

Разработка блокнота в Android Studio

Андрюенко Иван Сергеевич

Приамурский государственный университет имени Шолом-Алейхема

Студент

Аннотация

В данной статье представлено пошаговое руководство по разработке блокнота в Android Studio. Описывается процесс создания пользовательского интерфейса, работа с базой данных SQLite и реализация основной функциональности, такой как добавление, редактирование и удаление заметок. Читатели ознакомятся с основными концепциями разработки Android-приложений и получат необходимые навыки для создания собственного блокнота. Конечный результат - полнофункциональное приложение блокнота, готовое к развертыванию на устройствах Android.

Ключевые слова: Android, блокнот, мобильное приложение, Android Studio, пользовательский интерфейс, база данных, SQLite.

Notepad development in Android Studio

Andrienko Ivan Sergeevich

Sholom-Aleichem Priamursky State University

Student

Abstract

This article provides a step-by-step guide to the development of notepad in Android Studio. Describes the process of creating a user interface, working with the SQLite database and implementing basic functionality such as adding, editing and deleting notes. Readers will get acquainted with the basic concepts of Android application development and get the necessary skills to create their own notebook. The end result is a fully functional notepad application ready to be deployed on Android devices.

Keywords: Android, Notepad, Mobile application, Android Studio, User interface, database, SQLite.

1 Введение

1.1 Актуальность

Блокноты являются неотъемлемой частью повседневной жизни пользователей, предоставляя удобный способ организации и хранения заметок. Создание собственного блокнотного приложения в Android Studio позволяет расширить навыки в разработке мобильных приложений и овладеть фундаментальными принципами разработки на платформе Android. Это включает изучение компонентов Android, работу с пользовательским интерфейсом и базами данных. Также разработка блокнотного приложения

поможет понять принципы управления данными и создание привлекательного пользовательского интерфейса. Эти навыки могут быть применены в различных сферах, где организация информации является важной.

1.2 Обзор исследований

В своей работе А.Р. Нафикова рассмотрела пример разработки простейшего мобильного приложения для работы с базой данных SQLite в среде Android Studio [1]. Р.Р. Асмандияров описал автоматизированный процесс поиска и вывода расписания при автономной работе без сети Интернет с использованием среды разработки Android Studio, являющейся официальной средой под Android, работающей под операционными системами Windows, Linux, MacOSX [2]. В своей работе Люманов О.Э., Усеинов Э.А. провели анализ актуальности разработки мобильных приложений, их значения в жизни человека. Описано как мобильные приложения помогают решать различные задачи. Также рассмотрен функционал мобильного приложения «Auto Parts» и обоснована его эффективность при подборе запчастей [3]. В своей работе В.Е. Анасьев, О.И. Чуйко представили этап создания мобильного приложения для автоматизации сметных расчетов строительных организаций и предварительно оценки стоимости строительства объектов недвижимости [4]. С.В. Козлов, А.И. Юхимчук рассмотрели платформу XAMARIN, как фреймворк, позволяющий разрабатывать мобильные приложения с использованием языка программирования C#. Авторами было разработано приложение «Календарь репетитора» для Android на языке C# в среде разработки MS Visual Studio 2019 Enterprise with XAMARIN на базе .NET Framework 4.5 [5].

1.3 Цель исследования

Цель исследования – разработать приложение блокнота в Android Studio, предназначенное для создания и управления заметками.

2 Материалы и методы

Процесс создания мобильного приложения проделан в среде Android Studio, с использованием языка программирования Java.

3 Результаты и обсуждения

На первом этапе процесса создания блокнота в Android Studio требуется создание нового проекта. Для этого выбирается шаблон Empty Activity, который предоставляет базовую структуру проекта с одной активностью. Также необходимо указать невысокую версию SDK, чтобы обеспечить совместимость с широким спектром устройств. В качестве языка программирования будет использоваться java.

Одним из ключевых аспектов разработки блокнота в Android Studio является создание удобного и интуитивно понятного пользовательского интерфейса. В процессе разработки используются различные компоненты пользовательского интерфейса, такие как текстовые поля, кнопки и списки,

чтобы пользователи могли создавать, просматривать и редактировать свои заметки. Размети главную активность: добавим место для заметок, поле для ввода текста и кнопку «добавить» (рис. 1).

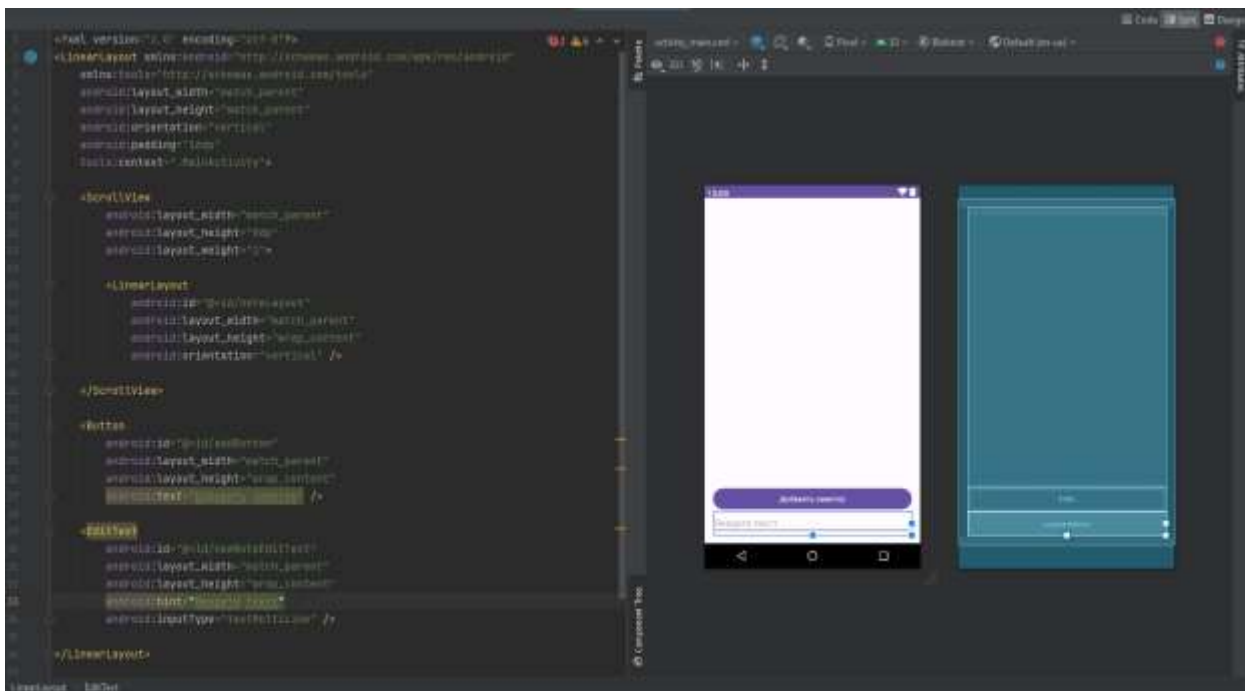


Рисунок 1 – Создание главного макета

Далее создадим макет, на котором будут находиться кнопки «удалить» и «редактировать», когда заметка будет создана. Необходимо создать новый макет в папке «layout» и прописать кнопки, а также место для текста (рис. 2).

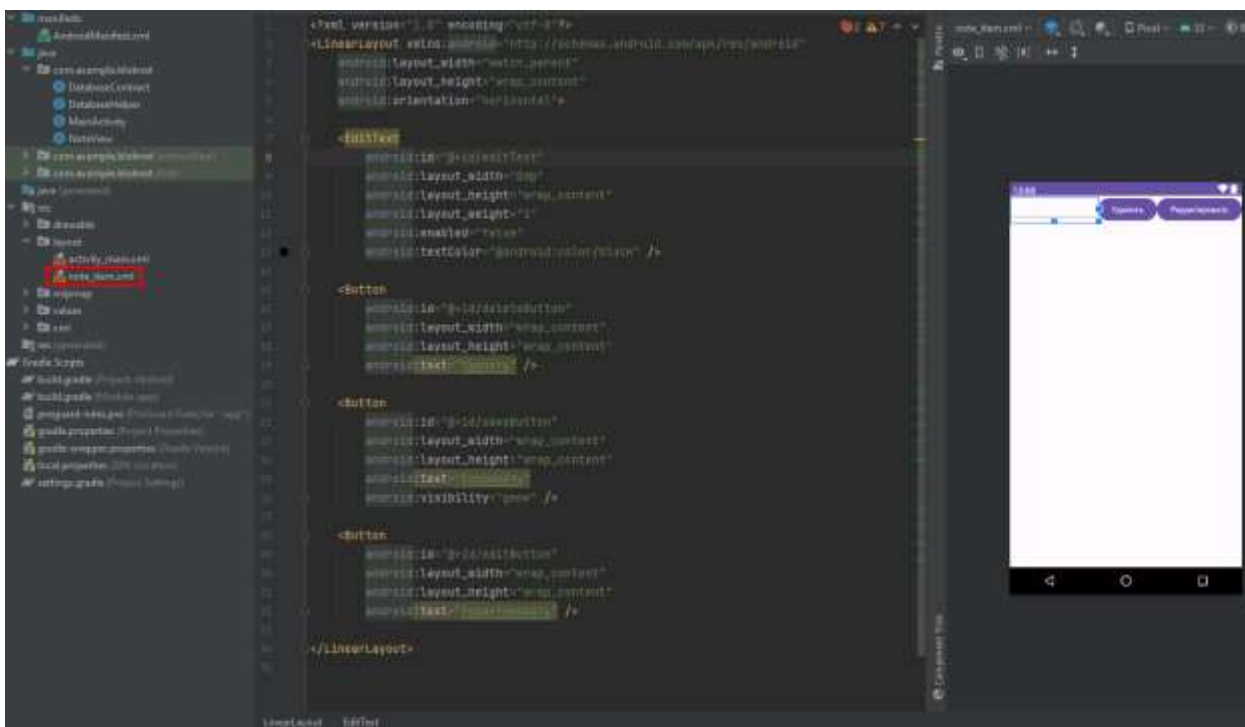


Рисунок 2 – Создание дополнительного макета для заметок

Для сохранения заметок и задач используется подходящий механизм хранения данных. В данном случае, выбрана база данных SQLite, встроенный инструмент для работы с данными в Android. Создается таблица с определенными полями, предназначенными для хранения текста заметок. Этот выбор обеспечивает эффективное и надежное хранение данных приложения, что позволяет пользователям безопасно хранить и управлять своими заметками и задачами.

Создадим два класса DatabaseContract.java и DatabaseHelper.java для удобного управления базами данных. Класс DatabaseContract.java создается для определения структуры базы данных и предоставления констант, которые используются при работе с базой данных SQLite в приложении. Этот класс содержит описание базы данных, а именно названия столбцов и название таблицы (рис. 3).

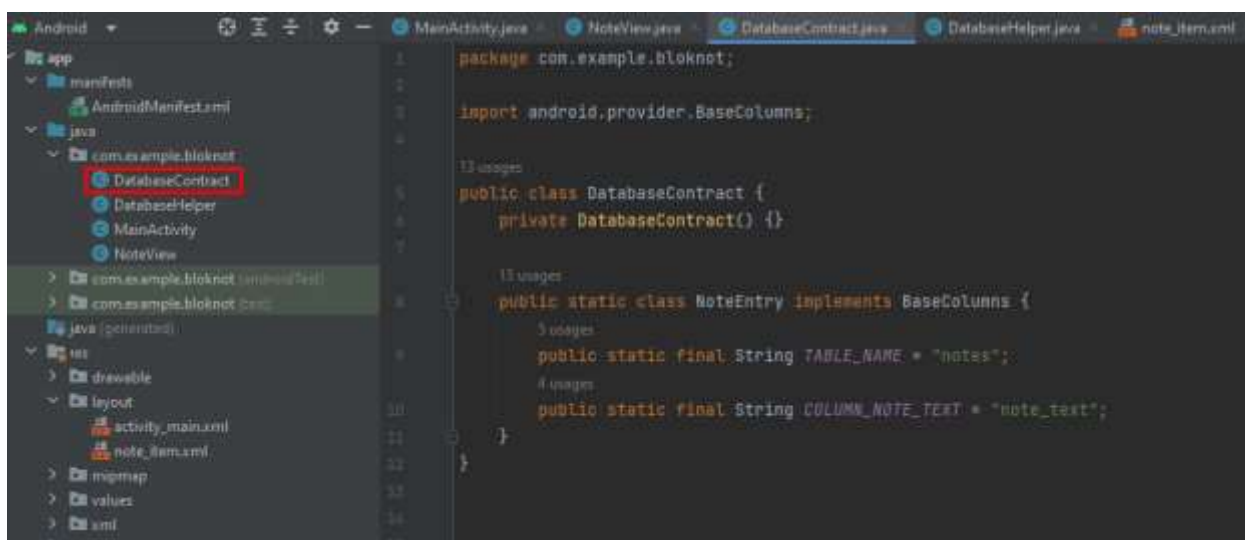


Рисунок 3 – Создание класса «DatabaseContract.java»

Класс DatabaseHelper создан для управления базой данных в приложении блокнота. Он наследуется от класса SQLiteOpenHelper, который является встроенным классом в Android для создания и обновления базы данных SQLite. Класс DatabaseHelper выполняет функции создание и обновления базы данных. В методе onCreate() определяется структура базы данных. В данном случае, создается таблица "notes" с двумя столбцами: "_ID" и "note_text". Столбец "_ID" является первичным ключом, а "note_text" предназначен для хранения текста заметки. Метод db.execSQL() используется для выполнения SQL-запроса, который создает указанную таблицу. Обновление базы данных происходит в методе onUpgrade() (рис. 4).

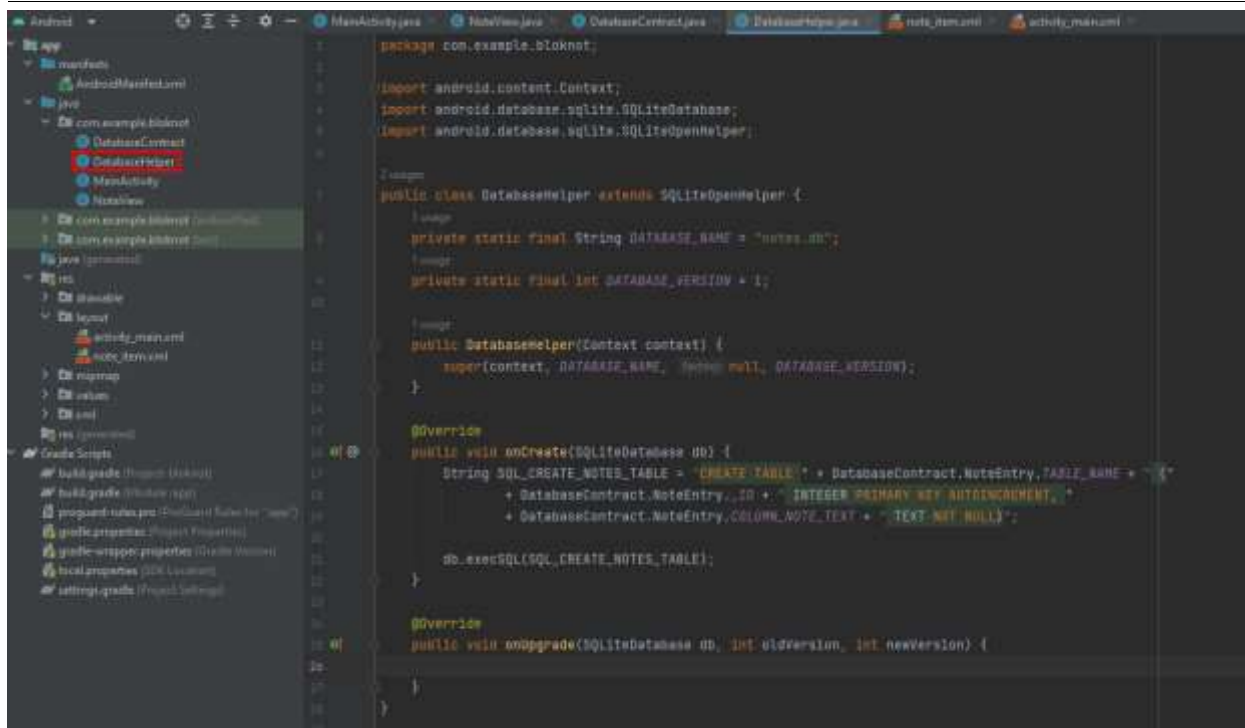


Рисунок 4 – Создание класса «DatabaseHelper.java»

Далее создадим класс `NoteView.java`. Он наследуется от `LinearLayout` и содержит различные компоненты пользовательского интерфейса, такие как `EditText` (текстовое поле) для ввода текста заметки, а также кнопки для удаления, сохранения и редактирования заметки. У класса есть флаг `isEditMode`, который указывает, находится ли заметка в режиме редактирования. При нажатии кнопки "Редактировать" (`editButton`), заметка переключается в режим редактирования, и `EditText` становится доступным для ввода текста. Также в классе определены обработчики событий для кнопок "Удалить", "Сохранить" и "Редактировать". При нажатии каждой из этих кнопок выполняются соответствующие действия. В методах `deleteNote()`, `saveNote()` и `editNote()` класса происходит взаимодействие с базой данных через методы. Это позволяет обновлять, сохранять и удалять заметки в базе данных на основе пользовательских действий (рис. 5, 6, 7).


```
1 usage
private void setEditMode() {
    editText.setEnabled(isEditMode);
    editText.setTextColor(Color.BLACK);
    saveButton.setVisibility(isEditMode ? VISIBLE : GONE);
    editButton.setVisibility(isEditMode ? GONE : VISIBLE);
}

1 usage
private void deleteNote() {
    ((MainActivity) getContext()).deleteNoteFromDatabase(noteId);
    ((MainActivity) getContext()).loadSavedNotes();
}

1 usage
private void saveNote() {
    String noteText = editText.getText().toString();
    ((MainActivity) getContext()).updateNoteInDatabase(noteId, noteText);

    isEditMode = false;
    setEditMode();
}

1 usage
private void editNote() {
    isEditMode = true;
    setEditMode();
}
}
```

Рисунок 7 – Создание класса «NoteView.java»

Пропишем код для главной активности в приложении блокнота. Он будет отвечать за управление основным пользовательским интерфейсом приложения и взаимодействие с базой данных для добавления, удаления и обновления заметок. Метод `loadSavedNotes()` выполняет запрос к базе данных для получения всех сохраненных заметок. Полученные заметки отображаются в виде объектов, которые добавляются для отображения пользователю. Также создан метод, который выполняется, когда пользователь нажимает кнопку "Добавить". Он берет текст, который был введен в поле. После добавления заметки, поле ввода очищается, чтобы пользователь мог ввести новую заметку, и вызывается функция `loadSavedNotes()`, чтобы обновить список заметок на экране. Методы `deleteNoteFromDatabase()` и `updateNoteInDatabase()` вызываются, когда пользователь хочет удалить или изменить существующую заметку. Они выполняют операции удаления или обновления записей в базе данных на основе идентификаторов заметок, которые передаются в качестве аргументов (рис. 8, 9, 10).


```
public void addNewNote() {
    String newNoteText = newNoteEditText.getText().toString().trim();
    if (!newNoteText.isEmpty()) {
        ContentValues values = new ContentValues();
        values.put(DatabaseContract.NoteEntry.COLUMN_NOTE_TEXT, newNoteText);

        database.insert(DatabaseContract.NoteEntry.TABLE_NAME, null, values);

        newNoteEditText.setText("");
        showNewNotes();
    }
}

public void deleteNoteFromDatabase(int noteId) {
    String selection = DatabaseContract.NoteEntry._ID + " = ?";
    String[] selectionArgs = {String.valueOf(noteId)};

    database.delete(DatabaseContract.NoteEntry.TABLE_NAME, selection, selectionArgs);
}

public void updateNoteInDatabase(int noteId, String newNoteText) {
    String selection = DatabaseContract.NoteEntry._ID + " = ?";
    String[] selectionArgs = {String.valueOf(noteId)};

    ContentValues values = new ContentValues();
    values.put(DatabaseContract.NoteEntry.COLUMN_NOTE_TEXT, newNoteText);

    database.update(DatabaseContract.NoteEntry.TABLE_NAME, values, selection, selectionArgs);
}
```

Рисунок 10 – Создание класса «MainActivity.java»

Тестируем приложение (рис. 11, 12).

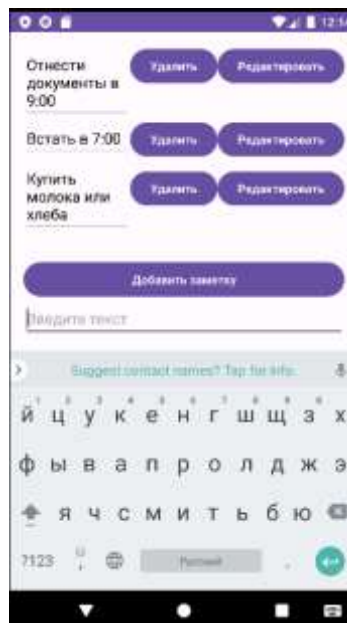


Рисунок 11 – Добавление заметки



Рисунок 12 – Редактирование и сохранение заметки

Выводы

В данной статье был представлен процесс разработки блокнота в среде разработки Android Studio. В результате работы было создано приложение, позволяющее пользователям создавать, просматривать, редактировать и удалять заметки и задачи. Была достигнута цель создания функционального приложения для работы с заметками и задачами на платформе Android. В дальнейшем возможно расширение функциональности и улучшение дизайна приложения для обеспечения лучшего пользовательского опыта.

Библиографический список

1. Нафикова А.Р. Технология работы с базами данных в среде android studio // Математическое моделирование процессов и систем. Материалы XI Международной молодежной научно-практической конференции. Отв. редактор С.В. Викторов. Стерлитамак, 2021. С. 92-97.
2. Асмандияров Р.Р. Мобильное приложение для вывода расписания и поиска пригородного автобуса // Актуальные проблемы науки и техники. Материалы I Межвузовской научно-технической конференции с международным участием, посвященной 75-летию Победы в Великой Отечественной войне и 100-летию начала производства авиационной техники в городе Сарапуле. 2020. С. 160-162.
3. Люманов О.Э., Усеинов Э.А. Разработка мобильного приложения для автоматизации процесса подбора запчастей для ремонта автомобилей // Информационно-компьютерные технологии в экономике, образовании и социальной сфере. 2018. № 1 (19). С. 95-98.
4. Анасьев В.Е., Чуйко О.И. Разработка мобильного приложения для просмотра и автоматизации сметных расчетов в строительных организациях // Современные тенденции и проекты развития

информационных систем и технологий. материалы Межрегиональной научно-исследовательской конференции студентов и школьников. Хабаровск, 2022. С. 25-29.

5. Козлов С.В., Юхимчук А.И. Особенности разработки еженедельника средствами визуальных программных сред // Актуальные проблемы науки и техники. Материалы II Международной научно-технической конференции, посвященной 70-летию ИМИ - ИжГТУ и 60-летию СПИ (филиал) ФГБОУ ВО "ИжГТУ имени М.Т. Калашникова". Ижевск, 2022. С. 413-417.