**Оглавление**

[I ВВЕДЕНИЕ 2](#__RefHeading___Toc2938_2419586467)

[II ОСНОВНАЯ ЧАСТЬ 5](#__RefHeading___Toc3045_2419586467)

[1. Исследовательская часть 5](#__RefHeading___Toc3349_2419586467)

[2. Проектная часть 8](#__RefHeading___Toc3351_2419586467)

[3. Технологическая часть 10](#__RefHeading___Toc3353_2419586467)

[III Вывод 22](#__RefHeading___Toc3355_2419586467)

[IV Список используемой литературы 22](#__RefHeading___Toc3357_2419586467)

# **I ВВЕДЕНИЕ**

* 1. **Цель**

Цель разработки приложения для ведения заметок и задач заключается в предоставлении пользователю возможность записывать нужную информацию и организовывать задачи.

* 1. **Задачи**

Для достижения поставленной цели необходимо решить следующие задачи:

1) Анализ пользовательских требований:

* Выявить функционал, который будет эффективно полезным для пользователя;
* Разработать архитектуру приложения.

2) Разработка UI:

* Разработать удобный интерфейс, в котором будет легко ориентироваться.

3) Фактический функционал:

* Реализовать все пользовательские и системные функции.

4) Тестирование:

* Провести тестирование для выявления и устранения ошибок.

Приложение для ведения заметок и задач предназначено для персонального использования. Оно помогает пользователям эффективно организовывать свои мысли, идеи и задачи, обеспечивая удобное управление ими. Основная цель приложения — упростить процесс фиксации важной информации, систематизации дел и отслеживания их выполнения.

Пользователи смогут создавать текстовые заметки, добавлять задачи и экспортировать или импортировать заметки и задачи.

Для выполнения данного проекта проводился анализ программного обеспечения, реализация приложения и его тестирование. Для получения информационной и теоретической базы использовались учебные материалы, документации и научные статьи.

* 1. **Объект исследования.**

Объектом исследования в данной курсовой работе является мобильное приложение заметок, предназначенное для ведения полезных записей и задач с возможностью быстрой записи заметок.

* 1. **Предметная область исследования.**

Предметная область исследования предполагает разработку программного комплекса для создания заметок и задач, что включает в себя необходимое изучение современных технологий, методов разработки ПО, проектирование пользовательского интерфейса и внедрение баз данных.

Проектирование пользовательского интерфейса включает с себя разработку интуитивного визуала.

Внедрение баз данных позволяет хранить созданную пользователем информацию. Для этого можно использовать такие решения, как SQLite, Room или Firebase.

Все остальные функциональные возможности непосредственно предоставляют пользователю доступ к внесению информации в базу данных.

* 1. **Актуальность.**

Современный ритм жизни требует эффективного управления информацией, задачами и личными идеями. Ежедневно человек сталкивается с большим количеством дел, мыслей и планов, которые важно фиксировать, чтобы не упускать важные детали и повышать продуктивность. Приложение для ведения заметок и задач становится незаменимым инструментом, помогающим структурировать информацию и контролировать выполнение поставленных целей.

Актуальность разработки такого приложения обусловлена следующими факторами:

1. Рост потребности в организации времени – в условиях высокой загруженности люди ищут удобные способы планирования и фиксации задач.

2. Популярность мобильных устройств – смартфоны и планшеты стали основными инструментами для быстрого доступа к информации, что делает цифровые заметки более удобными, чем бумажные.

# **II ОСНОВНАЯ ЧАСТЬ**

# **1. Исследовательская часть**

**1.1 Описание предметной области.**

Приложение для ведения заметок и задач предназначено для помощи пользователям в организации личной и рабочей информации. Оно позволяет:

* Создавать и редактировать текстовые заметки.
* Формировать списки задач с приоритетами и сроками выполнения.

Такие приложения востребованы среди студентов, профессионалов и всех, кто стремится к эффективному тайм-менеджменту.

**1.2 **Анализ существующих аналогов****

Существующие решения либо узкоспециализированы, либо перегружены функциями. Оптимальным будет создание простого, но функционального приложения с удобным интерфейсом.

**Популярные аналоги:**

* Google Keep – простые заметки с цветовыми метками, но без сложного управления задачами.
* Microsoft To-Do – удобное планирование дел, но ограниченные возможности для заметок.
* Notion – мощный инструмент с гибкой структурой, но сложный для новичков.
* Evernote – богатый функционал для заметок, но платная подписка для полного доступа.

**1.3 Выбор инструментальных средств моделирования**

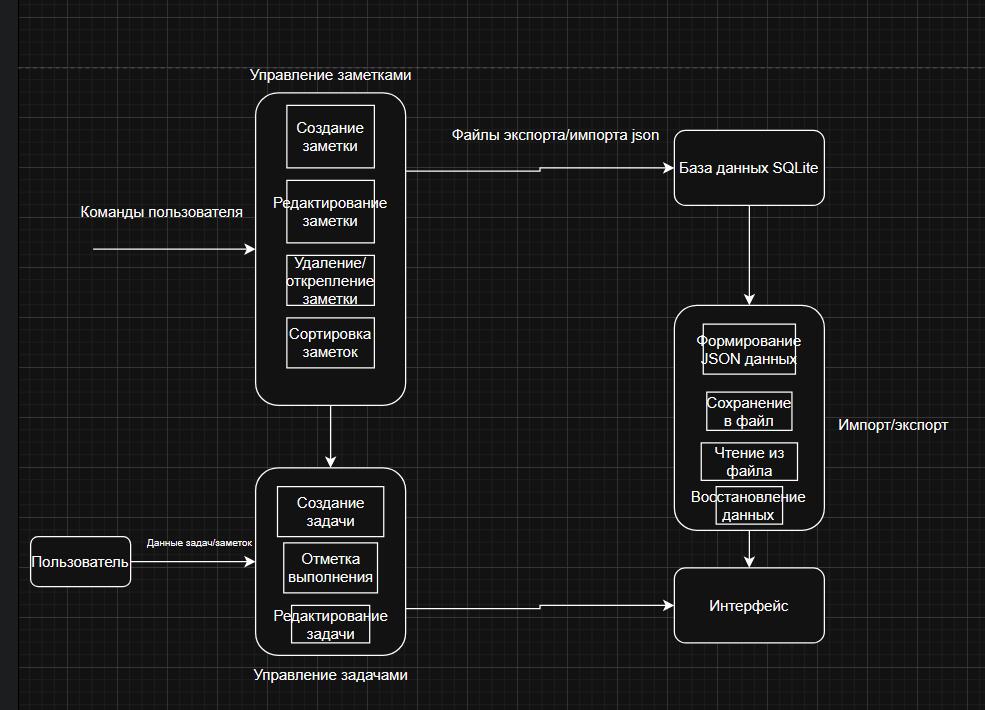
Для проектирования системы используются:

* IDEF0 – для описания бизнес-процессов и функциональных блоков.
* DFD – для визуализации потоков данных.

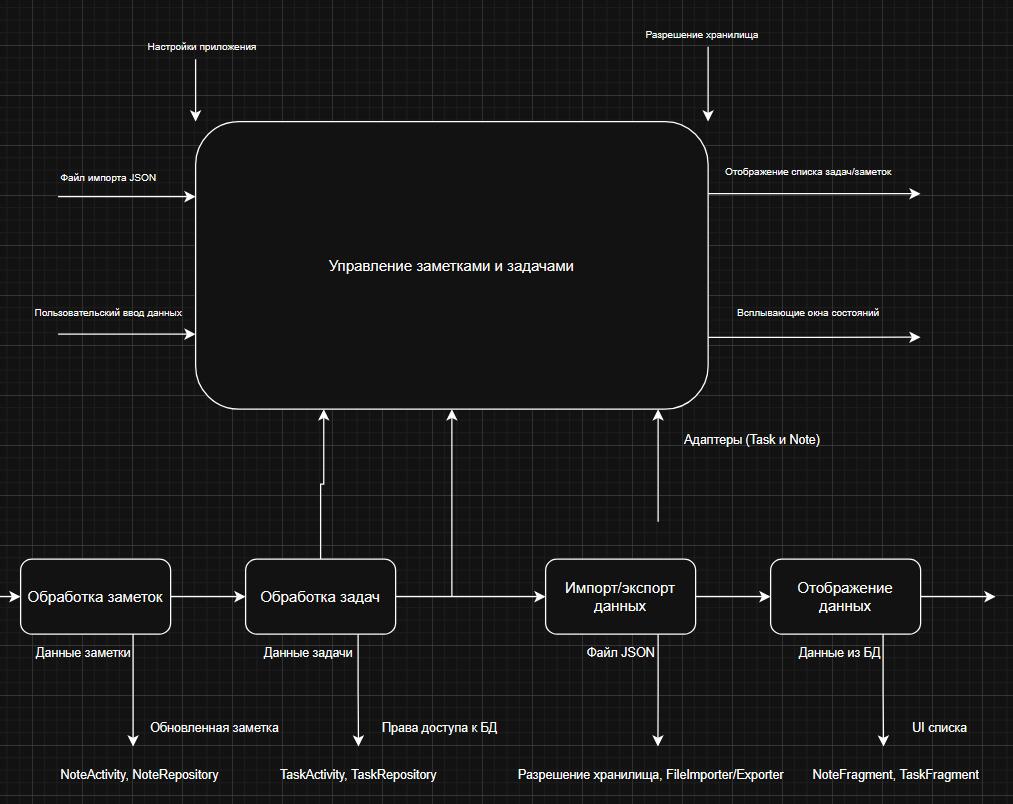
В качестве средств разработки используются:

* Языки программирования: Java и язык разметки XML.
* Среда разработки Android Studio.
* База данных: SQLite.

**1.4 Описание процесса в нотации IDEF0**

Рисунок 1 — Описание процесса в нотации IDEF0.

**1.5 Описание процесса в нотации DFD**

Рисунок 2 — Описание процесса в нотации DFD

# **2. Проектная часть**

**2.1. Определение пользовательских требований**

Приложение должно предоставлять пользователям следующие возможности:

Создание и редактирование заметок:

* Текстовый ввод с сохранением.
* Добавление заголовков.

Управление задачами:

* Создание задач с указанием названия, описания, срока выполнения.

Простота интерфейса:

* Минималистичный дизайн.

**2.2. Определение функциональных требований**

Основные функции системы:

1. Работа с заметками:

* CRUD (создание, чтение, обновление, удаление).

2. Управление задачами:

* Отметка о выполнении.
* Перенос сроков.

3. Резервное копирование:

* Экспорт данных в файл (JSON).

Нефункциональные требования:

4. Производительность:

* Быстрая загрузка заметок (до 1 сек).
* Поддержка оффлайн режима

**2.3. Описание используемых средств разработки**

Фронтентд:

* Мобильная разработка: Java (Android).
* XML разметка.

Бэкэнд:

* База данных: SQLite (локальная БД).

**2.4 Архитектура программной системы**

Выбранная архитектура: Model-View-ViewModel (MVVM).

Модель (Model):

* Локальная База Данных (SQLite).
* Классы данных (Note и Task).

Представление (View):

* Экраны активности (Activity).
* XML разметка.

ViewModel:

* Связь между Model и View.

Проектная часть включает: детальные требования пользователей и системы, стек технологий для реализации, четкую архитектуру для поддержки масштабируемости.

# **3. Технологическая часть**

**3.1 Описание моделей и структур данных.**

**Основные сущности:**

**1. Сущность заметки (Note):**

* **TABLE\_NOTES: String – название таблицы заметок.**
* **COLUMN\_NOTE\_ID: String – уникальный идентификатор заметки.**
* **COLUMN\_NOTE\_TITLE: String – заголовок заметки.**
* **COLUMN\_NOTE\_CONTENT: String – содержание заметки.**
* **COLUMN\_NOTE\_CREATED\_AT: String – дата и время создания заметки.**
* **COLUMN\_PINNED: String – статус закрепления заметки.**
* **COLUMN\_PINNED\_DATE: String – дата и время закрепления заметки**

**2. Сущность задачи (Task):**

* **TABLE\_TASKS: String – название таблицы задач.**
* **COLUMN\_TASK\_ID: String – уникальный идентификатор задачи.**
* **COLUMN\_TASK\_TITLE: String – заголовок задачи.**
* **COLUMN\_TASK\_DESCRIPTION: String – описание задачи.**
* **COLUMN\_TASK\_COMPLETED: String – завершенность задачи.**
* **COLUMN\_TASK\_CREATED\_AT: String – дата и время создания задачи.**
* **COLUMN\_TASK\_DUE\_DATE: String – дата и время выполнения задачи.**

**3.2 Описание клиентской части.**

**Мобильное приложение (Java).**

**1. Стек:**

* **Язык: Java.**
* **Локальная БД: SQLite.**

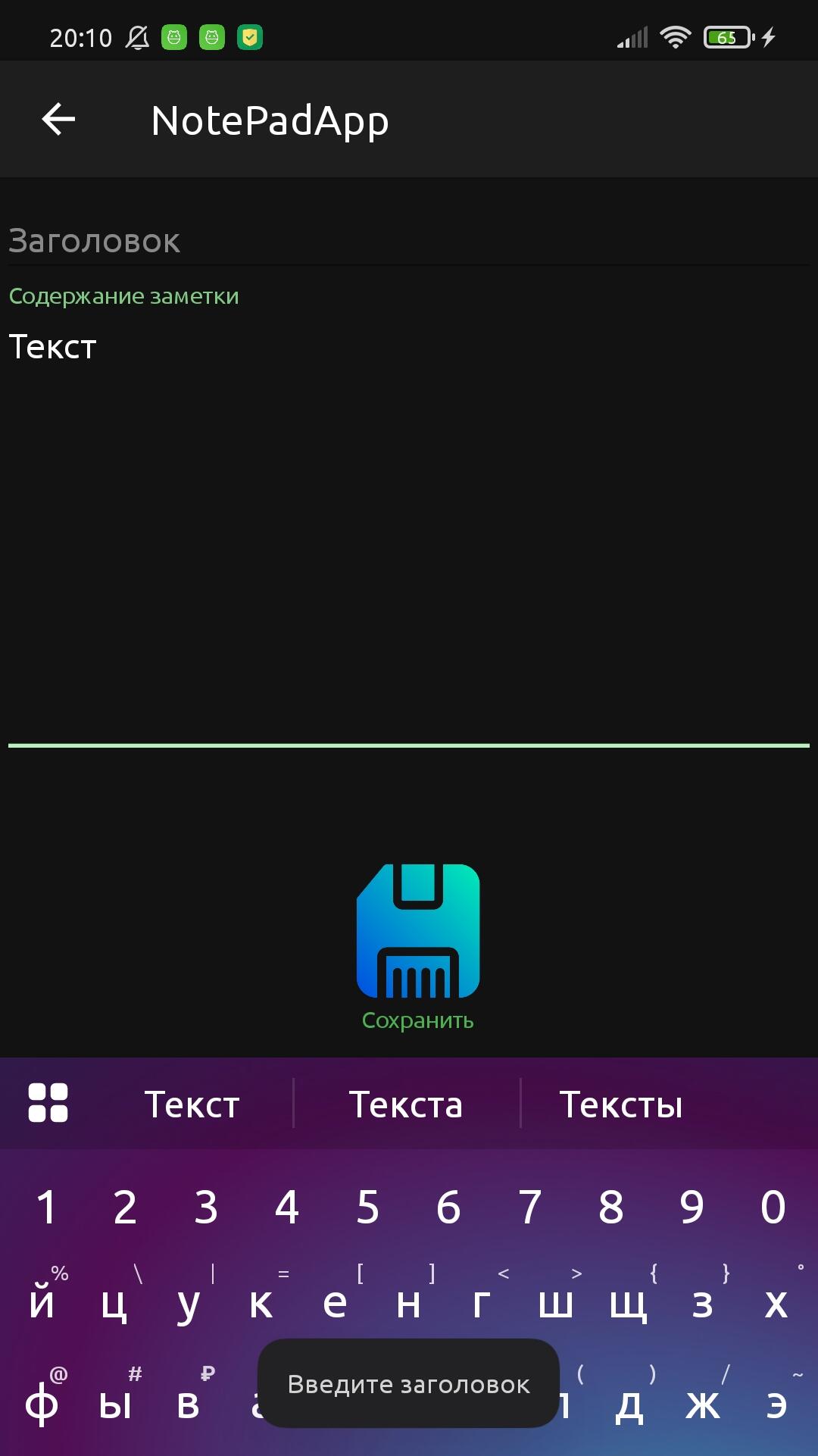
**2. Экраны:**

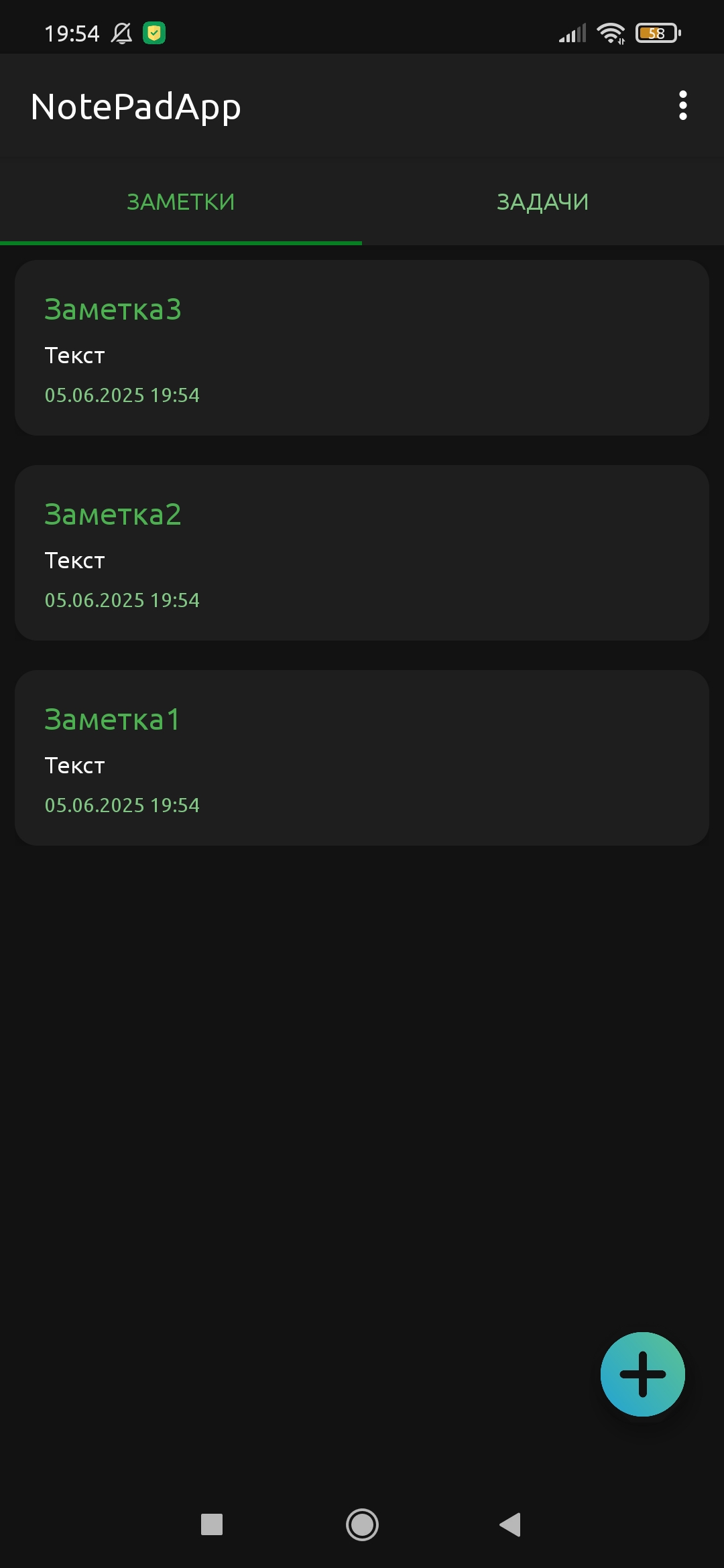
* **Список заметок и задач: отображение в виде карточек, кнопка добавления задачи/заметки.**
* **Редактор заметок и задач: экран с полями ввода и кнопкой сохранить.**
* **Настройки с кнопками экспорт/импорт данных.**

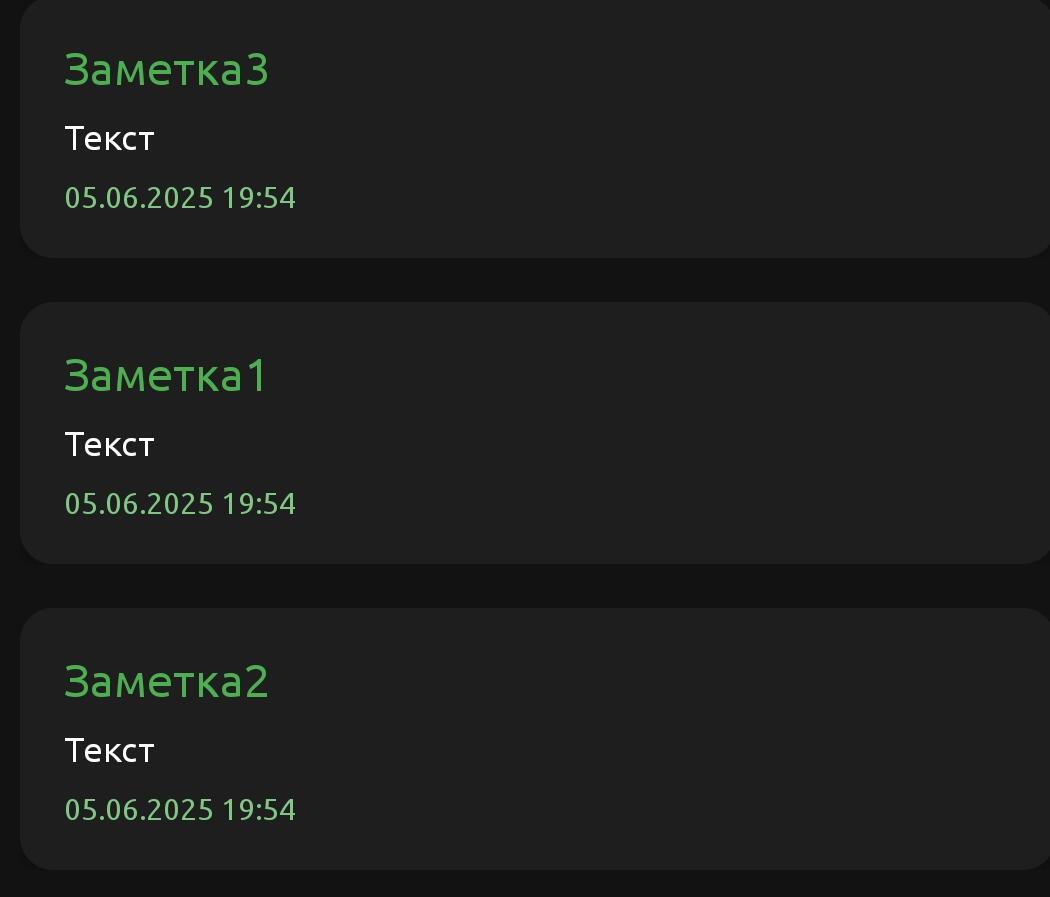
**3.3 Тестирование и верификация прототипа.**

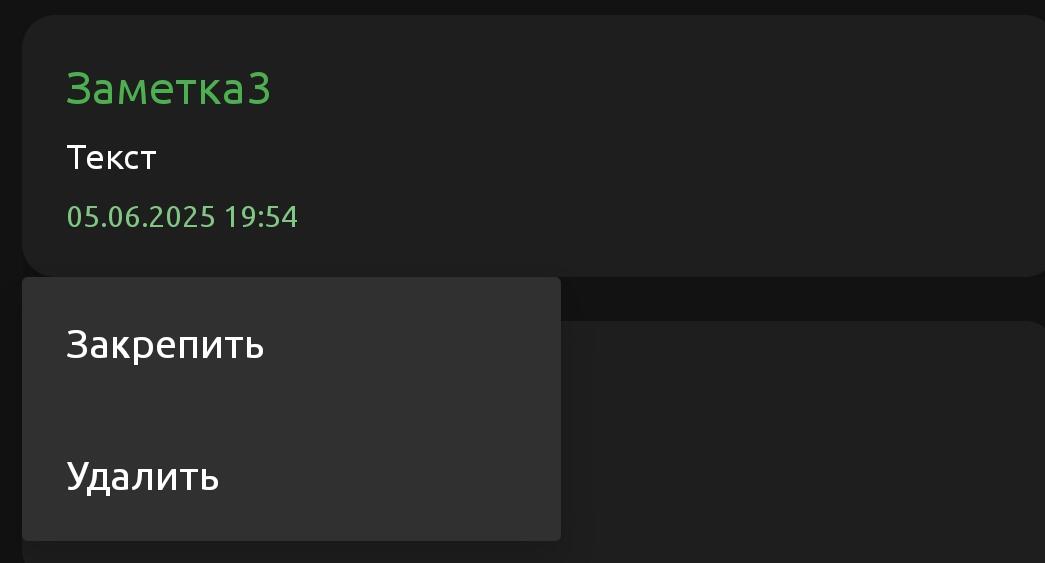
**Проверим готовую программу на работу.**

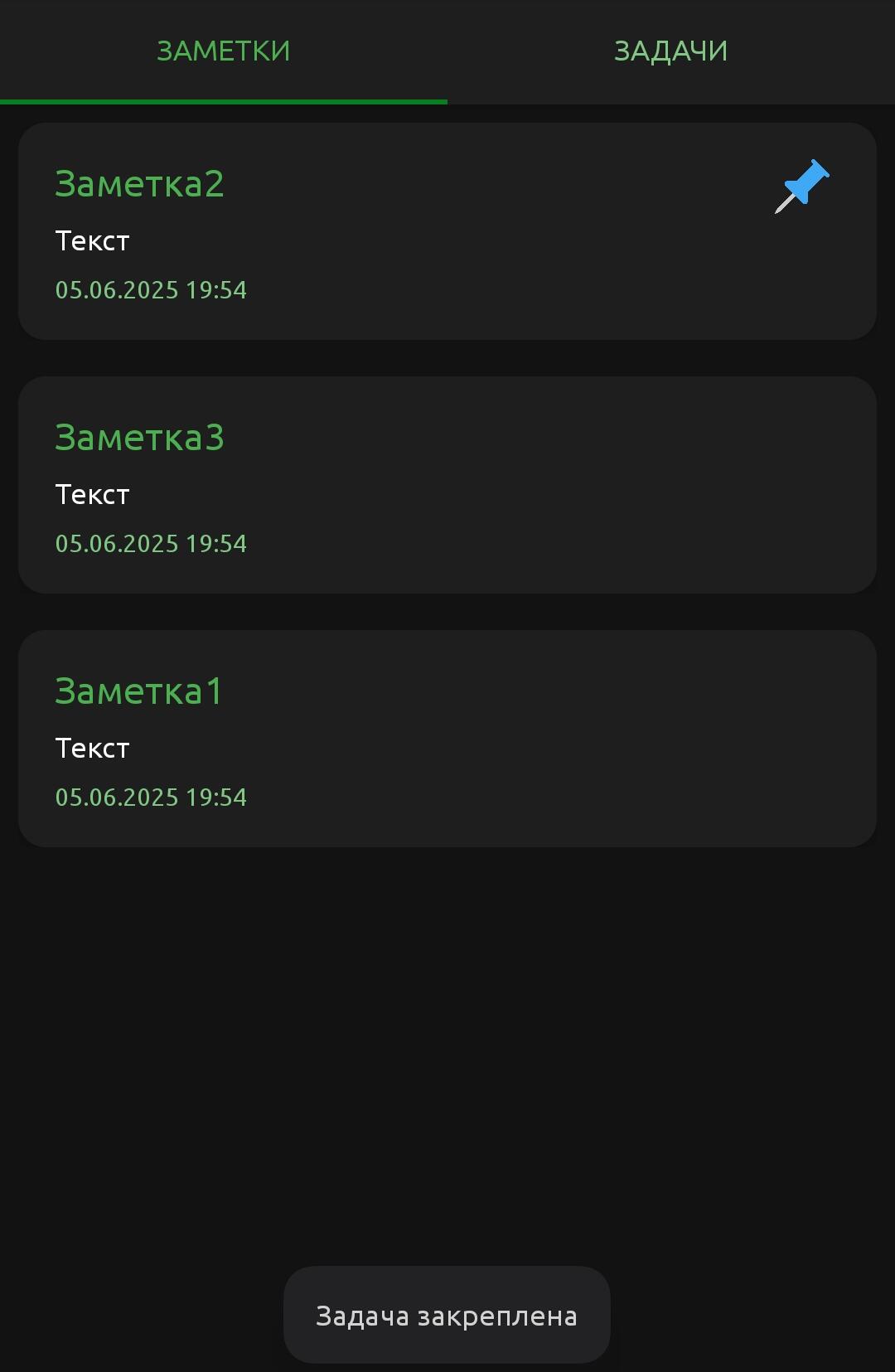
Рисунок 3 — Пустой экран заметок с кнопкой добавления

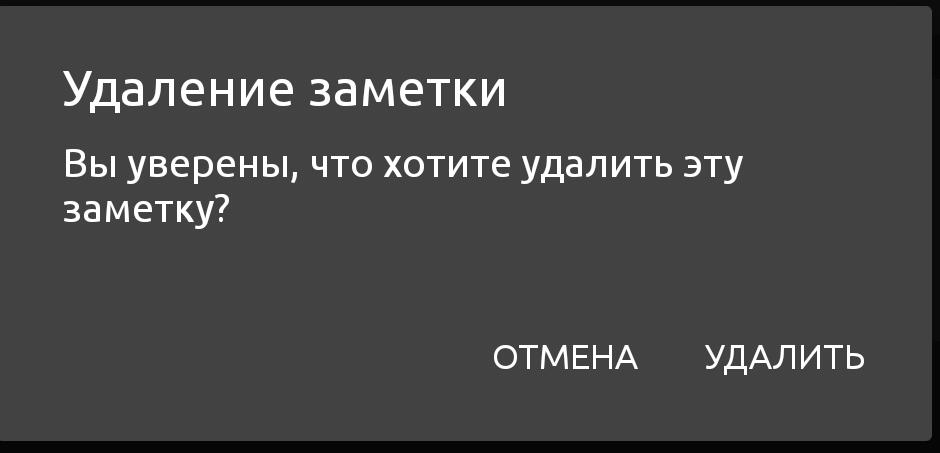
Рисунок 4 — При попытке сохранить заметку без заголовка приложение не даст этого сделать

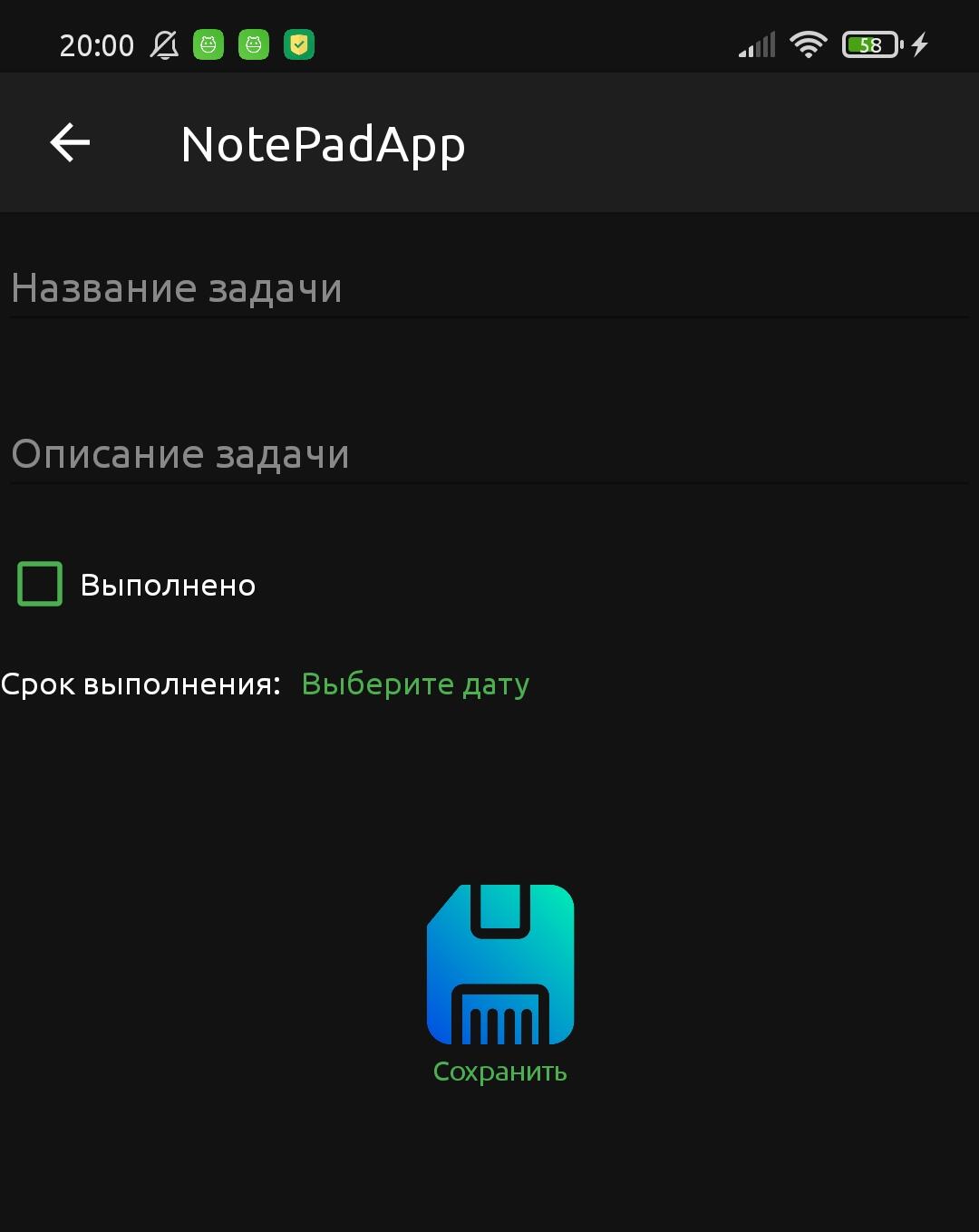
Рисунок 5 — Добавим несколько заметок

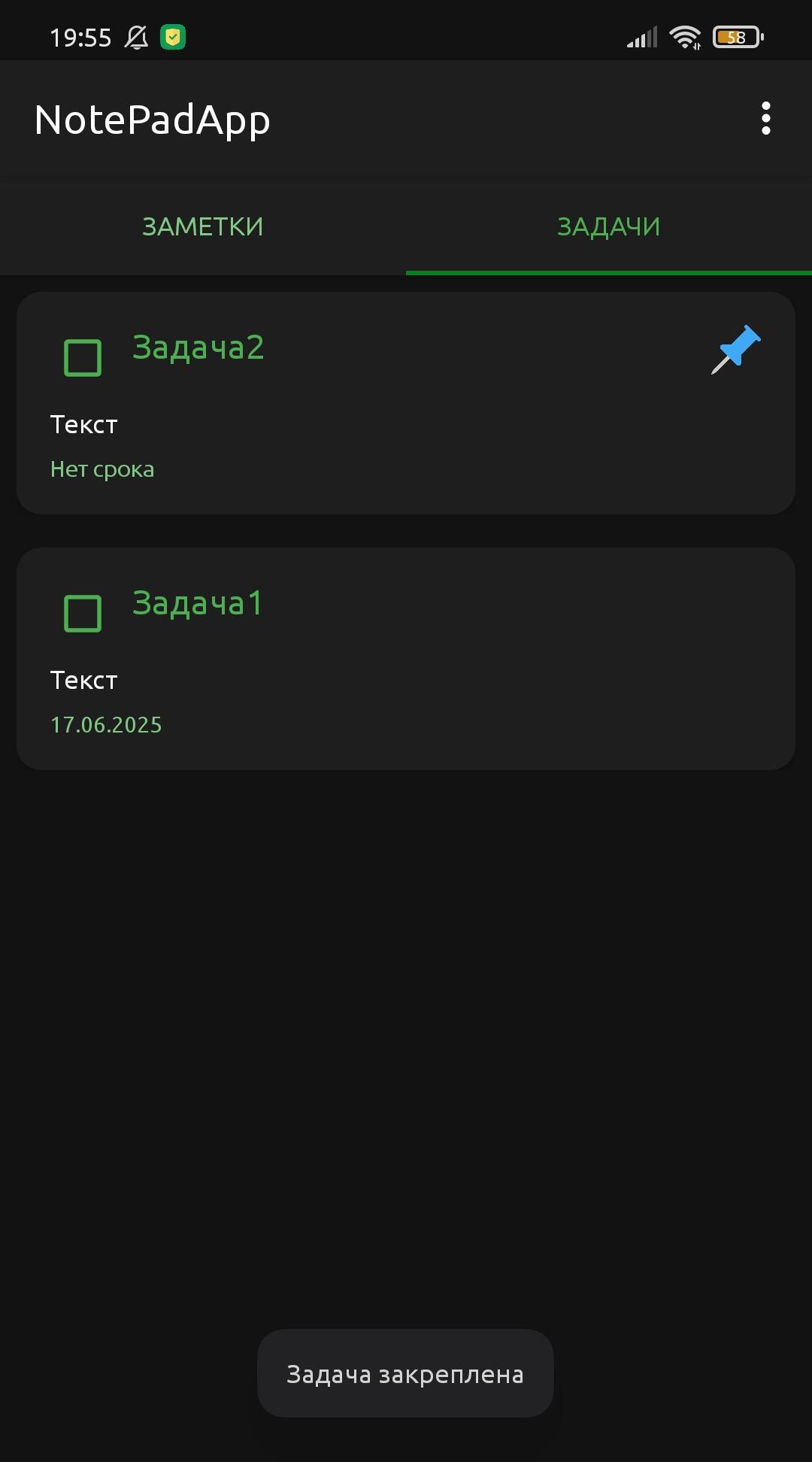
Рисунок 6 — Проверяем возможность перемещать заметки (и задачи также)

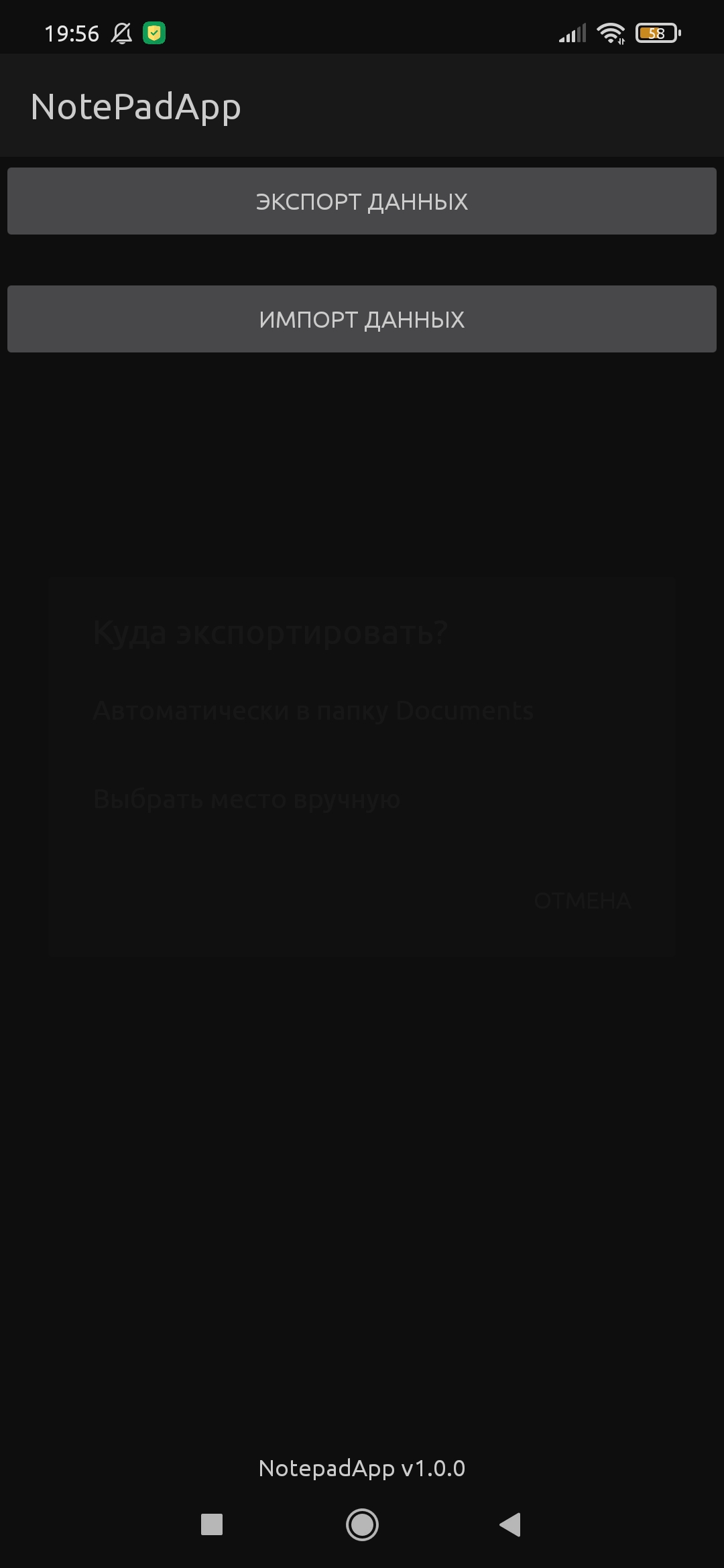
Рисунок 7 — Проверяем контекстное меню на зажатие

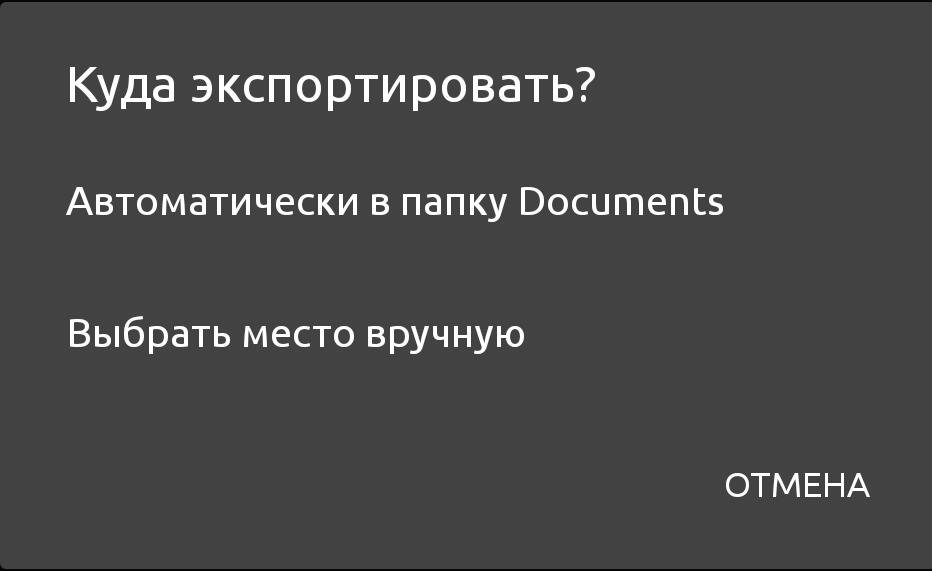
Рисунок 8 — Проверяем возможность закреплять задачи и заметки

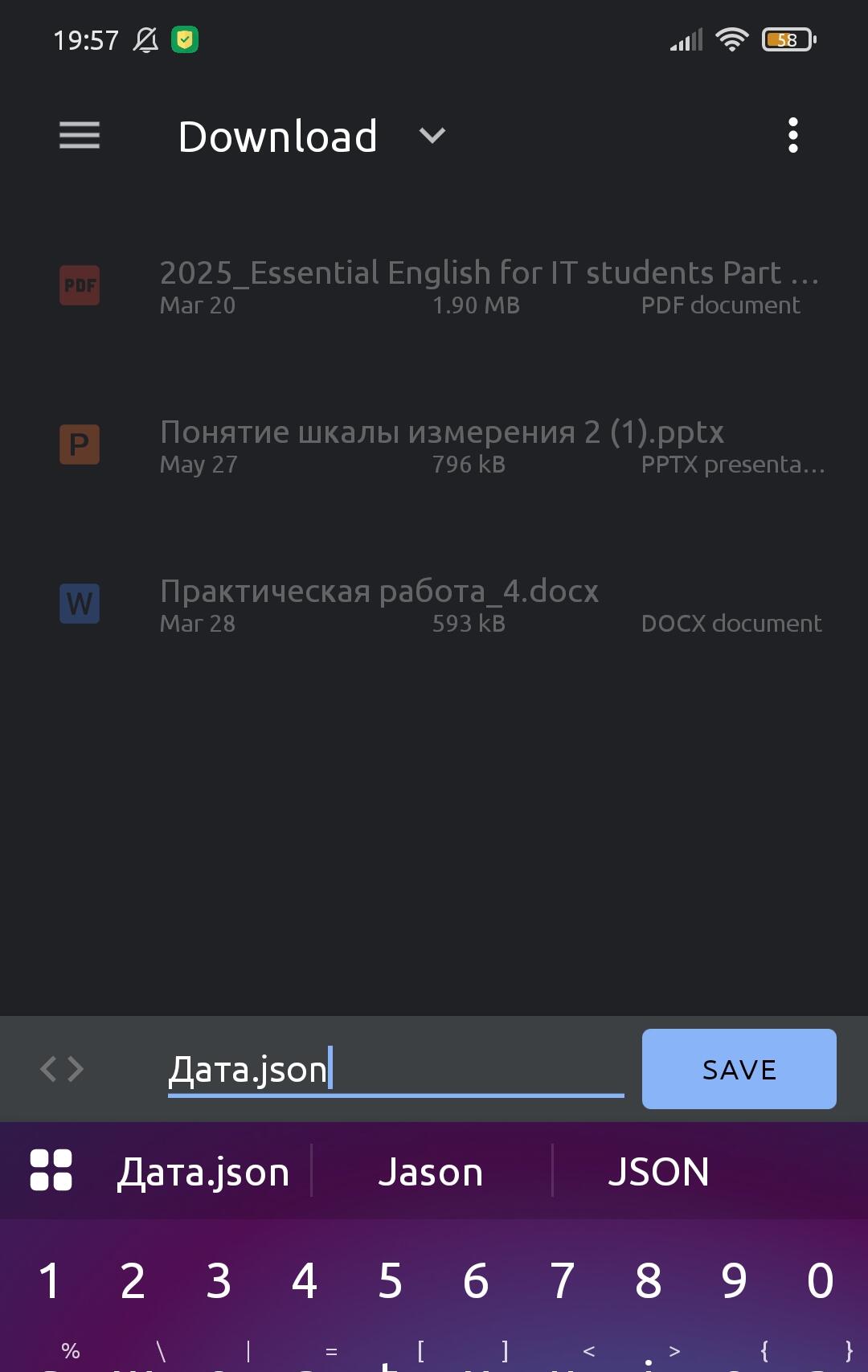
Рисунок 9 — Проверяем удаление заметки/задачи

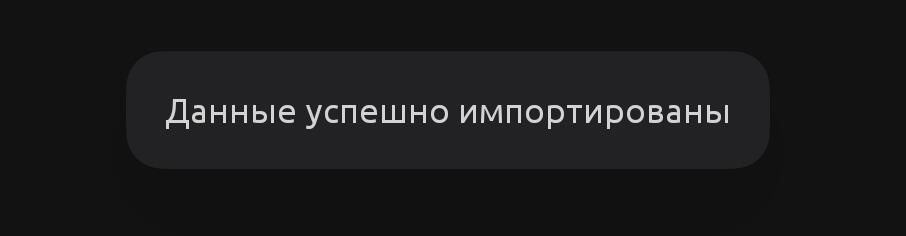
Рисунок 10 — Экран с добавлением задачи

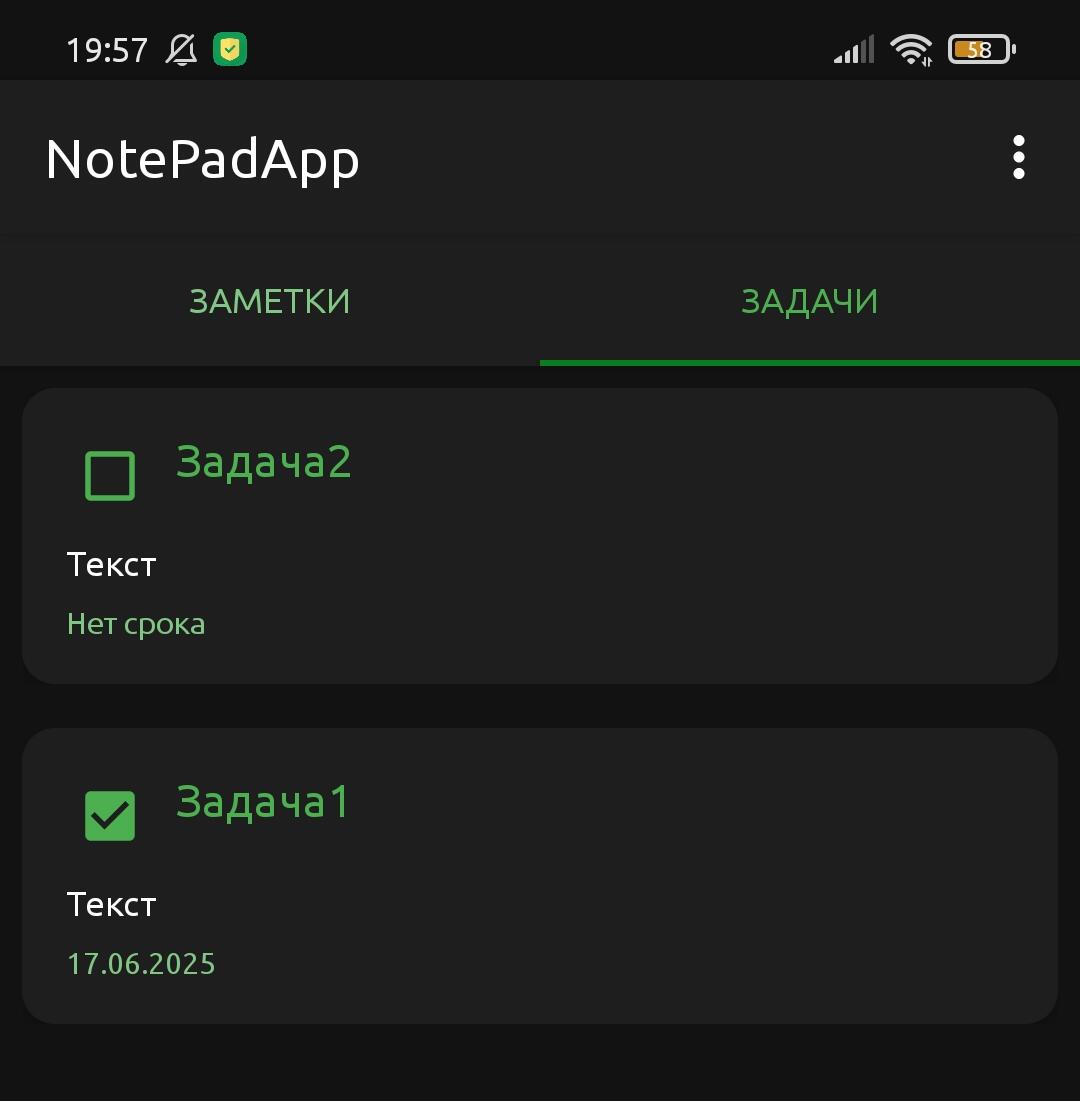
Рисунок 11 — Проверяем добавленные задачи и закрепляем одну для примера

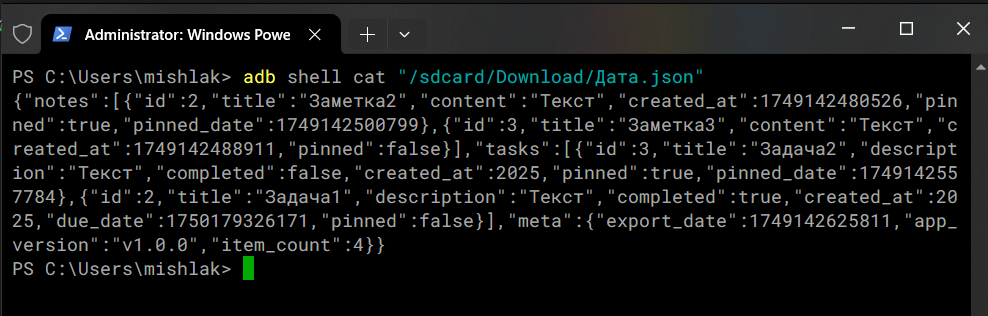
Рисунок 12 — Проверяем экран с настройками, где мы можем сделать экспорт или импорт данных

Рисунок 13 — Для теста создадим бэкап в папкую, выбранную вручную

Рисунок 14 — Выбираем папку Download и сохраняем туда файл Дата

Рисунок 15 — Импортируем json по кнопке, предварительно удалив задачи

Рисунок 16 — Восстановленные задачи после импорта данных

Рисунок 17 — Проверяем содержимое json

# **III Вывод**

В ходе работы было разработано приложение для ведения заметок и управления задачами, которое предоставляет пользователям удобный инструмент для организации личной и рабочей информации. Проект включал в себя исследование предметной области, анализ существующих аналогов, проектирование архитектуры системы, а также детальную проработку технологической части.

Данное приложение может быть полезно широкому кругу пользователей — от студентов до профессионалов, которым необходимо эффективно организовывать свои задачи и мысли. В перспективе возможны доработки, такие как добавление веб-версии, расширенных инструментов форматирования или интеграции с внешними сервисами.

Таким образом, проект достиг поставленных целей, предложив современное и удобное решение для управления заметками и задачами.

Ссылка на Github: [репозиторий](https://github.com/mihailkl/mobile-razrabotka_semestr4/tree/main/NotePadApp)

# **IV Список используемой литературы**

1. Чернов Е.А. Лекционные материалы по дисциплине «Разработка

мобильных приложений», 2025.

2. Android Developers [Электронный ресурс]. URL:

https://developer.android.com/ (дата обращения 05.06.2025)