**КУРСОВАЯ РАБОТА**

**По дисциплине:** /МДК.01.03 Разработка мобильных приложений

**Специальность:** 09.02.07 Информационные системы и программирование

**Тема:** «Разработка мобильного приложения «Нестандартный конвертер чисел»

**Группа:** ИСП311, **Курс:** 3

**Выполнил студент** \_\_\_\_\_\_\_\_\_\_ /Васильев Г.А/

подпись ФИО

**Руководитель** \_\_\_\_\_\_\_\_\_\_\_/Коротков Д.В/

подпись ФИО

**Оценка** \_\_\_\_\_\_\_\_\_\_

**Дата** \_\_\_\_\_\_\_\_\_\_

Москва, 2023

ОГЛАВЛЕНИЕ

[ОГЛАВЛЕНИЕ 2](#_Toc134648607)

[ВВЕДЕНИЕ 3](#_Toc134648608)

[1.ТЕОРЕТИЧЕСКАЯ ЧАСТЬ 5](#_Toc134648609)

[1.1 Определение мобильного приложения 5](#_Toc134648610)

[1.2 Анализ рынка программных продуктов 5](#_Toc134648611)

[1.3 Выбор платформы и средств разработки 10](#_Toc134648612)

[1.3.1 Операционная система iOS 11](#_Toc134648613)

[1.3.2 Операционная система Android 11](#_Toc134648614)

[1.3.3 Вывод 13](#_Toc134648615)

[1.4 Выбор средств разработки для ОС Android 14](#_Toc134648616)

[1.4.1 Среда разработки Eclipse 15](#_Toc134648617)

[1.4.2 Среда разработки Android Studio 16](#_Toc134648618)

[1.4.3 KivyMD 17](#_Toc134648619)

[1.4.4 Xamarin.Forms 18](#_Toc134648620)

[1.4.5 Вывод 18](#_Toc134648621)

[2. ПРАКТИЧЕСКАЯ ЧАСТЬ 19](#_Toc134648622)

[2.1 Определение требований 19](#_Toc134648623)

[2.2 Проектирование пользовательского интерфейса 19](#_Toc134648624)

[2.3 Реализация приложения 21](#_Toc134648625)

[2.4 Тестирование приложения 24](#_Toc134648626)

[ЗАКЛЮЧЕНИЕ 25](#_Toc134648627)

[СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ 26](#_Toc134648628)

# **ВВЕДЕНИЕ**

Счисление – возможно, одного из самых древних изобретений человека, после огня и языка. Считать вещи древний человек научился тогда, когда начал осознавать свою индивидуальность. Жизнь современного человека абсолютно невозможно представить без вычислений и счета. Мы считаем, когда идем в магазин, когда оплачиваем жилищно-коммунальные услуги, когда работаем, когда приглашаем знакомых в гости. Считать вещи стало такой обыденностью, что многие вычислительные операции мы не фиксируем, не обращаем на них внимания. Но всегда ли так было?

Отнюдь. Древние люди придумали множество самых различных систем счисления, начиная от использования просто палочек и заканчивая сложными позиционными системами с использованием большого количества символов и правил. На самом деле, человеческой цивилизации понадобился не один век, чтобы прийти к арабской системе счисления, включающей в себя цифры от 0 до 9. Но и тут всё не так просто. Не все племена и народы были готовы перейти от системы, которую начали использовать еще их древние предки, к новой, непривычной и неблизкой им.

На данный момент существует специальное программное обеспечение, которое позволяет переводить числа из привычной, десятичной системы счисления в любую другую, общепринятую – двоичную, восьмеричную и шестнадцатеричную. Однако, очень сложно найти ПО, способное переводить число в нестандартные системы счисления, например, в систему древних майя или систему племени Аламбак.

Проектирование и разработка мобильного приложения по типу «Нестандартный конвертер чисел», которое, за счёт своей функциональности, позволяло бы проводить преобразования целых чисел в различные, как общепринятые, так и в экзотические системы счисления - цель данной курсовой работы. Для достижения поставленной цели необходимо:

1. Выбрать целевую платформу для создания мобильного приложения;
2. Изучить технологии разработки программного обеспечения для выбранной платформы;
3. Обозначить требования к целевой системе и к конечному продукту;
4. Разработать приложение, определить основные элементы интерфейса и основные механики.

В процессе выполнения курсовой работы студентом будут приобретены навыки разработки мобильных приложений, работы с инструментами разработки. Также, будут приобретены навыки в области анализа распределения пользователей мобильных платформ.

1.ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

# **1.1 Определение мобильного приложения**

Мобильное приложение – вид прикладного программного обеспечения, разработанное и предназначенное для использования на мобильных устройствах, к которым относятся смартфоны, планшеты, смарт-часы и т.д. Мобильные приложения предназначены для обеспечения пользователю доступ к информации «на ходу» - без привязки к стационарному компьютеру.

Существует великое множество мобильных приложений, которые, в свою очередь можно разделить на различные категории в зависимости от их функционала и предназначения. К популярным категориям мобильных приложений относятся: социальные сети, мессенджеры, мобильные игры, фитнес-трекеры, приложения для готовки, построения тренировочных планов, редакторы мультимедиа и прочие.

К категориям также можно отнести и различные инструменты, в том числе калькуляторы и конвертеры. Создание конвертера чисел – актуальный проект, который можно приспособить как для личного пользования, так и для обучающего процесса, так как данное приложение может содержать в себе наглядные примеры нестандартных численных операций.

# **1.2 Анализ рынка программных продуктов**

Для мобильных устройств существует большое количество программ для конвертации чисел, так как с развитием интернет-технологий спрос на подобное специфическое программное обеспечение растет.

Одно из таких приложений – «Конвертер систем счисления» (100 тысяч загрузок в магазине Google Play). Это свободно распространяемое приложение, которое поддерживает все основные функции конвертера систем счисления и имеет следующие отличительные особенности:

1. Небольшой объем приложения;
2. Высокая скорость работы;
3. Возможность работы не только с общепринятыми системами счисления, но и с прочими системами с базой от 3 до 15.

Графический интерфейс приложения представлен на рисунке 1.

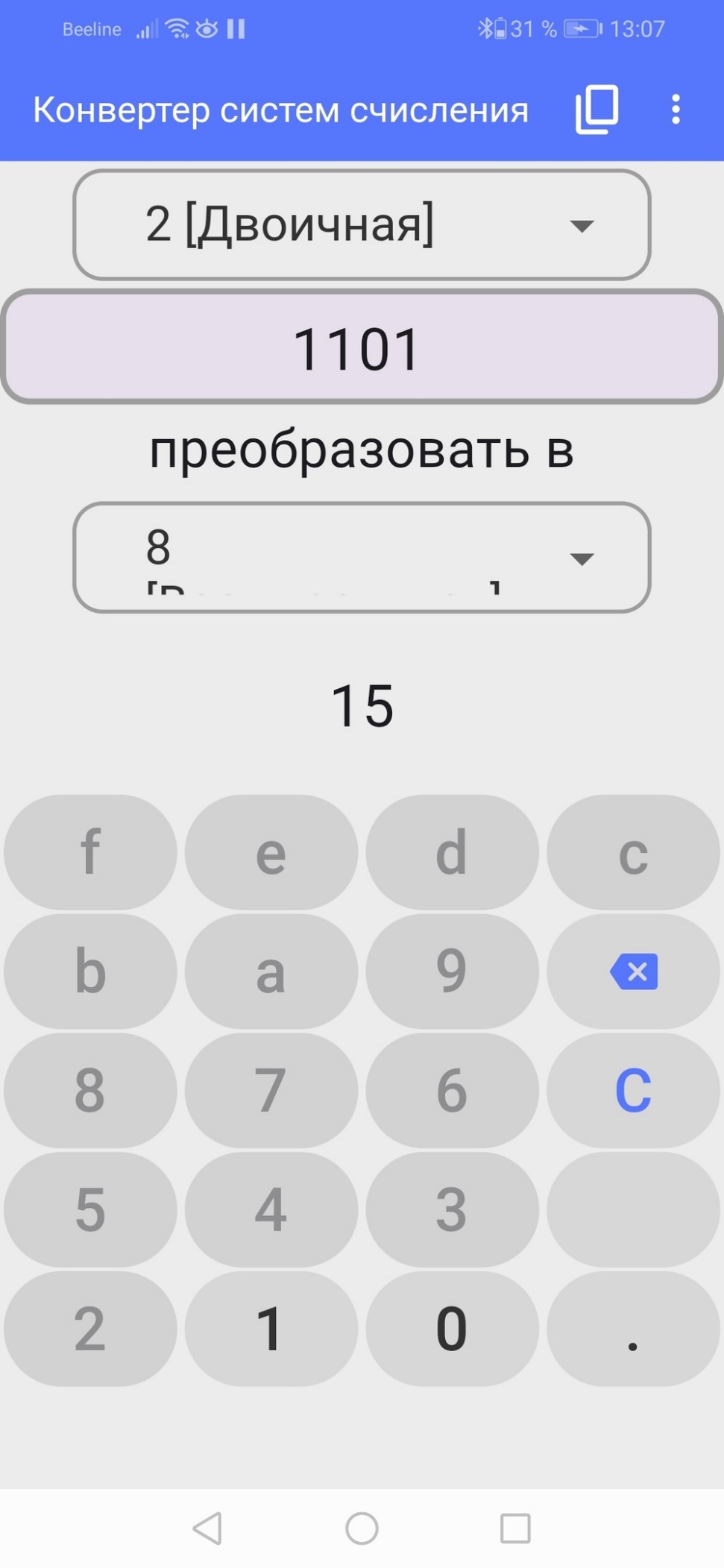


Рисунок 1 – интерфейс главной страницы приложения «Конвертер систем счисления»

Однако, у этого ПО есть также ряд недостатков, а именно: некорректно отображаемый интерфейс, не совсем некорректно работающее кнопки, нет возможности конвертировать числа в нестандартные вычислительные системы.

Все же, несмотря на все недочёты, это ПО имеет успех на рынке – его постоянно используют рядовые пользователи, что говорит о высокой актуальности подобного рода приложений.

Следующим к рассмотрению будет мобильное приложение Бинарный калькулятор (более 500 000 скачиваний). Это бесплатно распространяемое приложение, которое имеет следующие особенности:

1. Очень высокое быстродействие приложения;
2. Очень скромный объем приложения;
3. Высокий уровень точности вычислений;
4. Возможность производить элементарные арифметические операции (сложение, вычитание, умножение, деление) сразу в нескольких системах счисления.

Графический интерфейс данного приложения представлен на рисунке номер 2.

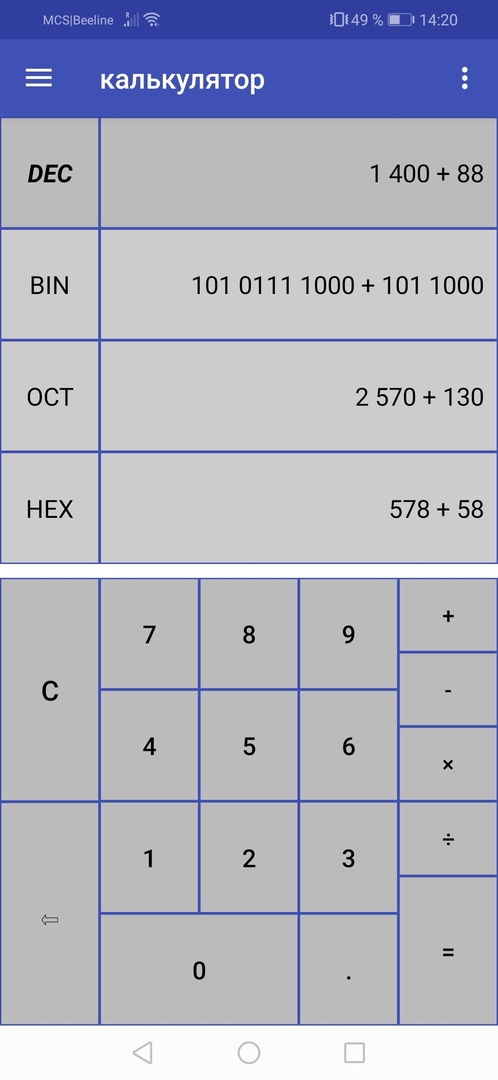


Рисунок 2 – интерфейс главной страницы приложения Бинарный калькулятор

Данное программное обеспечение также обладает рядом недостатков, среди которых:

1. Перегруженный, неинтуитивный интерфейс;
2. Неправильное наложение кнопок;
3. Монотонный интерфейс;
4. Нет возможности переводить числа в нестандартные системы

Тем не менее, данный программный продукт, несмотря на проблемы с оформлением пользовательского интерфейса весьма популярен среди пользователей мобильных устройств, о чем говорят положительные отзывы с площадки Google Play. Данный факт еще раз подтверждает актуальность и релевантность приложений подобного типа.

# **1.3 Выбор платформы и средств разработки**

После проведённого анализа рынка мобильных приложений следует определить, для какой именно операционной системы будет разрабатываться программное обеспечение и с использованием каких средств разработки.

На текущий момент времени существует великое многообразие операционных систем (ОС) для мобильных устройств, самыми известными из них являются iOS, Android и Windows Mobile.

Данные операционные системы, за счет конкуренции между собой, обладают множеством фундаментальных отличий, о которых необходимо знать, чтобы корректно выбрать платформу для разработки.

Помимо упомянутых операционных систем, имеется еще несколько представителей данной ниши, например, Lineage OS, позиционирующая себя как мобильная ОС с исходным кодом, Magic OS и Harmony, предназначенные в основном для китайских смартфонов и прочие. Но, учитывая низкую распространненость, заострять свое внимание на них мы не будем.

# **1.3.1 Операционная система iOS**

iOS – закрытая операционная система, разработанная корпорацией Apple специально для своих устройств в 2006-2007 годах. Предназначалась для использования на фирменных устройствах iPhone и iPod, но позже стала использоваться и на планшетах Apple iPad.

Ядро этой ОС основано на XNU, которое является старым ядром для настольных ПК Apple и обладает высоким уровнем защиты от вирусов и прочих мошеннических программ. Является почти полной копией ядра macOS и OS X. Ядро операционной предназначено на процессорах с архитектурой ARM.

iOS не позволяет распространять разработанное программное обеспечение без получения сертификатов безопасности и специальных разрешений – разработчикам необходимо проходить различные проверки и процедуры, чтобы получить разрешения на распространение своего продукта. После всех процедур безопасности, приложение отправляется на длительную процедуру предрелизной модерации. Для распространения приложения используется фирменный магазин приложений AppStore. Нельзя не отметить, что iOS – проприетарная система, её исходный код закрыт, а сама операционная система и, соответственно, приложения, разработанные для неё – могут использоваться только на фирменных устройствах Apple.

Для создания приложений используются языки Objective-C и Swift, однако, Objective-C считается морально устаревшим и поддержка данного языка полностью прекращена в современных моделях устройств Apple, в том числе мобильных. В качестве основной среды разработки может выступать XCode.

# **1.3.2 Операционная система Android**

Android – открытая операционная система, которая появилась в 2008 году, основана на ядре Linux. Отличительной особенностью данной ОС является её широкое применение: смартфоны, планшеты, электронные книги, цифровые проигрыватели, смарт-часы, телевизоры, проекторы и прочие устройства.

Данная платформа является самой распространенной ОС в мире. По данным Google на 2017 год в мире было выпущено более 2 миллиардов устройств, работающих на данной ОС. Это обусловлено всесторонним развитием системы корпорацией Google, а также открытым исходным кодом, который даёт возможность производителям электроники легко адаптировать и модифицировать систему под свои устройства. Каждый год для платформы выходят новые обновления и API, которые добавляют новый функционал в систему, а также дают более удобные инструменты для разработки, что существенно облегчает создание приложений. При этом стоит уделить особое внимание выбору минимально 8 поддерживаемой версии API поскольку, чем младше версия, тем больше устройств будет способно запустить наше приложение.

Преимуществом данной ОС является возможность разработки приложений на многих языках программирования. В эту ОС встроенная виртуальная машина JVM, что позволяет с легкостью разрабатывать мобильные приложения на языках Java и Kotlin.

Помимо этого, есть возможность использовать для разработки ПО многие другие языки программирования. Реализовано это с помощью программного комплекса Android Native Development Kit (ANDK). С помощью ANDK можно с поразительной легкостью создавать и запускать приложения на Си-подобных языках – C, C++ и C#.

Разработанное приложение можно выпустить как на предназначенных для этого площадках, после прохождения всех необходимых барьеров безопасности, так и просто в сети Интернет.

Android является системой с открытым исходным кодом. Лишь некоторые корпоративные приложения данной системы имеют закрытый код, например, Chrome, Play Market, YouTube.

Главным рекомендованным средством разработки приложений является пакет Software Development Kit. Большой популярностью так же пользуются Android Studio и IntelliJ IDEA.

# **1.3.3 Вывод**

Подводя итог, справедливо утверждать, что выбор операционной системы стоит остановить на Android, так как данная ОС подойдет для проектируемого приложения как нельзя лучше. Она обладает серьезными преимуществами, а именно:

* Большое число устройств в мире, работающих на этой ОС;
* Свободное распространение приложений, возможность установки приложений из сторонних источников;
* Большое количество различных средств разработки, предназначенных для этой ОС;
* Открытый исходный код системы.

Согласно исследованию от компании StockApps на ноябрь 2022 года доля мобильных устройств на базе ОС Android составляет 69.74%. На втором месте стоит операционная система iOS – 25.49%. Оставшиеся 4.77% делят между собой прочие операционные системы. Диаграмма распределения пользователей изображена на рисунке номер 3.

Рисунок 3 – Статистика использования мобильных операционных систем

# **1.4 Выбор средств разработки для ОС Android**

Существует огромное количество средств разработки, предназначенных для операционной системы Android. Средства разработки намного разнообразнее и функциональнее аналогов, предназначенных для ОС iOS. Связано это с широким распространением Android, а также с тем, что данная операционная система имеет открытый исходный код, что, конечно, играет на руку разработчикам программного обеспечения.

На момент написания данной курсовой работы самыми популярными и актуальными средствами разработки для ОС Android являются:

* Eclipse
* Android Studio;
* Kivy/KivyMD;
* Xamarin.Forms;

# **1.4.1 Среда разработки Eclipse**

Eclipse – это бесплатная среда разработки от некоммерческой организации Eclipse Foundation. Данная среда представляет собой основу, к которой подключаются различные плагины и модули. Например, Java Development Tools (для создания приложений на Java), C/C++ Development Tools (для разработки программ на языке C или C++) и т.д.

Данная среда разработки обладает следующими преимуществами:

* Высокая производительность;
* Большое число плагинов и дополнений;
* Возможность групповой разработки ПО.

Однако, данная среда разработки не лишена минусов. Главный минус заключается в том, что поддержка функционала, необходимого для разработки мобильных приложений, была прекращена в 2014 году. Так же, к минусам можно отнести скудный базовый функционал (без плагинов и дополнений).

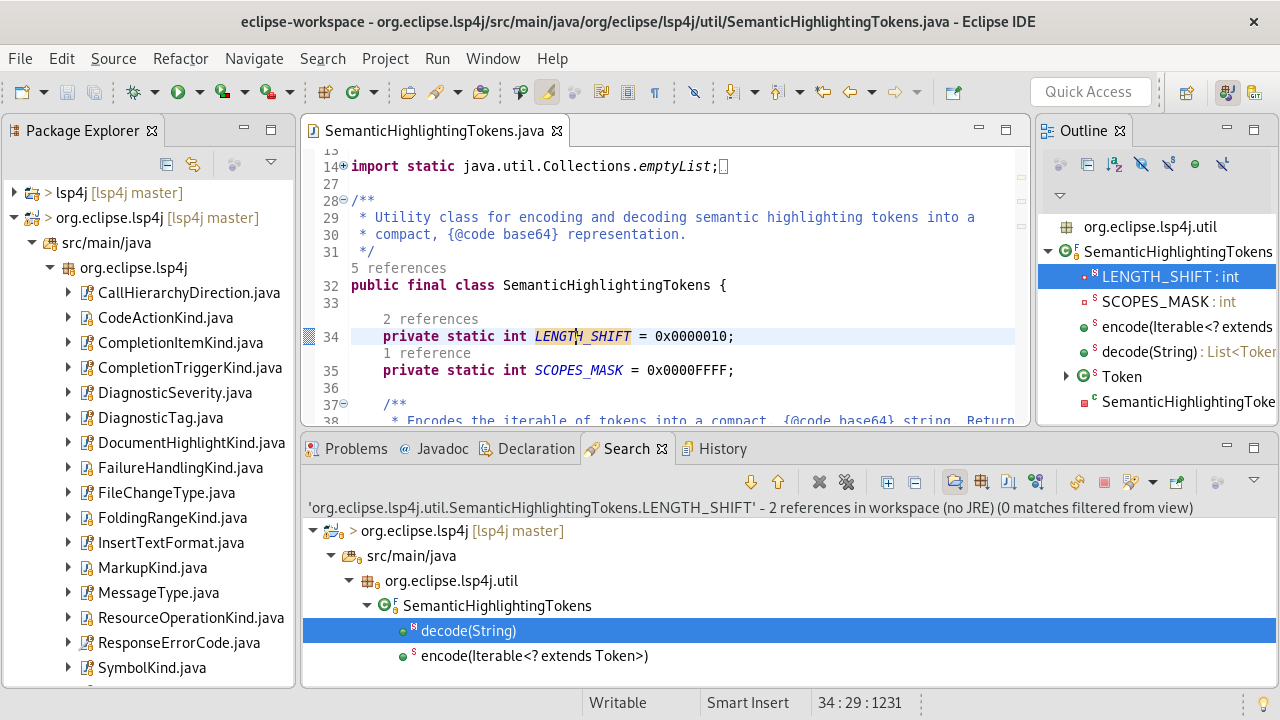


Рисунок 4 – Внешний вид Eclipse

# **1.4.2 Среда разработки Android Studio**

Android Studio является мощной средой разработки от компании Google для операционной системы Android; способна работать на всех известных операционных системах для настольных компьютеров и ноутбуков. Была выпущена компанией Google в 2013, основана на известной IDE IntelliJ IDEA от компании JetBrains.

Android Studio предоставляет пользователям множество интересных инструментов и возможностей:

1. Созданное приложение можно просматривать на нескольких вариантах разрешения экрана;
2. Возможность разработки не только для телефонов, но и для различных устройств на базе Android (смарт-часы, проекторы, телевизоры);
3. Широкие возможности редактора макетов и пользовательского интерфейса;
4. Анализатор кода для нахождения различных ошибок и проблем с производительностью;
5. Инструменты для работы со множеством языков программирования, включая Kotlin, Java, C++/C;
6. Встроенные средства контроля версий.

Одна из главных особенностей Android Studio – встроенный эмулятор с поддержкой тестирования готовых приложений на физическом устройстве – разработчик может подключить смартфон через USB и запустить созданное приложение на нём. К минусам данной среды можно отнести:

1. Низкую производительность;
2. Громоздкий, неинтуитивный интерфейс;
3. Очень низкая производительность;
4. Неудобная процедура установки библиотек.

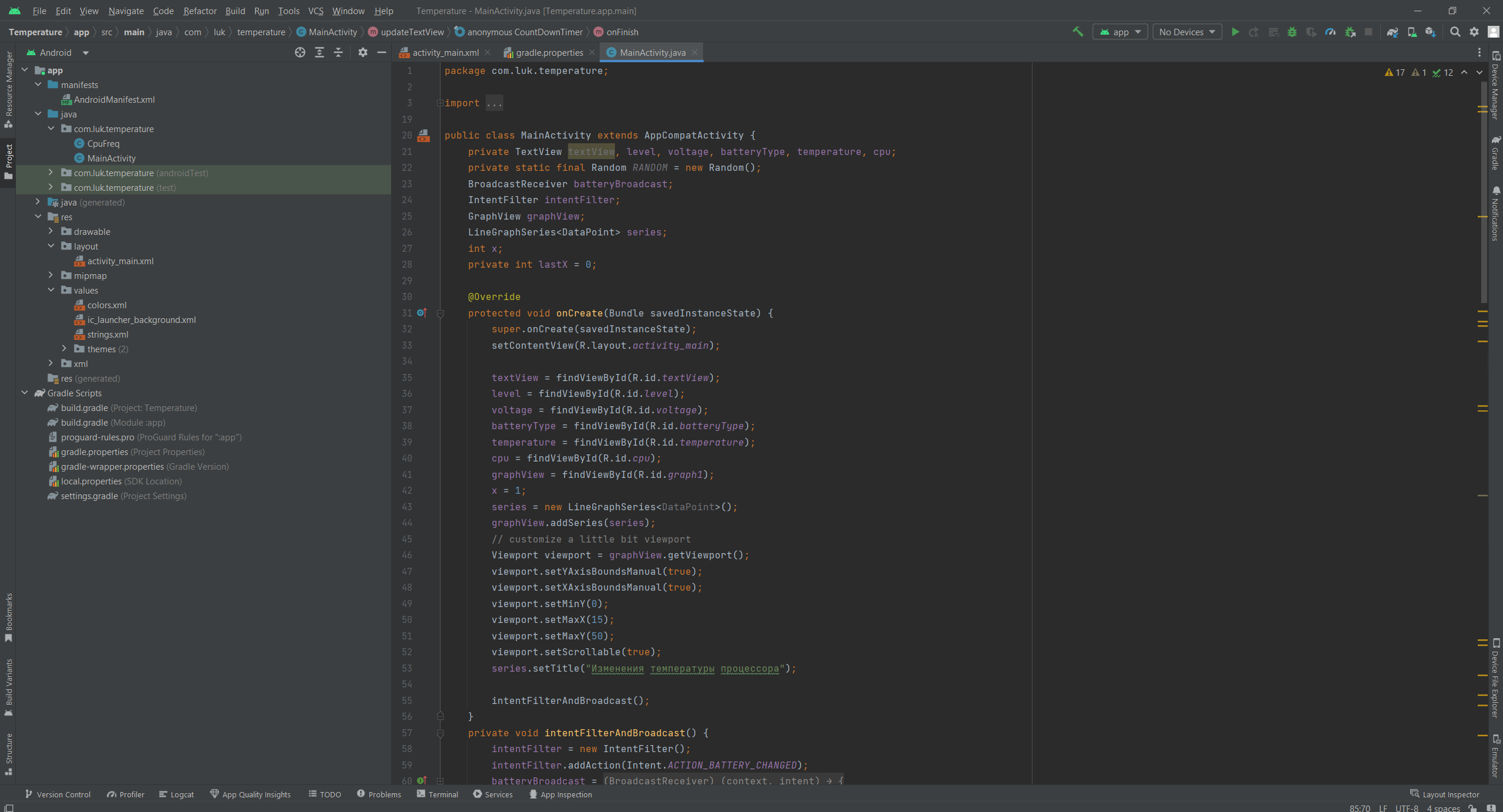


Рисунок 5 – Внешний вид Android Studio

# **1.4.3 KivyMD**

Данное средство разработки представляет собой фреймворк для разработки функциональных кроссплатформенных графических приложений на языке Python, которые могут скомпилированы для системы Android.

KivyMD содержит большое количество элементов пользовательского интерфейса в стиле Material Designс большим количеством настраиваемых параметров. К весомым преимуществам данного фреймворка относятся:

1. Простоту и скорость разработки приложений;
2. Широкий выбор настроек и параметров для каждого элемента интерфейса;
3. Возможность отладки в реальном времени;
4. Возможность быстро проектировать графический интерфейс приложения на собственном языке разметки – KvLang;
5. Полная кроссплатформенность. Однажды написанное приложение будет исправно работать в любой операционной системе.

К минусам данного фреймворка можно отнести следующие пункты:

1. Долгая компиляция мобильного приложения;
2. Проблематичная отладка приложения.

# **1.4.4 Xamarin.Forms**

Данное средство разработки представляет собой согласованный API на языке C# для создания кроссплатформенных приложений под разные системы, включая iOS, Android, Windows и прочее.

Xamarin.Forms имеет свой внутренний язык текстовой разметки XAML для создания пользовательского интерфейса. Программирование и добавление функциональности осуществляется на языке С#.

Данное средство мобильной разработки обладает следующими преимуществами, а именно:

1. Наличие собственного языка разметки XAML;
2. Разработка ведется на современном объектно-ориентированном языке – C#;
3. Быстрая работа эмулятора;
4. Возможность эмулировать работу широкого спектра устройств.

К недостаткам можно отнести:

1. Высокий уровень сложности разработки мобильных приложений;
2. Крайне долгая и проблематичная настройка начального окружения среды.

# **1.4.5 Вывод**

После проведенного анализа самых популярных средств разработки для мобильной операционной системы Android было решено выбрать KivyMD, так как с помощью данного фреймворка можно легко и быстро создавать эффективные приложения для мобильных устройств, используя удобный язык разметки KvLang и мощный функционал языка программирования Python. Стоит отметить, что фреймворк очень легко устанавливается и сам настраивает начальное окружение. А приложения, написанные с помощью KivyMD, обладают очень высокой скоростью выполнения.

2. ПРАКТИЧЕСКАЯ ЧАСТЬ

# **2.1 Определение требований**

Разрабатываемый программный продукт обязан соответствовать следующим требованиям:

1. Иметь минималистичный, адекватно воспринимающийся интерфейс;
2. Обладать высокой скоростью работы;
3. Обладать широким функционалом в плане конвертации чисел (возможность переводить число в любую из стандартных и нестандартных систем счисления).

Приложение должно содержать в себе три страницы пользовательского интерфейса – главную страницу, страницу стандартных преобразований и страницу «о приложении». На первых двух страницах должно быть поле ввода чисел, кнопки для преобразований и текстовое поле для вывода результата. Страница «о приложении» должна содержать ознакомительную информацию о использованных средствах разработки. Все страницы должны содержать заголовок и меню.

Для реализации данного программного продукта должен использоваться фреймворк KivyMD, в качестве основного языка программирования – Python, в качестве языка разметки – KvLang. Компиляция приложения будет производиться стандартной утилитой фреймворка buildozer.

# **2.2 Проектирование пользовательского интерфейса**

Было принято решение, что фон приложения будет темным, все основные элементы управления будут голубого цвета, весь текст выполнен в белом цвете.

Внешний вид главной страницы приложения представлен на рисунке 6.

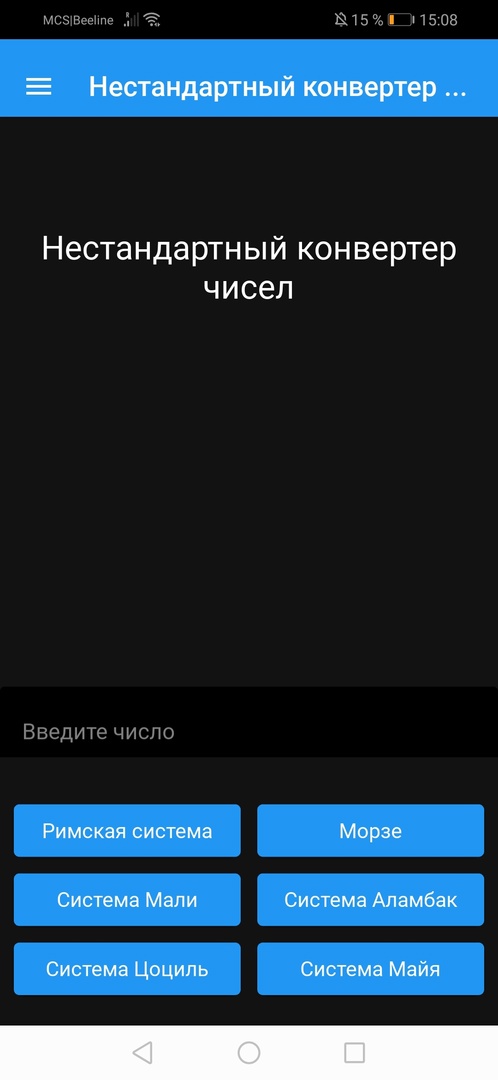


Рисунок 6 – Макет главной страницы приложения

Главная страница приложения позволяет переводить числа в следующие системы: Римскую, Морзе, Мали, Аламбак, Цоциль и Майя. Определим, что из себя представляют эти системы.

Римская система счисления - непозиционная система счисления, в которой для записи чисел используются буквы латинского алфавита: 1 - I, 5 - V, 10 - X, 50 - L, 100 - C, 500 - D и 1000 - M. При этом, если большая цифра стоит перед меньшей, то они добавляются, если же меньшая – перед большей, то меньшая вычитается из большей.

Азбука Морзе - способ знакового кодирования, в котором буквы алфавита, цифры, знаки препинания и другие символы представляются в виде последовательностей коротких и длинных сигналов, называемых точками и тире.

Система Мали – Непозиционная система счисления, в которой для исчисления применяются только числа 1, 5, 10, 20, 80 и 400, а также операции сложения и умножения.

Аламбак – аналогичная система, только ограничена числами 1, 2, 5 и 20.

Система цоциль – непозиционная система, в которой любое число записывается в виде «n человек m пальцев». Любое число сначала делится нацело на 20, в результате чего получаем количество «человек». Остатком от этого деления являются «пальцы».

Система Майя – непозиционная система счисления, в которой для записи чисел используются слова: 1 – тун, 20 – катун, 400 – бактун, 8000 – пиктун, 160000 – калабтун, 3200000 – кинчильтун, 64000000 – алавтун.

# **2.3 Реализация приложения**

Было решено, что весь пользовательский интерфейс будет написан на языке KvLang. Пример оформления одной из страниц интерфейса можно наблюдать на рисунке 7, оформление и логику меню навигации – на рисунках 8 и 9 соответственно. Весь основной программный код был написан на языке Python, потому что данный язык обладает следующими преимуществами:

1. Лаконичный синтаксис: код понятен и логичен;
2. Интерпретируемость, что позволяет легко отлаживать ход и значительно ускоряет разработку приложения;
3. Кроссплатформенность;
4. Высокая производительность.

Функции и реализованные методы будут опущены в данной курсовой работе в связи с большим количеством программного кода. На рисунке ниже продемонстрировано создание макета сетки GridLayout, В котором все элементы будут расположены на 5 строк с промежутком 24dp. Далее идет создание верхнего активного заголовка с кнопной меню, которая вызывает лямбда-функцию раскрытия меню, затем идет создание заголовка с названием страницы, заголовка вывода результата и поля ввода значений. Следом снова создается макетная сетка, в которую записываются кнопки преобразований, вызывающие соответствующие функции.

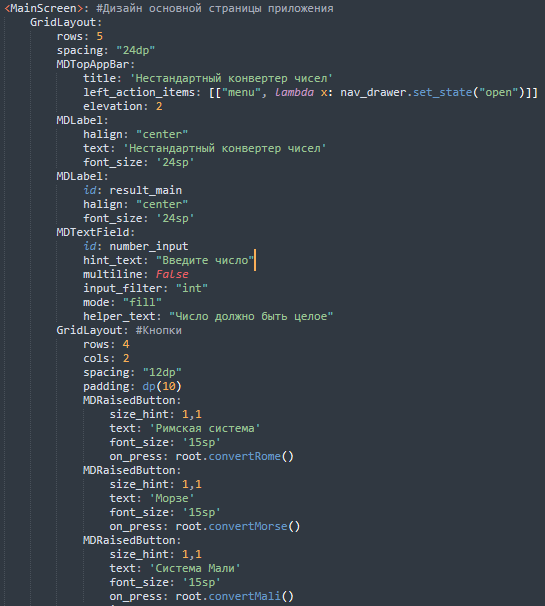


Рисунок 7 – Дизайн главной страницы приложения

На рисунке 8 изображен код дизайна меню. Для начала создается объект MDNavigationDrawer, представляющий собой отрисовщик меню. В нем создается сетка элементов BoxLayout с ориентацией объектов по вертикали и промежутком 10dp. Здесь же создается заголовок «Меню навигации» и прокручивающийся список ScrollView, содержащий список элементов с иконкой, именуемых OneLineIconListItem, каждый из которых при нажатии открывает соответствующую страницу пользовательского интерфейса.



Рисунок 8 – Дизайн меню навигации

На рисунке 9 описана логика методов меню, переключающих страницы пользовательского интерфейса. Каждый метод сначала вызывает функцию закрытия меню, затем переключает пользовательский экран на соответствующую страницу.

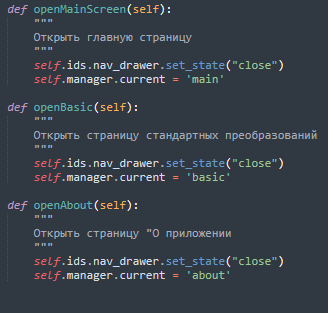


Рисунок 9 – Логика меню навигации

# **2.4 Тестирование приложения**

Тестирование реализованного приложения проводилось на нескольких устройствах, работающих на базе операционной системы Android, чтобы выяснить, какие программные ошибки были допущены при разработке программного продукта.

Тестирование программного решение проводилось непосредственно разработчиком приложения в ручном режиме. В ходе проведенного тестирования было определено, что есть трудности с определением изначальной системы счисления при вводе значений стандартных преобразований. Было принято решение использовать специальные указатели перед числом, чтобы обозначить систему счисления. Использованы указатели: 0b – для двоичных чисел, 0o – для восьмеричных и 0x – для шестнадцатеричных. Если число введено без указателя – то оно принимается за десятичное. Например: 0b01101 – двоичное число. 123 – десятичное.

Помимо этого, были обнаружены мелкие ошибки, такие как некорректное отображение элементов интерфейса. Однако, все ошибки были исправлены и можно считать, что разработанный программный продукт готов к использованию конечным пользователем.

ЗАКЛЮЧЕНИЕ

В процессе выполнения курсовой работы было спроектировано, разработано и протестировано приложение «Нестандартный конвертер чисел» на языке программирования Python с использованием фреймворка KivyMD. В процессе работы были изучены и рассмотрены принципы построения и разработки мобильных приложений на языке Python.

Были получены следующие навыки:

1. Навыки планирования и проектирования мобильных приложений;
2. Навыки работы с языком разметки KvLang и фреймворком KivyMD;
3. Навыки отладки и тестирования мобильных приложений;
4. Умения компилировать мобильные приложения.

Приложение представляет собой конвертер чисел, способный как на общепринятые преобразования в двоичную, шестнадцатеричную, десятичную и восьмеричную системы, так и в экзотические системы счисления. Оно имеет минималистичный, удобный и наглядный интерфейс, отвечающий пользовательским требованиям.

Выполнение данной курсовой оказалось сложным и многогранным процессом, который привел к приобретению многих новых навыков. В процессе работы были значительно усовершенствованы навыки программирования и обработки информации, а также понимания принципов преобразования и записи чисел в различных системах счисления.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. ОС Android, Википедия – свободная энциклопедия [Электронный ресурс]. URL: https://ru.wikipedia.org/wiki/Android. (Дата обращения: 02.05.2023);
2. ОС iOS, Википедия – свободная энциклопедия [Электронный ресурс]. URL: https://ru.wikipedia.org/wiki/IOS. (Дата обращения: 02.05.2023);
3. Android Studio, Википедия – свободная энциклопедия [Электронный ресурс]. URL: https://ru.wikipedia.org/wiki/Android\_Studio. (Дата обращения: 02.05.2023);
4. Material Design [Электронный ресурс]. URL: <https://m3.material.io/>. (Дата обращения 04.05.2023);
5. StockApps, Исследовательская группа [Электронный ресурс]. URL: <https://stockapps.com/blog/android-loses-8-of-its-global-os-market-share-in-five-years/>. (Дата обращения:05.05.2023).
6. Stack Overflow. URL: https://stackoverflow.com/ (Дата обращения: 10.04.2023)
7. Kivy Docs, URL: <https://kivy.org/doc/stable/> (Дата обращения: 25.04.2023)
8. Андреева Е.В., Босова Л.Л., Фалина И.Н. Математические основы информатики. Элективный курс: Учебное пособие. М.: БИНОМ. Лаборатория знаний, 2005.