

Министерство образования Новосибирской области
ГБПОУ НСО «Новосибирский авиационный технический колледж
имени Б.С. Галушака»

РАЗРАБОТКА МОБИЛЬНОГО ПРИЛОЖЕНИЯ ДЛЯ ЗАПИСИ ЗВУКОВ

Пояснительная записка к курсовому проекту
ПМ.01 Разработка модулей программного обеспечения
для компьютерных систем
МДК.01.03 Разработка мобильных приложений
НАТКиГ.201500.010.000ПЗ

Разработал:
студент группы ПР-21.101
Мысин А.К.

Содержание

Введение.....	3
1 Исследовательский раздел	4
1.1 Описание предметной области	4
1.2 Образ клиента	4
1.3 Сценарии	5
1.4 Сбор и анализ прототипов.....	5
2 Проектирование приложения.....	10
2.1 UI/UX дизайн проекта:	10
2.2 Выбор технологии, языка и среды программирования:	12
3 Разработка мобильного приложения.....	13
3.1 Взаимодействие с файловой системой	13
3.2 Разработка мультимедийного контента.....	13
3.3 Описание используемых плагинов:	14
3.4. Описание разработанных процедур и функций.....	16
4 Тестирование	20
4.1 Протокол тестирования дизайна приложения	20
4.2 Протокол тестирования функционала приложения	22
Заключение	25
Библиография	26
Приложение А (обязательное) Техническое задание.....	27

					<i>НАТКиГ.201500.010.000ПЗ</i>		
Изм.	Лист	№ докум	Подпись	Дата			
Разраб		Мысин А.К.			РАЗРАБОТКА МОБИЛЬНОГО ПРИЛОЖЕНИЯ ДЛЯ ЗАПИСИ ЗВУКОВ		Литера
Пров		Санников И.М.					Лист
							Листов
Н. Контр		Тышкевич Е. В.			<i>ПР-21.101</i>		
Утв		Тышкевич Е. В.					

Введение

Современное информационное общество стремительно развивается, и с ним растет потребность в эффективных и удобных инструментах для получения актуальной информации. В этом контексте особенно важными становятся мобильные приложения, обеспечивающие быстрый и легкий доступ к разнообразным сервисам. Таким, как запись видео, съемка фотографий, запись разнообразных песен, лекций, диалогов, звуков и их редактирование.

Значимость темы проекта обусловлена образом жизни современного общества, где время является ценным ресурсом. Приложение для звукозаписи позволит распоряжаться своим временем более эффективно, занимаясь другими делами, не требующими отлагательств, пока приложение записывает необходимую для прослушивания аудиоинформацию.

Целью курсового проекта является разработать мобильное приложение для звукозаписи, которое направлено на удовлетворение потребностей пользователей.

Задачи:

- изучение предметной области;
- рассмотрение приложения с точки зрения пользователя для выявления функций приложения;
- разработка и реализация дизайна приложения;
- написание кода приложения;
- тестирование полученного продукта.

Реализуемый продукт - мобильное приложение, предназначенное для записи звуков.

1 Исследовательский раздел

1.1 Описание предметной области

Приложение для записи звуков предоставляет удобный способ зафиксировать моментальные мысли, деловые встречи или важные презентации, которые могут легко потеряться в потоке повседневных дел. Они позволяют пользователям делиться идеями, записывать лекции или просто сохранять интересные звуки из окружающей среды. Это особенно важно для студентов, которые могут записывать лекции и далее переслушивать их для более глубокого понимания материала. Профессионалы также могут использовать диктофоны для записи важных моментов в рабочей среде, будь то встречи, интервью или идеи для проектов.

Однако, просто записывать аудио может быть недостаточно эффективным, если нет возможности быстро находить нужные моменты в записях. Поэтому важно, чтобы диктофонные приложения предоставляли возможность организации записей по категориям или темам. Это позволяет пользователям легко находить нужную информацию в больших объемах записей, сокращая время, затраченное на поиск конкретного момента или идеи. Такие функции существенно повышают эффективность использования диктофона как инструмента для сохранения и организации информации. С помощью диктофона можно легко управлять своими заметками и аудиозаписями, сокращая время, затраченное на поиск необходимой информации.

1.2 Образ клиента

Основной целевой аудиторией для диктофона являются пользователи мобильных устройств, которым требуется функция записи звука. Это могут быть студенты, использующие приложение для записи лекций, журналисты, записывающие интервью или человек пришедший на концерт

					НАТКиГ.201500.010.000ПЗ	Лист
Изм.	Лист	№ докум	Подпись	Дата		4

записывающий музыку, и многие другие. Важно учесть предпочтения и потребности этих пользователей при разработке приложения, чтобы обеспечить его удобство и функциональность

1.3 Сценарии

Сценарии – это правдоподобные ситуации, происходящие с каждым образом клиента.

Сценарий 1: Алексей, студент вуза, посещающий множество лекций. Из-за большой нагрузки он не успевает записывать всю лекцию, поэтому он использует диктофон для их записи. Прослушивая записанные лекции вечером или в другое свободное время, чтобы не пропустить пройденные на лекции темы, переписывая лекции в тетрадь или переносить на другой рабочий носитель.

1.4 Сбор и анализ прототипов

В GooglePlay и AppStore существует несколько приложений, выполняющих роль приложения для звукозаписи, но мы рассмотрим только два из них. Первое – «Простой диктофон» (Русское приложение), второе – «Диктофон» (Русское приложение). Оба этих приложения имеют возможность записывать и проигрывать звуки. Рассмотрим эти два приложения подробнее и сравним их.

Оба этих приложения имеют главный экран, на котором есть соответствующая кнопка (рисунок 1).

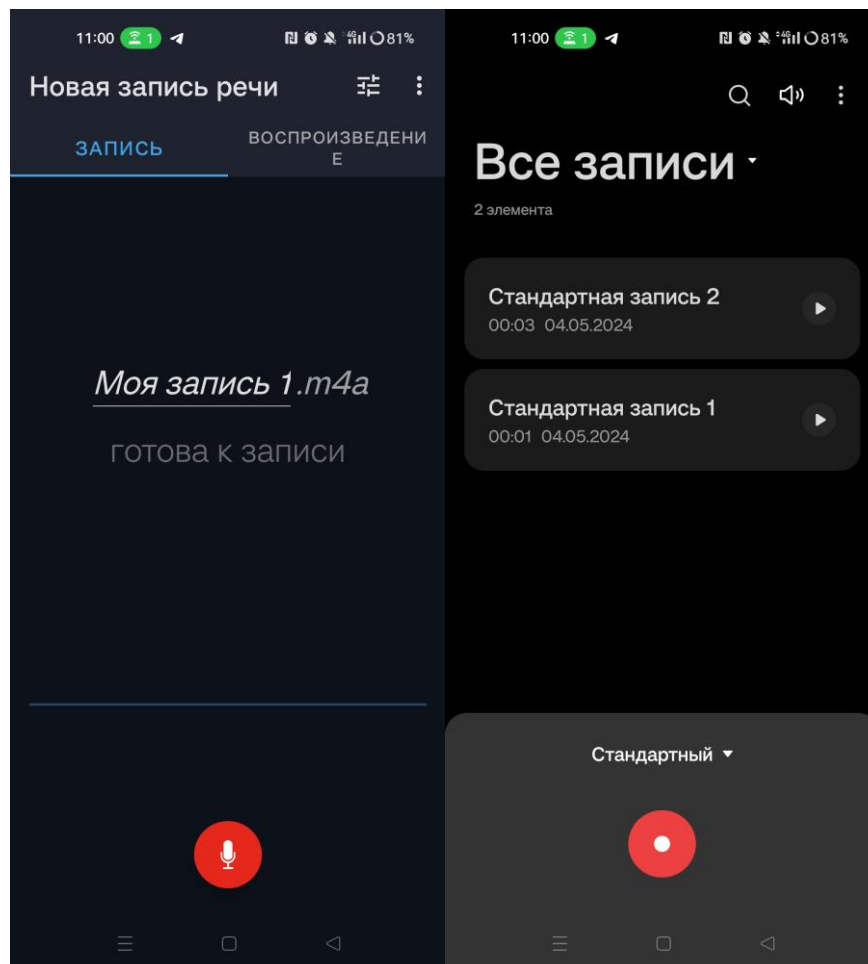


Рисунок 1 — Главный экран приложений.

В приложениях на главном экране можно увидеть верхнюю панель, отвечающие за навигацию по приложению и поиск. На экранах в обоих приложениях почти одинаковые кнопки:

- настройки;
- кнопка записи;
- режим записи.

Во время записи есть небольшое различие. В приложениях есть звуковая волна, показывая громкость поступающего звука, но в приложении «Диктофон» - есть возможность ставить закладки на определенные моменты записи как текстом, так и фотографией, а кнопки сохранения или отмены записи находятся наверху экрана. В приложении «простой диктофон» не реализована функция закладок, но кнопки

сохранения и отмены записей расположены рядом с кнопкой паузы внизу экрана, что является хорошим решением (Рисунок 2).

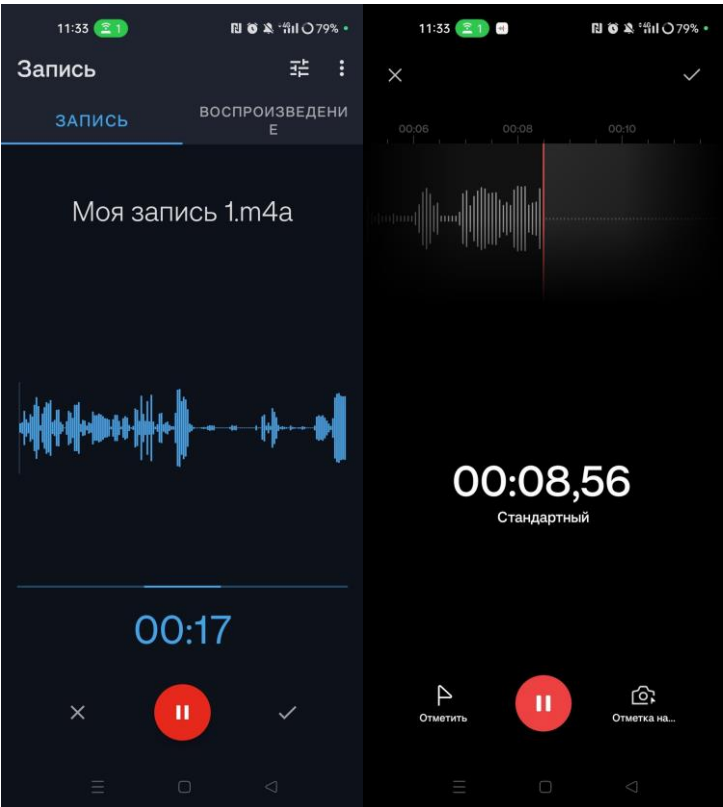


Рисунок 2 — Экраны записи приложений.

Вывод аудиозаписей отличается. В приложении «Простой диктофон» - вывод файлов осуществляется на другом экране, а в «Диктофон» – вывод осуществлен на главном экране. «Простой диктофон» находится слева, а приложение «Диктофон» – находится справа (Рисунок 3):

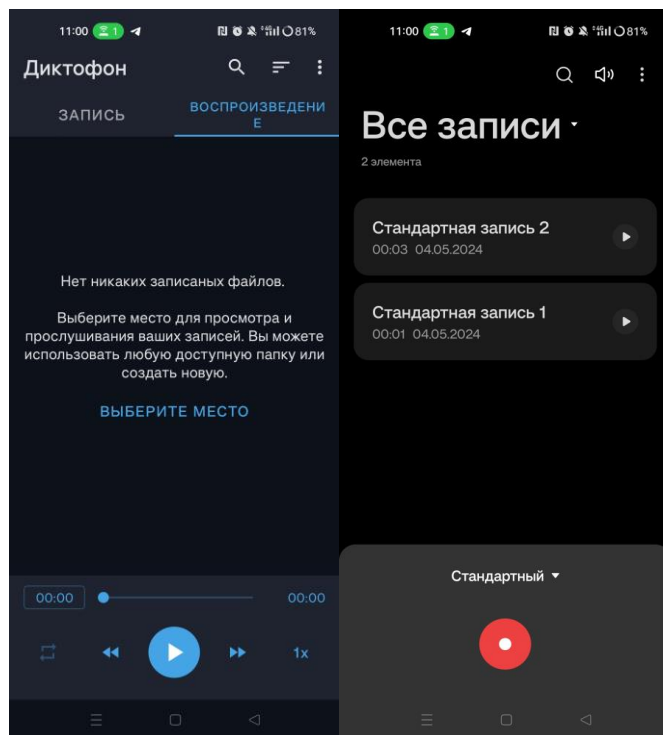


Рисунок 3 — Экран сохраненных записей.

На экране настроек в приложении «Простой диктофон» заметно, что вариантов настроек намного больше, чем в «Диктофоне», в котором есть только выбор формата файла (Рисунок 4).

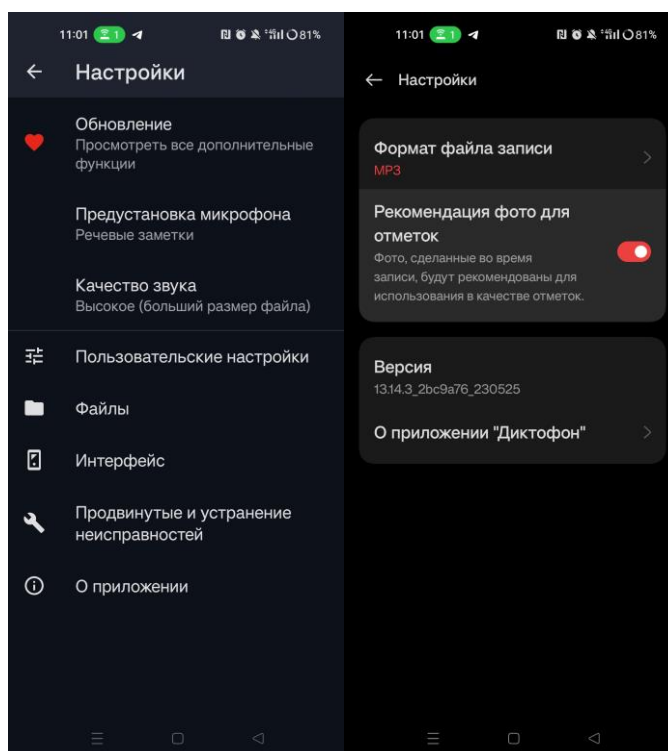


Рисунок 4 — Экран настроек приложений.

Сравнение по основным критериям данных приложений представлено в таблице 1.

Таблица 1 – Сравнение приложений

Параметры	Простой диктофон	Диктофон
Стоимость	Бесплатно	Бесплатно
Основной язык	Русский	Русский
Возможность записывать аудио	Есть	Есть
Возможность менять название файла	Есть	Есть
Возможность ставить закладки от определенного времени	Отсутствует	Есть
Наличие поиска	Есть	Есть
Выбор темы интерфейса	Отсутствует	Отсутствует
Возможность менять формат файла	Есть	Есть
Возможность прослушивать существующие записи	Есть	Есть
Наличие сортировки	Есть	Отсутствует

Рассмотрев пару приложений, выполняющих похожие задачи, было решено написать приложение, которое имело бы простой интерфейс без лишних функций и с акцентом внимания на важных для пользователя элементах, в котором можно записывать окружающие звуки, ставить запись на паузу и менять название записи. За основу будут взяты разные элементы двух приложений, так как каждое из них имеет как свои плюсы, так и минусы.

2 Проектирование приложения

2.1 UI/UX дизайн проекта:

Дизайн проекта разработан в программе Figma.

Для проекта были определены основные экраны:

- экран записи;
- экран с сохраненными аудио;
- экран с настройками.

Для темы приложения определена следующая цветовая схема. Тема реализуется в ярких тонах, поэтому в ней основными цветами являются: красный, черный, белый, темно-серый. Фон оформлен в белых ярких цветах. Навигационная панель и другие интерактивные элементы выделены ярким красным или черным цветом. Все используемые цвета показаны на рисунке 5.

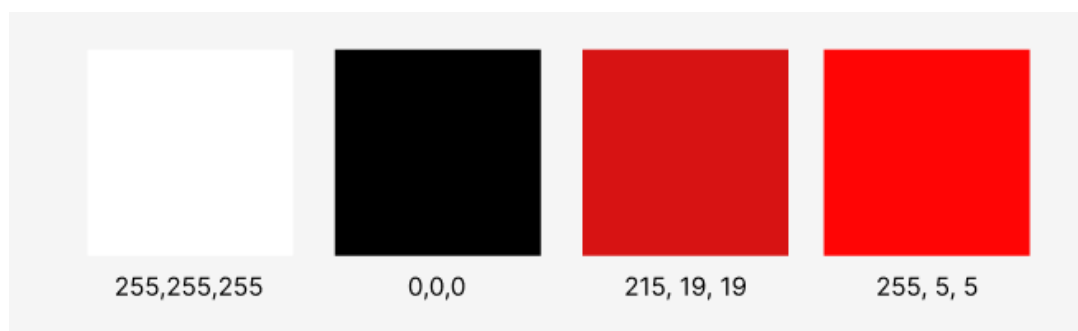


Рисунок 5 — Цветовая схема.

Логотип приложения представлен белым фоном, с изображенным на нем красным кругом, окружающим микрофон.

Ниже на рисунке 6 представлен логотип приложения для звукозаписи.



Рисунок 6 — логотип приложения.

Цвет логотипа красный, который является ярким и бросающимся в глаза и привлекающим внимание цветом. Цвет логотипа представлен на рисунке 7.

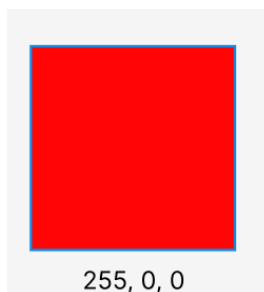


Рисунок 7 — цвет логотипа приложения.

Определившись с цветовой схемой приложения и создав его логотип, был разработан дизайн следующих экранов:

- экран записи (MainActivity);
- экран с сохраненными аудио (Storage);
- экран с настройками (Settings).

Ниже на рисунке 8 представлен дизайн приложения.

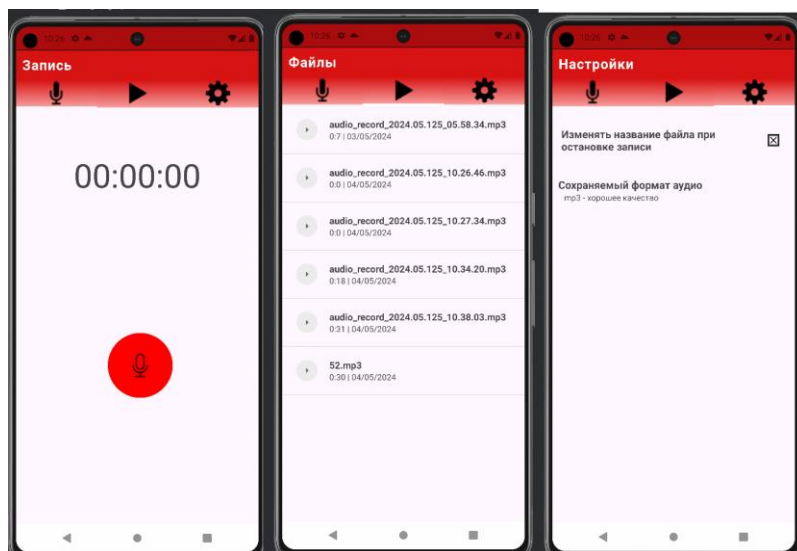


Рисунок 8 —Дизайн приложения.

При запуске приложения пользователь попадает на экран записи. При нажатии на одно из двух не выбранных полей навигационной панели, с главного экрана можно перейти на экран с сохраненными аудио или на экран настроек. Если пользователь запишет аудио, то оно сохранится и отобразится во вкладке сохраненных.

2.2 Выбор технологии, языка и среды программирования:

Средой программирования выбрана программа Android Studio, так как она довольно проста и удобна для создания приложения и является довольно популярной, также с данным приложением уже был опыт работы, поэтому в случае возникновения ошибки, легче найти способ её решения. Языком программирования является Java, который является одним из наиболее распространенных и широко используемых языков в мире программирования. Java обладает обширной экосистемой инструментов и библиотек, что делает его идеальным выбором для разработки мобильного приложения.

Работа с файлами в Android Studio может включать в себя множество задач, таких как чтение и запись файлов на внутреннее хранилище устройства, работа с файлами внешнего хранилища.

3 Разработка мобильного приложения

3.1 Взаимодействие с файловой системой

В момент, когда пользователь останавливает запись. Она сохраняется в файловой системе. При переходе на экран с сохраненными записями приложение подгружает все аудиофайлы из системы. Пользователь также может вносить изменения в отображаемые данные, например, изменять название файла или удалить его. После внесения изменений пользователь имеет возможность сохранить их обратно в файловую систему устройства для последующего доступа. Приложение также предоставляет функциональность управления файлами, удаление существующих и переименование файлов. Для работы с файловой системой приложение запрашивает необходимые разрешения на доступ, такие как разрешения на запись аудио, на чтение и запись во внутреннее хранилище устройства или на SD-карт, а также обеспечивает обработку возможных ошибок, связанных с доступом к файлам.

3.2 Разработка мультимедийного контента

Весь мультимедийный контент разрабатывался с помощью языка разметки XML. Вёрстка выполнялась по дизайну, разработанному ранее в приложении Figma. Однако, по мере разработки, в дизайн были внесены изменения.

Иконки, кнопки и другие элементы приложения, были импортированы в проект в виде XML-файлов (Рисунок 9). Такой способ хранения уменьшает вес приложения, а также, избавляет от проблем с потерей качества мультимедийного контента. Все ресурсы хранятся в папке «drawable».

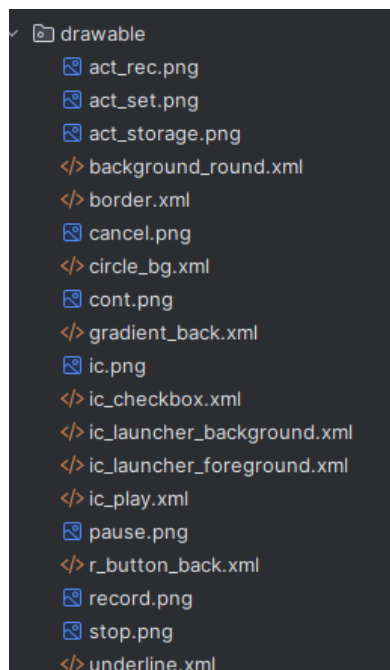


Рисунок 9 — Мультимедийный контент.

Приложение использует светлую цветовую схему, которая хранится в файле `colors.xml`. Эти цвета определяют тему приложения для светлой темы. Пользователь имеет возможность выбрать светлую тему, чтобы наслаждаться ярким и приятным визуальным опытом.

Цвета в файле `colors.xml` могут быть адаптированы под различные элементы пользовательского интерфейса, такие как фоны, текст, кнопки и т. д.

3.3 Описание используемых плагинов:

В проекте применяются библиотеки, содержащие разнообразные компоненты и функции. Их полный список и описание представлены в таблице 2.

Таблица 2 — Библиотеки и их описание

Библиотека	Описание
androidx.appcompat	Предоставляет классы для создания современных пользовательских интерфейсов, совместимых с различными версиями платформы Android.
android.widget	Содержит базовые виджеты пользовательского интерфейса, такие как кнопки, текстовые поля, изображения и другие.

Продолжение таблицы 2

libs.constraintlayout	Эта библиотека представляет ConstraintLayout, который является гибким и эффективным макетом для размещения элементов пользовательского интерфейса в Android-приложениях.
libs.ext.junit	Эта библиотека представляет расширение JUnit для написания и запуска инструментальных тестов Android-приложений. Она обеспечивает интеграцию с Android SDK и предоставляет возможности для тестирования.
libs.ext.junit	Эта библиотека представляет Espresso, фреймворк для написания и запуска функциональных тестов Android-приложений.
android.widget.TextView	Виджет для отображения текста на экране.
android.widget.EditText	Виджет для ввода текста пользователем.
android.widget.Button	Виджет для создания кнопок в пользовательском интерфейсе.
android.widget.ImageView	Виджет для отображения изображений на экране.
android.widget.LinearLayout	Группирует другие виджеты в вертикальную или горизонтальную последовательность.
android.content.Intent	Представляет намерение (интент) для выполнения операций в приложении, таких как запуск другой активности или сервиса.
java.util.ArrayList	Реализация списка, который расширяет класс AbstractList и реализует интерфейсы List, Collection и другие.
android.graphics.Color	Предоставляет методы для работы с цветами в Android приложениях.
android.graphics.Bitmap	Представляет изображение в виде массива пикселей.
android.widget.Toast	Класс для отображения всплывающих уведомлений на экране устройства.
android.view.View	Представляет базовый строительный блок пользовательского интерфейса Android, который может содержать различные виджеты (кнопки, текстовые поля и т. д.).

Вышеперечисленные библиотеки играют ключевую роль в разработке приложения, обеспечивая его функциональность и возможности. Они предоставляют инструменты и функции, необходимые для реализации задуманных идей и обеспечивают правильное функционирование приложения.

Без этих библиотек приложение не смогло бы реализовать все свои функции. Они являются неотъемлемой частью разработки приложения, обеспечивая его успешное выполнение и повышая его привлекательность для пользователей.

3.4. Описание разработанных процедур и функций

В приложении реализованы следующие методы (таблица 3):

Таблица 3 — Методы приложения

Метод	Описание
onCreate(Bundle savedInstanceState)	Метод, который вызывается при создании активности. В этом методе инициализируются элементы интерфейса и устанавливаются слушатели для кнопок.
onActivityResult(int requestCode, int resultCode, Intent data)	Метод, который вызывается при получении результата от другой активности, запущенной с помощью.
startActivityForResult()	Метод используется для получения выбранного изображения из галереи.
onClick(View v)	Метод, который вызывается при щелчке на кнопке. В этом методе определяется поведение для каждой из кнопок.
checkPermissions()	Метод проверяющий на наличие разрешений доступа к использованию микрофона и доступа к памяти

Продолжение таблицы 3

requestPermissions()	Метод, который проверяет были ли даны разрешения доступа.
onRequestPermissionsResult(int requestCode, @NonNull String[] permissions, @NonNull int[] grantResults)	Метод, подтверждающий или опровергающий, в зависимости от предоставленных разрешений, их наличие.
NameFile()	Метод вызываемый после остановки записи, выводит окно с вопросом о переименовании записанного файла.
StartRecord()	Один из методов управления приложением, в нём устанавливаются все исходные данные, а так же путь для сохранения файла и его базовое название.
Cancel()	Один из методов управления приложением, в нём функция звукозаписи останавливается а записываемый файл удаляется.
StopRecord()	Один из методов управления приложением, в нём функция звукозаписи останавливает свою работу и сохраняет файл.
startTextAnimation()	Метод, который отвечает за запуск анимации текста на экране («Идет запись...»).

Так же в приложении есть классы, которые не имеют методов (таблица 4):

Таблица 4 — Классы приложения

Timer	Этот класс отвечает за отображение времени на экране. Он предоставляет методы для запуска, остановки и приостановки таймера.
-------	--

Продолжение таблицы 4

Adapter	Это адаптер, используемый для отображения списка файлов в ListView на экране. Он преобразует данные о файлах в элементы пользовательского интерфейса для их отображения.
ValueAnimator	Этот класс отвечает за анимацию изменения текста на экране. Он создает анимацию, которая постепенно изменяет текст, делая его более динамичным.
MediaRecorder	Этот класс предоставляет функциональность записи аудио с микрофона устройства. Он инициализирует, управляет и записывает аудиоданные в файл.

В приложении для звукозаписи, отсутствие авторизации и базы данных не является необычным, особенно если приложение ориентировано на простоту использования и сохранение файлов на устройстве пользователя. Вместо этого, редактор может предоставлять возможность прослушивать, редактировать и удалять аудиофайлы локально на устройстве.

Листинг 1 — Логика старта и остановки записи

```
private void StartRecord() {
    onRecord=true;
    play.setImageResource(R.drawable.pause);
    stop.setVisibility(View.VISIBLE);
    stop.setEnabled(true);
    cancel.setVisibility(View.VISIBLE);
    cancel.setEnabled(true);
    if (!checkPermissions()) {
        requestPermissions();
    }
    timer.startTimer();
    state.setVisibility(View.VISIBLE);
    startTextAnimation();
    dirName = getExternalCacheDir().getAbsolutePath()+"/";
    SimpleDateFormat sdf = new SimpleDateFormat("yyyy.MM.DD_hh.mm.ss");
    String date = sdf.format(new Date());
    fileName = "audio_record_"+date;
    mediaRecorder = new MediaRecorder();
    mediaRecorder.setAudioSource(MediaRecorder.AudioSource.MIC);
```

```

mediaRecorder.setOutputFormat(MediaRecorder.OutputFormat.MPEG_4);
mediaRecorder.setAudioEncoder(MediaRecorder.AudioEncoder.AAC);
mediaRecorder.setOutputFile(dirName+fileName+".mp3");
try{
    mediaRecorder.prepare();
    mediaRecorder.start();
}catch (IOException e){
    e.printStackTrace();
}
}
private void NameFile(){
    bottomSheetContainer.setVisibility(View.VISIBLE);
    bottomSheetContainer.setEnabled(true);
    fileNameInput.setText(fileName);
    timer.pauseTimer();
    mediaRecorder.pause();
    stopTextAnimation();
    onRecord = false;
}
private void StopRecord(){
    play.setImageResource(R.drawable.record);
    timer.stopTimer();
    stop.setVisibility(View.INVISIBLE);
    stop.setEnabled(false);
    cancel.setVisibility(View.INVISIBLE);
    cancel.setEnabled(false);
    state.setVisibility(View.INVISIBLE);
    Toast.makeText(this, "Файл сохранен в папке: "+dirName,
    Toast.LENGTH_SHORT).show();
    mediaRecorder.stop();

    mediaRecorder.release();
    mediaRecorder = null;
}

```

При нажатии пользователем на кнопку старта записи вызывается метод StartRecord(), в котором задается формат сохранения файла, устройство записи, его кодировка, путь для сохранения файла, а так же его базовое название, которое пользователь может изменить при остановке записи.

При нажатии же на кнопку стоп, которая появляется только после начала записи, пользователю предлагается сменить имя файла (если был выбран пункт «Спрашивать название записи» в настройках) в отдельном окне с полем для ввода названия и двумя кнопками «Отмена» и «Ок». При нажатии на кнопку «Отмена» запись просто удаляется, при нажатии же на кнопку «Ок» запись сохраняется со стандартным или же измененным названием.

4 Тестирование

4.1 Протокол тестирования дизайна приложения

Тестирование дизайна приложения проводится на Android SDK 30 и на более позднем Android SDK 33 с различной диагональю экранов для проверки разметки страниц и вёрстки приложения.

Примеры проверок отображения элементов на экране представлены на рисунках 7–9.

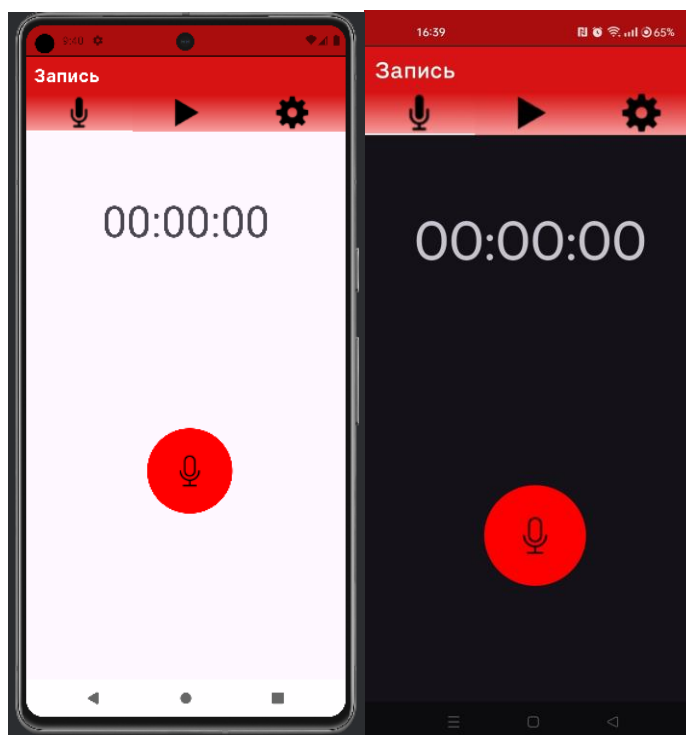


Рисунок 7 — Главный экран приложения

Элементы интерфейса в обоих случаях отображаются корректно.

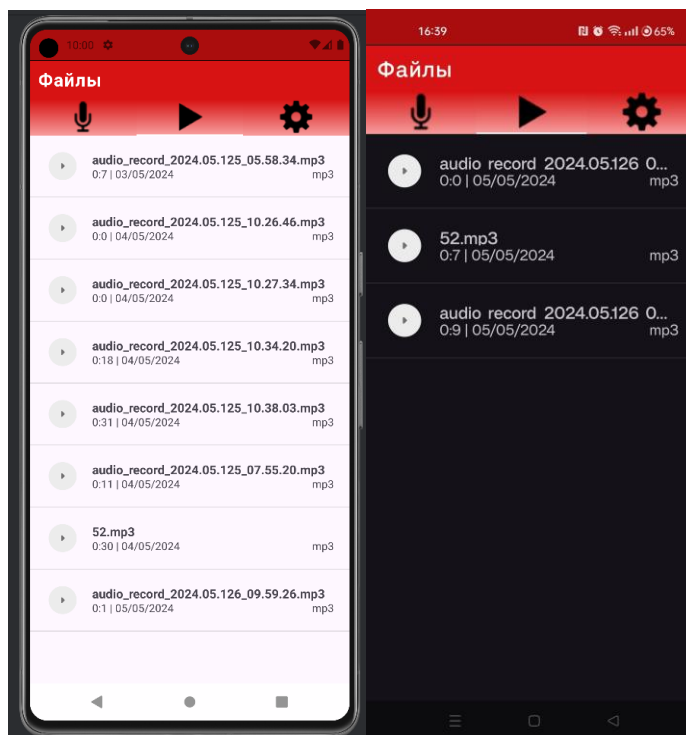


Рисунок 8 — Экран списка аудиофайлов.

Элементы интерфейса на этом экране так же отображаются корректно

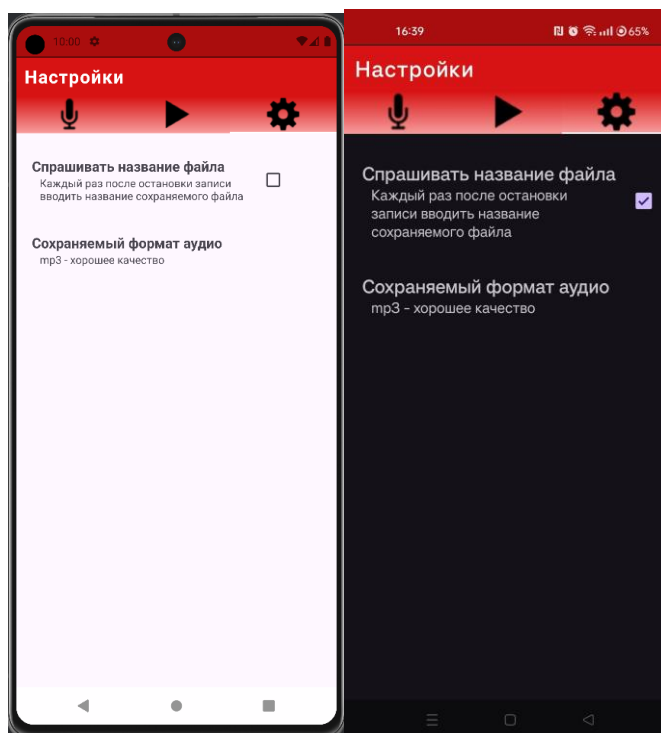


Рисунок 9 — Экран настроек.

Экран настроек так же отображается корректно на обоих устройствах, все элементы управления на своих местах.

UI-тестирование играет ключевую роль в обеспечении качества программного обеспечения и удовлетворении потребностей пользователей. Регулярное и комплексное тестирование пользовательского интерфейса помогает выявить и устранить проблемы до их выхода в продакшн.

4.2 Протокол тестирования функционала приложения

Так же необходимо проверить функционал приложения. Для этого, для каждой функции были разработаны TestCase.

В таблице 5 представлено тестирование функции создания нового изображения.

Таблица 5 – Тестирование функции старта записи

Название:	Recorder	
Функция:	Старт записи	
Действие	Ожидаемый результат	Результат теста: пройден провал заблокирован
Предусловие:		
Запустить приложение «Recorder»	Приложение открылось корректно	
Шаги теста (positive):		
Появление кнопок и установка изначальных параметров	Кнопки появились, а ошибок не возникло	пройден
Остановить запись	Запись успешно остановилась приложение не завершило свою работу	пройден
Ввести имя файла	Имя файла было введено и файл успешно сохранился	пройден
Шаги теста (negative):		
Появление кнопок и установка изначальных параметров	Кнопки появились, а ошибок не возникло	пройден
Остановить запись	Запись успешно остановилась приложение не завершило свою работу	пройден
Ввести имя файла	Появится всплывающее сообщение с определенной ошибкой	пройден

В таблице 6 представлено тестирование функции авторизации пользователя.

Таблица 6 – Тестирование функции отмены записи

Название:	Recorder	
Функция:	Отмена записи	
Действие	Ожидаемый результат	Результат теста: пройден провал заблокирован
Предусловие:		
Запустить приложение «Recorder»	Приложение открылось корректно	
Шаги теста (positive):		
Начать запись	Успешное старт записи	пройден
Нажать на кнопку остановки записи	Приложение остановило запись	пройден
Нажать на кнопку «Отмена» в окне переименования	Приложение удалило записанный материал	пройден
Шаги теста (negative):		
Начать запись	Успешное старт записи	пройден
Нажать на кнопку остановки записи	Приложение остановило запись	пройден
Не был активирован чекбокс на экране настроек	Запись была сохранена	пройден
Активировать чекбокс на экране настроек	Окно для переименования будет выводиться	пройден
Начать запись	Успешное старт записи	пройден
Нажать на кнопку остановки записи	Приложение остановило запись	пройден
Нажать на кнопку «Отмена» в окне переименования	Приложение удалило записанный материал	пройден

В таблице 7 представлено тестирование функции написания отзыва на игру.

Таблица 7 – Тестирование функции прослушивания аудио

Название:	Recorder	
Функция:	Прослушивание аудио	
Действие	Ожидаемый результат	Результат теста: пройден провал заблокирован
Предусловие:		
Запустить приложение «Recorder»	Приложение открылось корректно	
Шаги теста (positive):		

Продолжение таблицы 7

Перейти на экран с сохраненными аудиофайлами	Окно открыто успешно	пройден
Нажатие на аудиофайл с ненулевой длительностью	Успешное открытие плеера	пройден
Заккрыть плеер	Плеер закрыт воспроизведение остановлено	пройден
Шаги теста (negative):		
Нажатие на аудиофайл с нулевой длительностью	Появится всплывающее сообщение с определенной ошибкой	пройден
Нажатие на аудиофайл с ненулевой длительностью	Успешное открытие плеера	пройден
Заккрыть плеер	Плеер закрыт воспроизведение остановлено	пройден

Разработанные TestCase демонстрируют корректную работу функций приложений. Так же в процессе тестирования не выявлено ошибок в дизайне приложения и его логике. Каждый экран прошёл проверку на разных API на корректное отображение элементов соответствующих экранов.

Заключение

В ходе разработки мобильного приложения для звукозаписи были учтены предпочтения и потребности пользователей, в формате простого и удобного приложения. Функции для записи и проигрывания звуков успешно реализована, позволяя пользователям записывать и слушать происходящее вокруг них в любое время.

Интерфейс приложения разработан с учетом удобства использования и минимального количества действий для достижения пользовательской цели. Основные экраны спроектированы таким образом, чтобы не перегружать пользователей лишней информацией, добавлены полезные функции, такие как возможность сменить расширение файла и возможность моментально изменять название файла сразу же после остановки записи.

Важным этапом в разработке приложения было тестирование дизайна, которое подтвердило его корректность и успешное функционирование. Это гарантирует, что разработанное приложение соответствует поставленным целям и требованиям.

В целом разработка мобильной версии приложения для звукозаписи позволила создать функциональное и удобное приложение, учитывающее потребности пользователей. Прделанная работа успешно достигла поставленной цели курсового проекта и является основой для дальнейшего развития и улучшения приложения.

Библиография

Нормативно-правовые акты:

1 ГОСТ Р 2.105-2019. ЕСКД. Общие требования к текстовым документам. – Москва: Стандартинформ, 2019. – 36 с

Электронные ресурсы:

1 Android Developers [Электронный ресурс]. – Документация Android Studio. – URL: <https://developer.android.com/docs/> (дата обращения: 11.03.2024)

2 Firebase Documents [Электронный ресурс]. – Документация Firebase. Вытягивание данных с базы данных – URL: <https://firebase.google.com/support> (дата обращения: 11.03.2024)

3 Figma [Электронный ресурс]. – Прототипирование проектов. – URL: <https://www.figma.com/> (дата обращения: 1.03.2024)

4 Stackoverflow [Электронный ресурс]. – Полная документация по Android. – URL: <https://ru.stackoverflow.com/> (дата обращения: 1.03.2024)

5 GitHub [Электронный ресурс]. – Платформа для хостинга и совместной разработки программного обеспечения. – URL: <https://github.com/> (дата обращения: 1.03.2024)

Приложение А

(обязательное)

Министерство образования Новосибирской области
ГБПОУ НСО «Новосибирский авиационный технический колледж
имени Б.С. Галуцака»

РАЗРАБОТКА МОБИЛЬНОГО ПРИЛОЖЕНИЯ ДЛЯ ЗАПИСИ ЗВУКОВ

Техническое задание
НАТКиГ.201500.010.000ПЗ

Выполнил:
Студент группы ПР-21.101
Мысин А.К.

2024

					НАТКиГ.201500.010.000ПЗ	Лист
Изм.	Лист	№ докум	Подпись	Дата		27

Содержание

Введение.....	29
1 Назначения разработки.....	30
2 Требования к мобильному приложению	30
2.1 Требования к функциональным характеристикам	30
2.2 Требования к надёжности	30
2.3 Условия эксплуатации	30
2.4 Требования к составу и параметрам технических средств.....	31
2.5 Требования к информационной и программной совместимости.....	31
2.6 Требования к защите информации	31
2.7 Требования к маркировке и упаковке	31
3 Требования к программной документации	31
4 Техничко-экономические показатели	32
5 Стадии и этапы разработки	32
6 Порядок контроля и приёмки.....	33

Введение

Настоящее техническое задание распространяется на разработку мобильного приложения «Разработка мобильного приложения для звукозаписи звуков(диктофон)», используемого для звукозаписи, и проигрывания записанных аудиофайлов.

Наименование приложения: «Recorder».

Краткая характеристика области применения: мобильное приложение предоставляет возможность просмотра списка и воспроизведения аудио, а также записи аудиофайлов.

Основанием для проведения разработки является Протокол №6 от 21 февраля 2022 года.

Наименование темы разработки - «Разработка мобильного приложения для записи звуков».

Условное обозначение темы разработки – «Recorder».

1 Назначение разработки

Основное назначение приложения заключается в:

- обеспечении удобного и понятного интерфейса приложения для пользователя;
- предоставление возможности создания и прослушивания аудиофайлов.

Лица, которые могут работать с данной системой:

- администратор – управляет полностью всем приложением, следит за его работоспособностью и могут выполнять техническую поддержку;
- пользователь приложения – может просматривать список аудиозаписей, а также прослушивать их.

2 Требования к мобильному приложению

2.1 Требования к функциональным характеристикам

Требования к составу выполняемых функций:

- Создание аудиофайла;
- Возможность переименовать аудиофайл;
- Возможность прослушать аудиофайл;

2.2 Требования к надёжности

Обеспечение устойчивого функционирования должно выполняться несколькими действиями:

- корректной работы интерфейса и инструментов
- обработки ошибок и исключений для предотвращения сбоев приложения.

2.3 Условия эксплуатации

Пользователь должен иметь практические навыки использования мобильного устройства под управлением операционной системы Android.

2.4 Требования к составу и параметрам технических средств

Для работы приложения необходимо мобильное устройство с установленной операционной системой Android не ниже версии 10.0.

2.5 Требования к информационной и программной совместимости

Проектирование взаимодействия с файловой системой должно быть выполнено в рамках разработки курсового проекта. При разработке взаимодействия с файловой системой должен быть использован язык программирования Java.

2.6 Требования к защите информации

Доступ к сохранённым файлам имеет только пользователь.

2.7 Требования к маркировке и упаковке

Требования к маркировке и упаковке не предъявляются.

3 Требования к программной документации

Состав программной документации должен включать в себя:

- техническое задание;
- пояснительная записка.

4 Технико-экономические показатели

Экономические преимущества разработки и ориентировочная экономическая эффективность не рассчитывается.

5 Стадии и этапы разработки

№	Этапы разработки КП	Сроки выполнения	Отчётность
1	Определение цели и задач, объекта и предмета исследования	24.02.2024	Пояснительная записка
2	Описание предметной области	25.02.2024	Пояснительная записка
3	Выбор технологии, языка и среды программирования	27.02.2024	Пояснительная записка
4	Оформление технического задания	29.02.2024	Техническое задание
5	Проектирование UI/UX дизайна	02.03.2024	Спецификации программного обеспечения
6	Разработка мобильного приложения	07.03.2024	Схема структурная системы и спецификации компонентов
7	Разработка базы данных	16.03.2024	Программный продукт
8	Отладка и тестирование приложения	18.03.2024	Тексты программных компонентов
9	Оформление документации	20.03.2024	Программная документация
10	Защита	06.05.2024	

6 Порядок контроля и приёмки

Виды испытаний – защита курсового проекта.

Общие требования к приёмке:

- техническое задание;
- пояснительная записка;
- программный продукт;

					<i>НАТКиГ.201500.010.000ПЗ</i>	<i>Лист</i>
<i>Изм.</i>	<i>Лист</i>	<i>№ докум</i>	<i>Подпись</i>	<i>Дата</i>		33