1. Клиентский опыт+ИИ
 - 9.4.2. Клиентский опыт+блокчейн
 - 9.4.3. Клиентский опыт+IoT
- 9.5. Продукты CX (клиентский опыт)
 - 9.5.1. Отраслевые стандарты
 - 9.5.2. Телеприсутствие
 - 9.5.3. Клиентский опыт для всех участников разработки
- 9.6. Работа, ориентированная на пользователя
 - 9.6.1. Оборудование
 - 9.6.2. Дизайнерское мышление
 - 9.6.3. Полевые работы
- 9.7. Наука о пользователе
 - 9.7.1. Наука о пользователе. Золотое правило
 - 9.7.2. Повторение
 - 9.7.3. Распространенные ошибки
- 9.8. Прототипирование и вёрфрейминг
 - 9.8.1. Прототипирование и вёрфрейминг
 - 9.8.2. Практическая работа
 - 9.8.3. Продвинутый уровень



- 9.9. Мобильные интерфейсы
 - 9.9.1. Визуальный дизайн. Правила
 - 9.9.2. Интерфейс приложений. Ключевые моменты
 - 9.9.3. Лучшие практики разработки мобильных интерфейсов
- 9.10. Лучшие практики в области пользовательского опыта. Советы разработчикам
 - 9.10.1. Первый уровень. Лучшие практики в CX
 - 9.10.2. Второй уровень. Лучшие практики в UX
 - 9.10.3. Третий уровень. Лучшие практики пользовательского интерфейса

Модуль 10. Цикл жизни приложения для Android. Облако, Playstore и версионирование

- 10.1. Жизненный цикл программного обеспечения
 - 10.1.1. Жизненный цикл программного обеспечения
 - 10.1.2. Методологии Agile
 - 10.1.3. Непрерывный гибкий цикл программного обеспечения
- 10.2. Разработка продукта вручную
 - 10.2.1. Интеграция вручную
 - 10.2.2. Доставка вручную
 - 10.2.3. Развертывание вручную
- 10.3. Интеграция под наблюдением
 - 10.3.1. Непрерывная интеграция
 - 10.3.2. Ручной обзор
 - 10.3.3. Статические автоматические обзоры
- 10.4. Логическое тестирование
 - 10.4.1. Модульные тесты
 - 10.4.2. Интеграционное тестирование
 - 10.4.3. Поведенческое тестирование
- 10.5. Непрерывная интеграция
 - 10.5.1. Цикл непрерывной интеграции
 - 10.5.2. Зависимости между интеграциями
 - 10.5.3. Непрерывная интеграция как методология управления репозиториями

- 10.6. Непрерывная доставка
 - 10.6.1. Непрерывная доставка. Типология проблем, требующих решения
 - 10.6.2. Непрерывная доставка. Устранение неполадок
 - 10.6.3. Преимущества непрерывной доставки
- 10.7. Непрерывное развертывание
 - 10.7.1. Непрерывное развертывание. Типология проблем, требующих решения
 - 10.7.2. Непрерывное развертывание. Устранение неполадок
- 10.8. *Firebase Test Lab*
 - 10.8.1. Настройка из *GCloud*
 - 10.8.2. Настройка Jenkins
 - 10.8.3. Использование Jenkins. Преимущества
- 10.9. Настройка *Gradle*
 - 10.9.1. Система автоматизирования *Gradle*
 - 10.9.2. Компонент *Gradle Build Flavors*
 - 10.9.3. Компонент *Gradle Lint*
- 10.10. Жизненный цикл приложения для Android. Пример
 - 10.10.1. Настройка *SemaphoreCI* и GitHub
 - 10.10.2. Настройка рабочих блоков
 - 10.10.3. Продвижение и *Deployment*



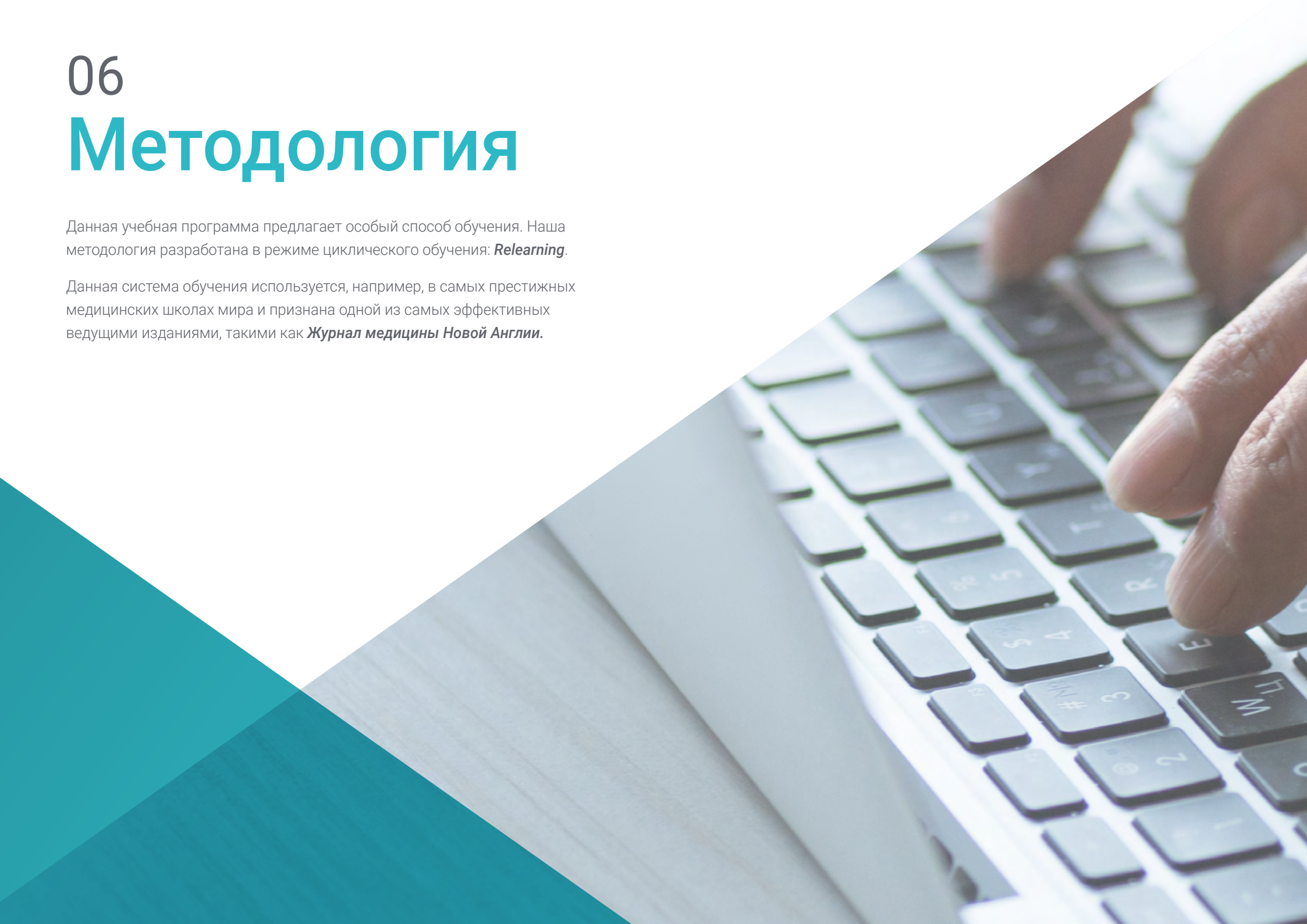
Сделайте этот шаг вперед,
чтобы быть в курсе самых
последних новостей,
связанных с разработкой
приложений на Android”

06

Методология

Данная учебная программа предлагает особый способ обучения. Наша методология разработана в режиме циклического обучения: **Relearning**.

Данная система обучения используется, например, в самых престижных медицинских школах мира и признана одной из самых эффективных ведущими изданиями, такими как **Журнал медицины Новой Англии**.



“

Откройте для себя методику *Relearning*, которая отвергает традиционное линейное обучение, чтобы показать вам циклические системы обучения: способ, который доказал свою огромную эффективность, особенно в предметах, требующих запоминания”

Исследование кейсов для контекстуализации всего содержания

Наша программа предлагает революционный метод развития навыков и знаний. Наша цель - укрепить компетенции в условиях меняющейся среды, конкуренции и высоких требований.

“

С TECH вы сможете познакомиться со способом обучения, который опровергает основы традиционных методов образования в университетах по всему миру”



Вы получите доступ к системе обучения, основанной на повторении, с естественным и прогрессивным обучением по всему учебному плану.



В ходе совместной деятельности и рассмотрения реальных кейсов студент научится разрешать сложные ситуации в реальной бизнес-среде.

Инновационный и отличный от других метод обучения

Эта программа TECH - интенсивная программа обучения, созданная с нуля, которая предлагает самые сложные задачи и решения в этой области на международном уровне. Благодаря этой методологии ускоряется личностный и профессиональный рост, делая решающий шаг на пути к успеху. Метод кейсов, составляющий основу данного содержания, обеспечивает следование самым современным экономическим, социальным и профессиональным реалиям.

“*Наша программа готовит вас к решению новых задач в условиях неопределенности и достижению успеха в карьере*”

Кейс-метод является наиболее широко используемой системой обучения лучшими преподавателями в мире. Разработанный в 1912 году для того, чтобы студенты-юристы могли изучать право не только на основе теоретического содержания, метод кейсов заключается в том, что им представляются реальные сложные ситуации для принятия обоснованных решений и ценностных суждений о том, как их разрешить. В 1924 году он был установлен в качестве стандартного метода обучения в Гарвардском университете.

Что должен делать профессионал в определенной ситуации? Именно с этим вопросом мы сталкиваемся при использовании кейс-метода - метода обучения, ориентированного на действие. На протяжении всей курса студенты будут сталкиваться с многочисленными реальными случаями из жизни. Им придется интегрировать все свои знания, исследовать, аргументировать и защищать свои идеи и решения.

Методология *Relearning*

ТЕСН эффективно объединяет метод кейсов с системой 100% онлайн-обучения, основанной на повторении, которая сочетает различные дидактические элементы в каждом уроке.

Мы улучшаем метод кейсов с помощью лучшего метода 100% онлайн-обучения: *Relearning*.

В 2019 году мы достигли лучших результатов обучения среди всех онлайн-университетов в мире.

В ТЕСН вы будете учиться по передовой методике, разработанной для подготовки руководителей будущего. Этот метод, играющий ведущую роль в мировой педагогике, называется *Relearning*.

Наш университет - единственный вуз, имеющий лицензию на использование этого успешного метода. В 2019 году нам удалось повысить общий уровень удовлетворенности наших студентов (качество преподавания, качество материалов, структура курса, цели...) по отношению к показателям лучшего онлайн-университета.



В нашей программе обучение не является линейным процессом, а происходит по спирали (мы учимся, разучиваемся, забываем и заново учимся). Поэтому мы дополняем каждый из этих элементов по концентрическому принципу. Благодаря этой методике более 650 000 выпускников университетов добились беспрецедентного успеха в таких разных областях, как биохимия, генетика, хирургия, международное право, управленческие навыки, спортивная наука, философия, право, инженерное дело, журналистика, история, финансовые рынки и инструменты. Наша методология преподавания разработана в среде с высокими требованиями к уровню подготовки, с университетским контингентом студентов с высоким социально-экономическим уровнем и средним возрастом 43,5 года.

Методика Relearning позволит вам учиться с меньшими усилиями и большей эффективностью, все больше вовлекая вас в процесс обучения, развивая критическое мышление, отстаивая аргументы и противопоставляя мнения, что непосредственно приведет к успеху.

Согласно последним научным данным в области нейронауки, мы не только знаем, как организовать информацию, идеи, образы и воспоминания, но и знаем, что место и контекст, в котором мы что-то узнали, имеют фундаментальное значение для нашей способности запомнить это и сохранить в гиппокампе, чтобы удержать в долгосрочной памяти.

Таким образом, в рамках так называемого нейрокогнитивного контекстно-зависимого электронного обучения, различные элементы нашей программы связаны с контекстом, в котором участник развивает свою профессиональную практику.



В рамках этой программы вы получаете доступ к лучшим учебным материалам, подготовленным специально для вас:



Учебный материал

Все дидактические материалы создаются преподавателями специально для студентов этого курса, чтобы они были действительно четко сформулированными и полезными.

Затем вся информация переводится в аудиовизуальный формат, создавая дистанционный рабочий метод TECH. Все это осуществляется с применением новейших технологий, обеспечивающих высокое качество каждого из представленных материалов.



Мастер-классы

Существуют научные данные о пользе экспертного наблюдения третьей стороны.

Так называемый метод обучения у эксперта укрепляет знания и память, а также формирует уверенность в наших будущих сложных решениях.



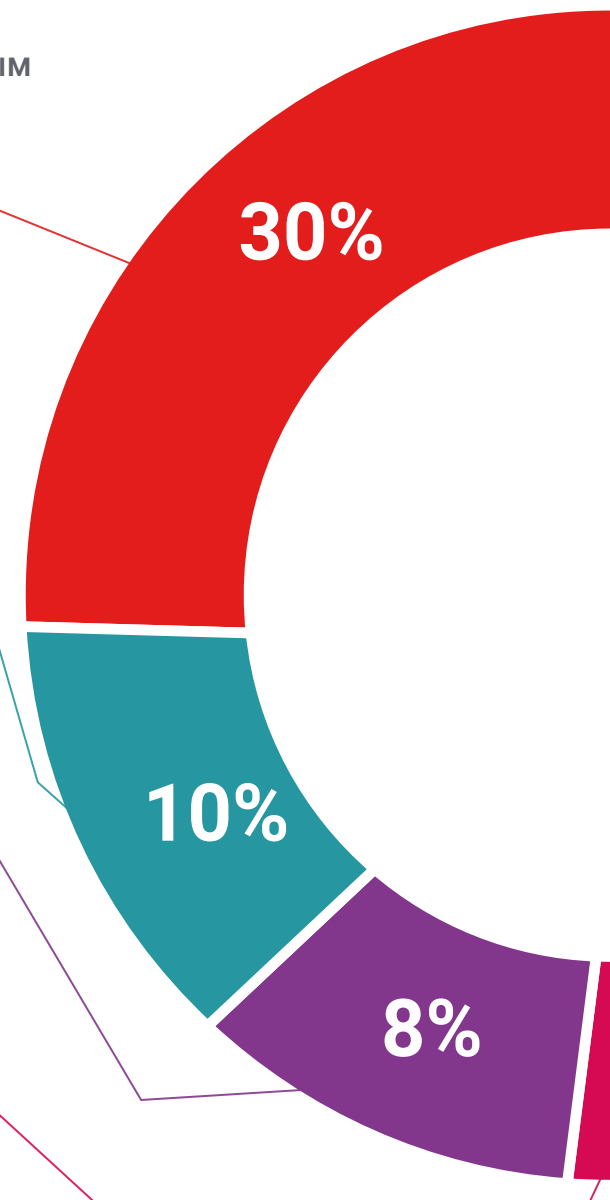
Практика навыков и компетенций

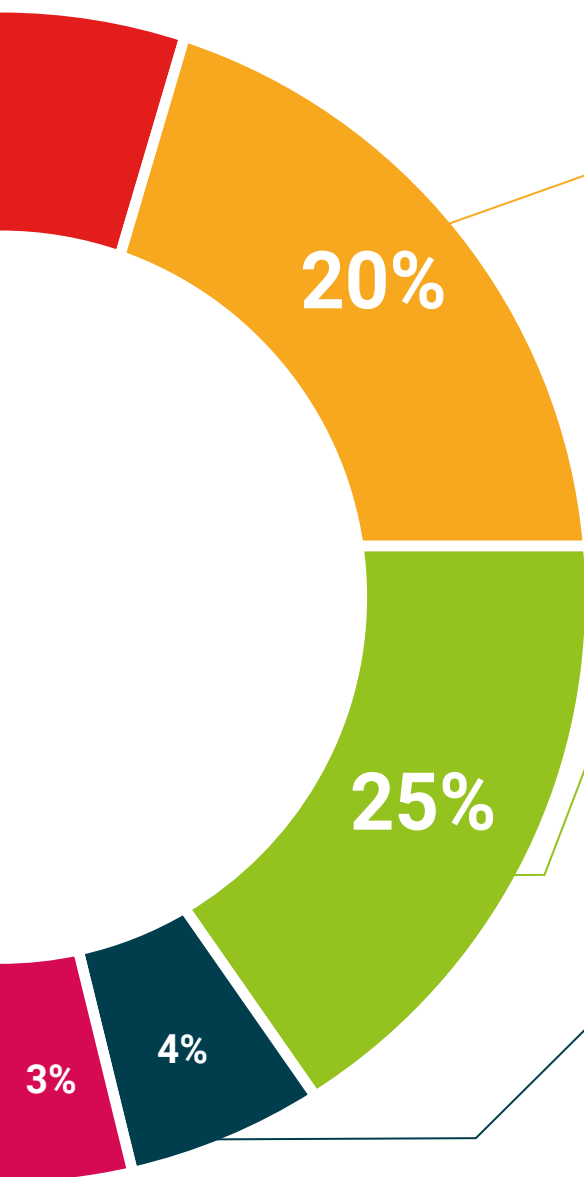
Студенты будут осуществлять деятельность по развитию конкретных компетенций и навыков в каждой предметной области. Практика и динамика приобретения и развития навыков и способностей, необходимых специалисту в рамках глобализации, в которой мы живем.



Дополнительная литература

Новейшие статьи, консенсусные документы и международные руководства включены в список литературы курса. В виртуальной библиотеке TECH студент будет иметь доступ ко всем материалам, необходимым для завершения обучения.





Метод кейсов

Метод дополнится подборкой лучших кейсов, выбранных специально для этой квалификации. Кейсы представляются, анализируются и преподаются лучшими специалистами на международной арене.



Интерактивные конспекты

Мы представляем содержание в привлекательной и динамичной мультимедийной форме, которая включает аудио, видео, изображения, диаграммы и концептуальные карты для закрепления знаний. Эта уникальная обучающая система для представления мультимедийного содержания была отмечена компанией Microsoft как "Европейская история успеха".



Тестирование и повторное тестирование

На протяжении всей программы мы периодически оцениваем и переоцениваем ваши знания с помощью оценочных и самооценочных упражнений: так вы сможете убедиться, что достигаете поставленных целей.



07

Квалификация

Специализированная магистратура в области разработки приложений для Android гарантирует, помимо самого строгого и современного обучения, получение диплома об окончании Специализированной магистратуры, выдаваемого ТЕСН Технологическим университетом.



“

Успешно пройдите эту программу
и получите университетский
диплом без хлопот, связанных с
поездками и бумажной волокитой”

Данная **Специализированная магистратура в области разработки приложений для Android** содержит самую полную и современную программу на рынке.

После прохождения аттестации студент получит по почте* с подтверждением получения соответствующий диплом **Специализированной магистратуры**, выданный **ТЕСН Технологическим университетом**.

Диплом, выданный **ТЕСН Технологическим университетом**, подтверждает квалификацию, полученную в Специализированной магистратуре, и соответствует требованиям, обычно предъявляемым биржами труда, конкурсными экзаменами и комитетами по оценке карьеры.

Диплом: **Специализированная магистратура в области разработки приложений для Android**

Формат: **онлайн**

Продолжительность: **12 месяцев**



*Гаагский апостиль. В случае, если студент потребует, чтобы на его диплом в бумажном формате был проставлен Гаагский апостиль, TECH EDUCATION предпримет необходимые шаги для его получения за дополнительную плату.

Будущее

Здоровье Доверие Люди

Образование Информация Тьюторы

Гарантия Аккредитация Преподавание

Институты Технология Обучение

Сообщество Обязательство

Персональное внимание Инновации

Знания Настоящее Качество

Веб обучение

Развитие Институты

Виртуальный класс Языки

tech технологический
университет

Специализированная
магистратура

Разработка приложений
для Android

- » Формат: онлайн
- » Продолжительность: 12 месяцев
- » Учебное заведение: ТЕСН Технологический университет
- » Расписание: по своему усмотрению
- » Экзамены: онлайн

Специализированная магистратура

Разработка приложений
для Android

```
//up/down
if (e.keyCode == 38 || e.keyCode == 40) {
    e.preventDefault();
    e.stopImmediatePropagation();
    if (e.keyCode == 38) { // up
        // show previous search query
        if (hist.currentIndex == hist.history.length) {
            hist.currentQuery = input.value;
            // skip previous search if we're already showing the same
            if (hist.currentQuery == hist.history[hist.currentIndex - 1])
                hist.currentIndex--;
        }
    }
}
```