МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ УЧРЕЖДЕНИЕ

КАЛИНИНГРАДСКОЙ ОБЛАСТИ

ПРОФЕССИОНАЛЬНАЯ ОБРАЗОВАТЕЛЬНАЯ ОРГАНИЗАЦИЯ

«КОЛЛЕДЖ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И СТРОИТЕЛЬСТВА»

(ГБУ КО ПОО «КИТиС»)

|  |  |  |
| --- | --- | --- |
| Курсовой проект  допущен к защите  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (дата)  Зам.директора по УМР  \_\_\_\_\_\_\_\_\_\_\_\_\_Павленко Г.Я.  (подпись) |  | Курсовой проект  защищен с оценкой  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (оценка)  Руководитель работы  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись, дата) |

Пояснительная записка к курсовому проекту

по дисциплине: МДК 01.01 Разработка программных модулей

Тема: Разработка мобильного приложения под Android «Записная книжка с функцией будильника»

Специальность 09.02.07 «Информационные системы и программирование»

|  |  |
| --- | --- |
| Выполнила: | студентка 3 курса,  группы ИСп 19-2к  Руденкова Анна Александровна  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись) |
| Руководитель: | Большакова-Стрекалова Анна Викторовна, преподаватель ГБУ КО ПОО КИТиС  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (подпись) |

Калининград

2022

Содержание

[Введение 3](#_Toc102080863)

[1 Описание предметной области 4](#_Toc102080864)

[1.1 Аналоги разрабатываемого приложения 4](#_Toc102080865)

[1.2 Техническое задание 7](#_Toc102080866)

[1.2.1 Общие сведения 7](#_Toc102080867)

[1.2.2 Назначение и цели создания мобильного приложения 7](#_Toc102080868)

[1.2.3 Характеристика объекта автоматизации 8](#_Toc102080869)

[1.2.4 Требования к мобильному приложению 8](#_Toc102080870)

[1.2.4.1 Требования к информационному обеспечению 8](#_Toc102080871)

[1.2.4.2 Требования к техническому обеспечению 8](#_Toc102080872)

[1.2.4.3 Требования к программному обеспечению 8](#_Toc102080873)

[1.2.5 Состав и содержание работ по созданию мобильного приложения 9](#_Toc102080874)

[1.2.6 Порядок контроля и приёмки мобильного приложения 9](#_Toc102080875)

[1.2.7 Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу мобильного приложения в действие 9](#_Toc102080876)

[1.2.8 Требования к документированию 10](#_Toc102080877)

[1.3 Описание структуры приложения 10](#_Toc102080878)

[2 Описание разработки приложения 12](#_Toc102080879)

[2.1 Обоснование средств разработки 12](#_Toc102080880)

[2.2 Разработка интерфейса 14](#_Toc102080881)

[2.3 Схема взаимодействия компонентов проекта 17](#_Toc102080882)

[2.4 Описание разработанных классов 18](#_Toc102080883)

[3 Тестирование и установка приложения 21](#_Toc102080884)

[Заключение 30](#_Toc102080885)

[Список использованной литературы 31](#_Toc102080886)

[Приложение А Код программы 32](#_Toc102080887)

Введение

Согласно статистике, более половины населения планеты пользуются мобильными приложениями. Рынок мобильных приложений очень быстро растёт. В России более 90 миллионов человек уже пользуются услугами мобильных приложений. Прибыли компаний, которые занимаются производством и разрабатывают мобильное ПО измеряется миллионами и миллиардами рублей. Каждый новый продукт, который предоставляет компания, служит как для сохранения существующей аудитории, так и для привлечения новых клиентов. Разработчики внедряют новые возможности и сервисы, улучшают старые технологии.

Смартфоны давно перестали быть чем-то необычным, сейчас это устройство есть почти у каждого. Они также, как и более знакомые всем телефоны отлично справляются со своей функцией – коммуникацией. Но за последнее время функционал смартфонов становится всё более разнообразным. Использование возможностей мобильных устройств для решения широкого круга задач в различных областях деятельности человека сейчас в приоритете для развития разных отраслей.

Тема работы: «Разработка мобильного приложения для Android «Записная книжка с функцией будильника».

Для выполнения поставленной нам цели необходимо выполнить ряд задач:

1. Исследовать предметную область.
2. Проанализировать аналогичные приложения.
3. Выбрать средства проектирования и разработки.
4. Разработать схему взаимодействия компонентов проекта.
5. Сделать описание интерфейса приложения.
6. Разработать и протестировать мобильное приложение.

Методы исследования: анализ действующих популярных приложений, выбор более оптимального интерфейса, изучение литературы.

# 1 Описание предметной области

## 1.1 Аналоги разрабатываемого приложения

Рассмотрим существующие на рынке мобильные приложения.

Мобильное приложение FairNote. Разработчики FairNote делают акцент на безопасности заметок. Есть возможность зашифровать выбранные записи и защитить их паролем. Платная версия программы позволяет шифровать сразу все файлы одним нажатием и использовать вместо пароля сканер отпечатков пальцев.

Интерфейс программы представлен на рисунке 1.1.



Рисунок 1 - Интерфейс программы FairNote

Рассмотрим особенности приложения:

1. Приложение поддерживает следующие типы заметок: текстовые записи, чек-листы, напоминания.
2. Приложение поддерживает каталогизацию по метке.
3. Приложение не поддерживает синхронизацию между устройствами, только сохранение заметок в Dropbox или Google Drive.
4. Приложение не поддерживает доступ через веб или компьютер.

Мобильное приложение Omni Notes. Ключевая особенность Omni Notes - продвинутая поддержка вложений. Есть возможность прикреплять к заметкам видео, снимки, аудиозаписи, гиперссылки и даже собственные рисунки. Кроме того, стоит отметить возможность объединения нескольких заметок в одну. Программа полностью бесплатна.

Интерфейс программы представлен на рисунке 2.

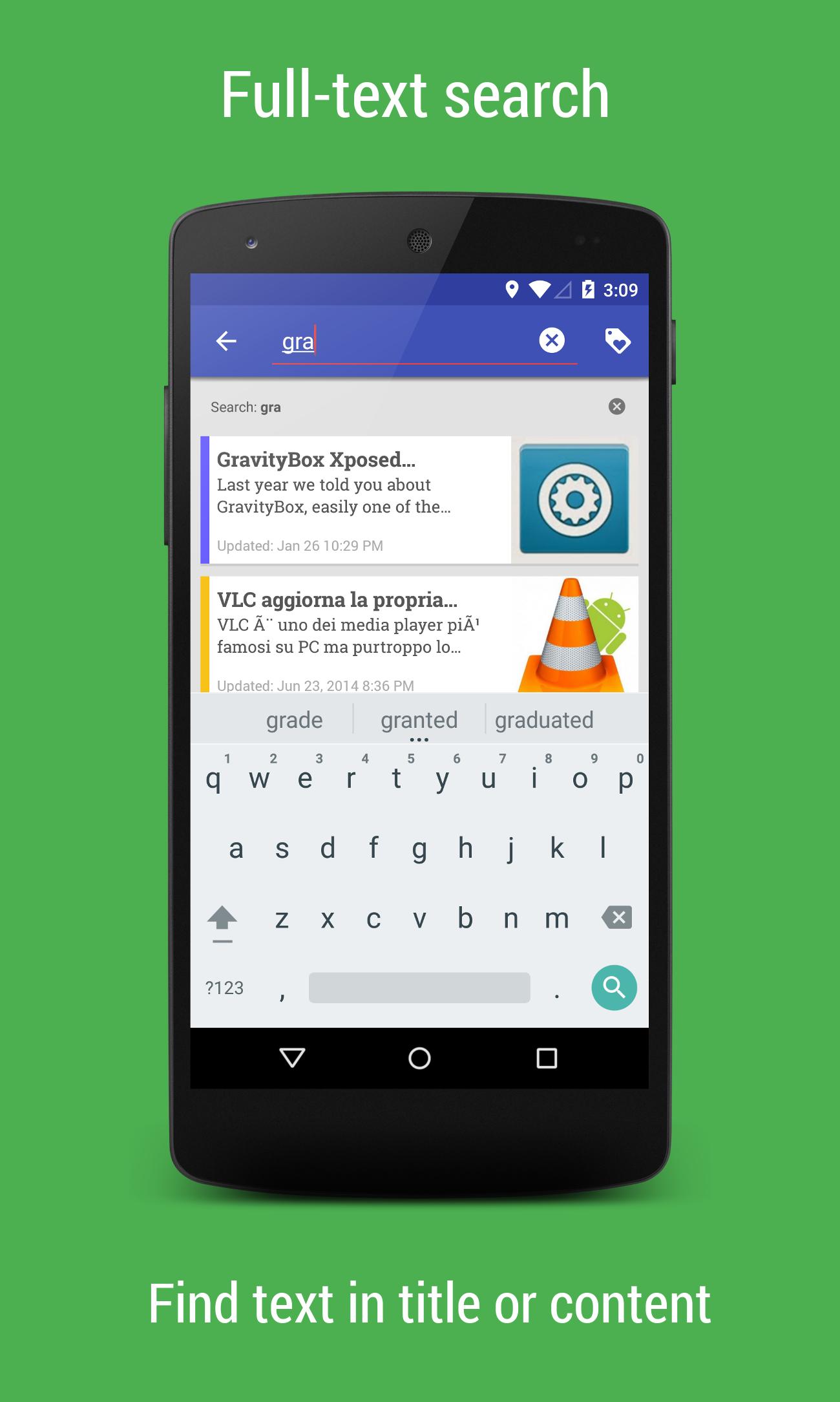


Рисунок 2 - Интерфейс программы Omni Notes

Рассмотрим особенности приложения:

1. Приложение поддерживает следующие типы заметок: текстовые записи с вложениями, изображения, чек-листы и напоминания.
2. Приложение поддерживает каталогизацию по категории и метке.
3. Приложение не поддерживает синхронизацию между устройствами, только сохранение заметок в локальную память.
4. Приложение не поддерживает доступ через веб или компьютер.

Мобильное приложение SomNote. Это приложение может понравиться любителям классических папок и файлов. Заметки можно защитить от посторонних, поставив на приложение ПИН-код. Помимо прочего, SomNote позволяет прикреплять к записям снимки и скетчи. Оформив платную подписку, можно избавиться от рекламы и получить множество тем оформления, 30 ГБ в облаке и возможность защищать только выбранные папки.

Интерфейс программы представлен на рисунке 3.

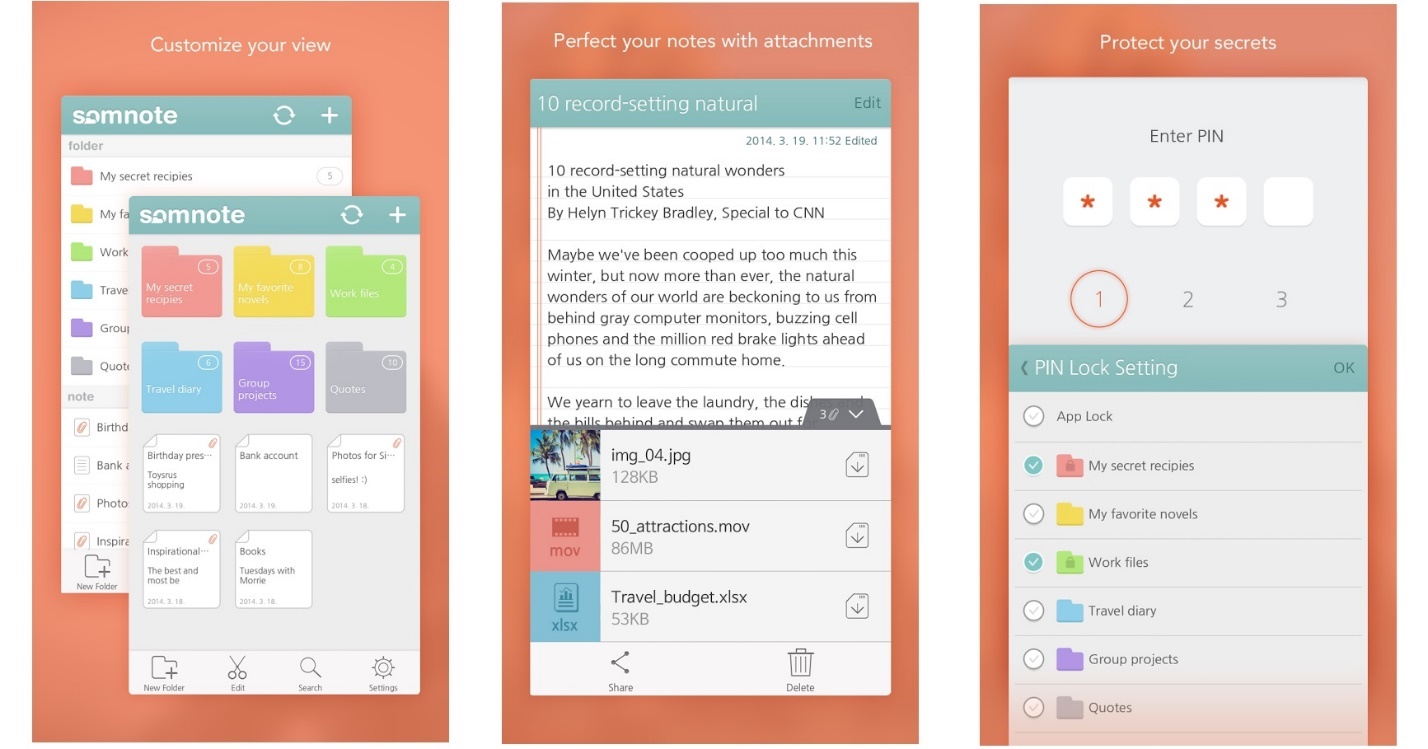


Рисунок 3 - Интерфейс программы SomNote

Рассмотрим особенности приложения:

1. Приложение поддерживает следующие типы заметок: текстовые записи с вложениями и скетчи.
2. Приложение поддерживает каталогизацию по папкам.
3. Приложение поддерживает синхронизацию между устройствами.
4. Приложение поддерживает доступ через веб или компьютер.
5. Приложение не поддерживает систему напоминаний.

Рассмотренные программные средства будут использоваться как шаблоны и примеры функционала, так как содержат в себе много интересных задумок. Тот же функционал можно легко реализовать в собственной разработке, и в долгосрочной перспективе это решение может оказаться более выгодным. Решение создать мобильное приложение собственными силами, которое бы отвечало всем запросам технического задания, будет наилучшим вариантом для решения поставленной задачи.

## 1.2 Техническое задание

1.2.1 [Общие сведения](http://www.prj-exp.ru/patterns/pattern_tech_task.php#tech_task1)

Название проекта: мобильное приложение «Записная книжка с функцией будильника».

Основание для разработки: приказ на курсовое проектирование.

## 1.2.2 [Назначение и цели создания мобильного](http://www.prj-exp.ru/patterns/pattern_tech_task.php#tech_task2) приложения

Необходимо разработать мобильное приложение, которое выполняет функции записной книжки и будильника.

В приложении должна быть реализована возможность установки будильника (часы и минуты). После установки будильник должен добавиться в будильники телефона. Когда будильник срабатывает – должен выполняется звуковой сигнал, который стоит на рингтоне телефона.

Данные записной книжки должны хранится в базе данных. В базе данных хранится дата создания записи, тема, текст и дата выполнения дела (не обязательно для заполнения).

В распоряжении пользователя должны быть:

* + 1. Все записи записной книжки (все записи, отсортированные по дате с возможностью выбора записи для просмотра/редактирования и добавления новой записи).
    2. Записи на сегодняшний день (дата выполнения равна текущей).
    3. Записи на текущую неделю (по дате выполнения).

Функциональные требования к мобильному приложению:

* корректно отображать данные;
* предоставлять возможность добавлять данные;
* предоставлять возможность редактировать и удалять данные.

Нефункциональные требования:

* легкость и простота использования;
* эффективность и устойчивость к сбоям.

1.2.3 [Характеристика объекта автоматизации](http://www.prj-exp.ru/patterns/pattern_tech_task.php#tech_task3)

В качестве объекта автоматизации представлен смартфон рядового пользователя стандартной комплектации.

1.2.4 [Требования к мобильном](http://www.prj-exp.ru/patterns/pattern_tech_task.php#tech_task4)у приложению

1.2.4.1 Требования к информационному обеспечению

Состав, структура и способы организации данных должны быть определены на этапе технического проектирования.

Уровень хранения данных должен быть построен на основе современных реляционных СУБД. Для обеспечения целостности данных должны использоваться встроенные механизмы СУБД.

Структура базы данных должна поддерживать кодирование хранимой и обрабатываемой информации в соответствии с общероссийскими классификаторами (там, где они применимы).

1.2.4.2 Требования к техническому обеспечению

Для решения задачи используется смартфон рядового пользователя стандартной комплектации.

1.2.4.3 Требования к программному обеспечению

Перечислим основные требования к программному обеспечению:

1. Интерфейс: мобильное приложение.
2. Язык программирования Java. Среда разработки Android Studio.
3. СУБД SQLite.
4. Проверка корректности входных данных.
5. Целостность данных. Каскадное удаление и каскадное обновление данных на уровне БД.

1.2.5 [Состав и содержание работ по созданию мобильного](http://www.prj-exp.ru/patterns/pattern_tech_task.php#tech_task5) приложения

Работы по созданию мобильного приложения включает следующие этапы:

1. Анализ предметной области.
2. Составление ТЗ.
3. Проектирование мобильного приложения.
4. Разработка и реализация.
5. Тестирование мобильного приложения.
6. Оформление документации.

1.2.6 [Порядок контроля и приёмки мобильного приложения](http://www.prj-exp.ru/patterns/pattern_tech_task.php#tech_task6)

Сдача-приемка осуществляется комиссией, в состав которой входят представители заказчика и исполнителя.

Все создаваемые в рамках настоящей работы программные изделия (за исключением покупных) передаются заказчику, как в виде готовых модулей, так и в виде исходных кодов, представляемых в электронной форме на стандартном машинном носителе (например, на компакт-диске).

1.2.7 [Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу мобильного приложения в действие](http://www.prj-exp.ru/patterns/pattern_tech_task.php#tech_task7)

Для того что бы привести мобильное приложение к вводу в действие нужно выполнить следующие операции:

1. Установить приложение на телефон пользователя.
2. Заполнить справочные данные.
3. Сохранить данные.
4. Обучить пользователя.

1.2.8 [Требования к документированию](http://www.prj-exp.ru/patterns/pattern_tech_task.php#tech_task8)

Пояснительная записка формат А4, должна быть оформлена в соответствии с ГОСТ.

# 1.3 Описание структуры приложения

В главном меню программы должны быть следующие пункты: все записи (все записи, отсортированные по дате с возможностью выбора записи для просмотра/редактирования и добавления новой записи), записи на сегодняшний день (дата выполнения равна текущей), записи на текущую неделю (по дате выполнения), установка будильника (ввод времени и подтверждение).

Диаграмма вариантов использования играет центральную роль в моделировании поведения информационной системы или подсистемы. Она отображает множество вариантов использования мобильного приложения, пользователей, а также отношения между ними.

Диаграмма вариантов использования построена с помощью MS Visio и представлена на рисунке 4.



Рисунок 4 – Диаграмма вариантов использования

На диаграмме вариантов использования представлены следующие варианты использования:

1. Работа с заметками (просмотр редактирование данных).
2. Работа с будильником (установка времени).
3. Просмотр заметок (просмотр данных за текущий день и неделю).

Выводы по первой главе

В первой главе представленной работы был проведен обзор аналогов разрабатываемого приложения, рассмотрены преимущества и недостатки программ: FairNote, Omni Notes и SomNote. Принято решение о разработке мобильного приложения собственными силами. Разработано техническое задание на разработку приложения, с помощью MS Visio построена диаграмма вариантов использования.

# 2 Описание разработки приложения

## 2.1 Обоснование средств разработки

Android – операционная система для смартфонов, планшетных компьютеров, электронных книг, цифровых проигрывателей, наручных часов, игровых приставок, нетбуков, смартфонов, очков Google, телевизоров, автомобилей и других устройств. В будущем планируется поддержка бытовых роботов. Основана на ядре Linux и собственной реализации виртуальной машины Java от Google [1].

Среди других мобильных ОС Android делает лидером открытость данной ОС, стремительное развитие и должная техническая поддержка. Открытый код и свободный доступ к публикации приложений позволяет любому разработчику создавать собственные мобильные приложения.

Приложения под операционную систему Android являются программами в нестандартном байт-коде. Разработка приложений для Android ведется на языке Java. Так как все устройства работают с помощью виртуальной машины Java, то это позволяет без особых усилий переносить приложения на другие устройства с данной ОС [2].

В качестве инструментария для разработки приложения Google свободно распространяет свою собственную среду разработку – Android Studio. К другим современным средам мобильных приложений для платформы Android относятся: EmbarcaderoRADStudio – XE5, NetBeansIDE, Eclipse.

RAD Studio является передовым инструментом разработки, когда речь идет о повышении производительности при создании приложений для мобильных устройств. RAD Studio позволяет разрабатывать приложения быстрее при помощи единой базы исходного кода, которая обеспечивает перекрестную компиляцию для Windows, macOS, iOS и Android. Программный каркас FireMonkey позволяет активно осуществлять разработку современных пользовательских интерфейсов, которые распознают межплатформенные различия, предлагают единую базу исходного кода и, в случае необходимости, добавочные блоки для тонкой настройки исходных компонентов, а также позволяют связываться с современными пользовательскими интерфейсами.

NetBeans IDE — свободная интегрированная среда разработки приложений (IDE) на языках программирования Java, Python, PHP, JavaScript, C, C++, Ада и ряда других. Проект NetBeans IDE поддерживается компанией Oracle, однако разработка NetBeans ведётся независимым сообществом разработчиков-энтузиастов (NetBeans Community) и компанией NetBeans Org. Последние версии NetBeans IDE поддерживают рефакторинг, профилирование, выделение синтаксических конструкций цветом, автодополнение набираемых конструкций на лету и множество предопределённых шаблонов кода. Среда разработки NetBeans по-умолчанию поддерживала разработку для платформ J2SE и J2EE. Начиная с версии 6.0 NetBeans поддерживает разработку для мобильных платформ J2ME [3].

Eclipse - свободная интегрированная среда разработки модульных кроссплатформенных приложений. Развивается и поддерживается Eclipse Foundation. Eclipse служит в первую очередь платформой для разработки расширений, чем он и завоевал популярность: любой разработчик может расширить Eclipse своими модулями. Уже существуют Java Development Tools (JDT), C/C++ Development Tools (CDT), разрабатываемые инженерами QNX совместно с IBM, и средства для языков Ada (GNATbench, Hibachi), COBOL, FORTRAN, PHP, X10 (X10DT) и пр. от различных разработчиков. Множество расширений дополняет среду Eclipse диспетчерами для работы с базами данных, серверами приложений и др.

Преимущества среды разработки Android Studio:

1. Большие библиотеки для изучения и внедрения в свой проект.
2. Отличный анализ кода.
3. Отображение файлов xml при разработке.
4. Большой вариант выбора шаблонов, которые позволяют сделать фактически готовое приложение.

Принимая во внимание перечисленные преимущества, а также тот факт, что разрабатываемое приложение предназначено для мобильных устройств, использующих операционную систему Android, логичным будет выбрать данную среду разработки, поскольку она ориентирована на создание программ именно для системы Android и, к тому же, создана компанией производителем данной операционной системы.

В качестве СУБД выбрана SQLite. Компактная встраиваемая СУБД. Исходный код библиотеки передан в общественное достояние. SQLite не использует парадигму клиент-сервер, то есть движок SQLite не является отдельно работающим процессом, с которым взаимодействует программа, а представляет собой библиотеку, с которой программа компонуется, и движок становится составной частью программы. Таким образом, в качестве протокола обмена используются вызовы функций (API) библиотеки SQLite. Такой подход уменьшает накладные расходы, время отклика и упрощает программу. SQLite хранит всю базу данных (включая определения, таблицы, индексы и данные) в единственном стандартном файле на том компьютере, на котором исполняется программа. Простота реализации достигается за счёт того, что перед началом исполнения транзакции записи весь файл, хранящий базу данных, блокируется. Несколько процессов или потоков могут одновременно без каких-либо проблем читать данные из одной базы. Запись в базу можно осуществить только в том случае, если никаких других запросов в данный момент не обслуживается; в противном случае попытка записи оканчивается неудачей, и в программу возвращается код ошибки. Другим вариантом развития событий является автоматическое повторение попыток записи в течение заданного интервала времени [4].

## 2.2 Разработка интерфейса

В Android Studio было создано новое приложение. Имя приложения AndroidApp. Создание проекта представлено на рисунке 5.

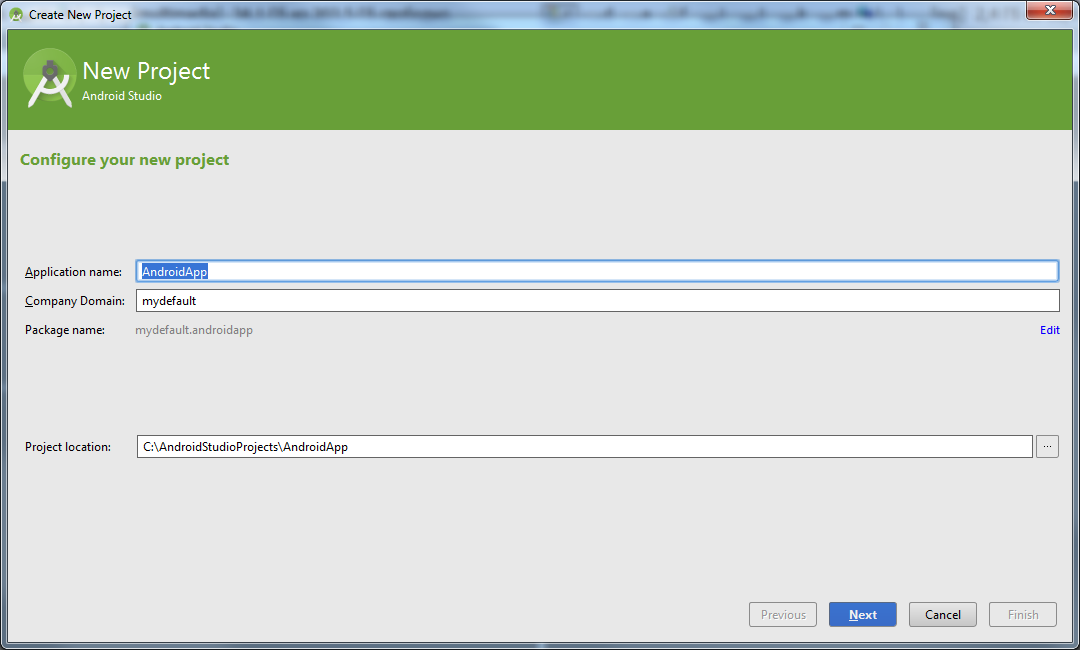


Рисунок 5 – Создание нового проекта в Android Studio

Мобильное приложение разрабатывается для мобильных телефонов. В качестве версии Android выбрана API21:Android 5.0 (Lollilop), рисунок 6.

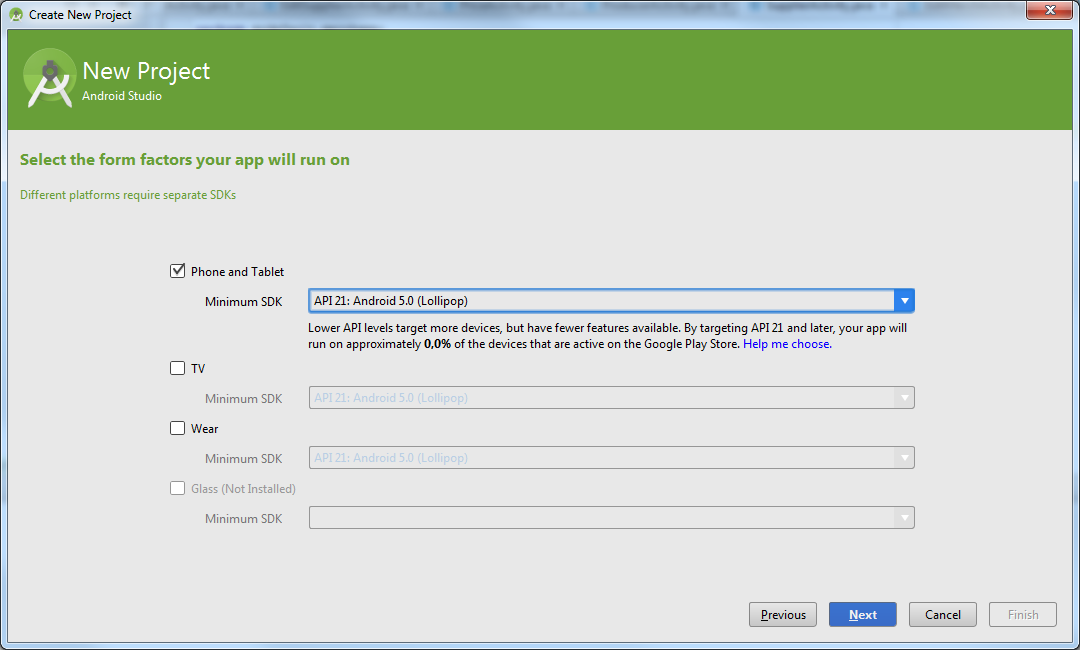


Рисунок 6 – Выбор форм фактора для приложения AndroidApp

Перечень разработанных layout представлен на рисунке 7.

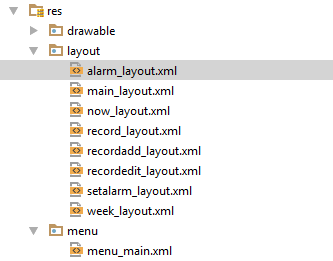


Рисунок 7 – Перечень разработанных layout

Описание разработанных layout представлено в таблице 1.

Таблица 1 - Описание разработанных layout

|  |  |
| --- | --- |
| Layout | Описание |
| main\_layout | Layout отвечает за отображение стартовой страницы |
| menu\_main | Layout отвечает за отображение главного меню программы (пункты меню: «Редактирование записей», «Дела на сегодня», «Дела на неделю», «Установка будильника», «Выход») |
| record\_layout | Layout отвечает за отображение данных таблицы «record» (номер заметки, дата создания, дата выполнения, тема, текст) |
| recordadd\_layout | Layout отвечает за добавление новой записи в таблицу «record» |
| recordedit\_layout | Layout отвечает за редактирование данных в таблице «record» |
| alarm\_layout | Layout выводится на экран в момент срабатывания будильника |
| now\_layout | Layout отвечает за отображение данных таблицы «record» с фильтром заметок за сегодняшний день |
| week\_layout | Layout отвечает за отображение данных таблицы «record» с фильтром заметок на неделю |
| setalarm\_layout | Layout отвечает за установку времени будильника |

## 

## 2.3 Схема взаимодействия компонентов проекта

Если диаграмма вариантов использования дает «вид сверху» на функциональность системы, диаграмма деятельности, напротив, позволяет подробно иллюстрировать отдельный вариант использования и его сценарии. В зависимости от степени детализации диаграммы деятельностей могут использоваться на разных этапах разработки. На этапе анализа требований и уточнения спецификаций диаграммы деятельностей позволяют конкретизировать основные функции разрабатываемого программного обеспечения [6].

Диаграмма деятельности для работы с заметками представлена на рисунке 8.



Рисунок 8 – Диаграмма деятельности для работы с заметками

Диаграмма деятельности для работы с будильником представлена на рисунке 9.



Рисунок 9 – Диаграмма деятельности для работы с будильником

## 2.4 Описание разработанных классов

Перечень разработанных классов представлен на рисунке 10.

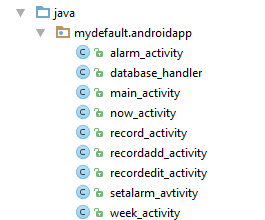


Рисунок 10 – Перечень разработанных классов

Описание разработанных классов представлено в таблице 2.

Таблица 2 - Описание разработанных классов

|  |  |
| --- | --- |
| Класс | Описание |
| database\_handler | Класс отвечает за:   1. Создание базы данных, обновление версии базы данных. 2. Подключение к БД, отключение от БД. 3. Добавление, редактирование и удаление данных в таблицу БД. 4. Вывод данных из таблицы БД. |
| main\_activity | Класс отвечает за отображение стартовой страницы |
| record\_activity | Класс отвечает за отображение данных таблицы «record» (номер заметки, дата создания, дата выполнения, тема, текст) |
| recordadd\_activity | Класс отвечает за добавление новой записи в таблицу «record» |
| recordedit\_activity | Класс отвечает за редактирование данных в таблице «record» |
| alarm\_activity | Класс отвечает за срабатывание будильника |
| now\_activity | Класс отвечает за отображение данных таблицы «record» с фильтром заметок за сегодняшний день |
| week\_activity | Класс отвечает за отображение данных таблицы «record» с фильтром заметок на неделю |
| setalarm\_activity | Класс отвечает за установку времени будильника |

Методы большинства классов представляют собой обработчики нажатия на кнопки форм. Наибольший интерес представляют собой методы класса database\_handler. Остановимся на этих методах подробнее в таблице 3.

Таблица 3 - Описание методов класса database\_handler

|  |  |
| --- | --- |
| Метод | Описание |
| DataBaseHelper | Создание базы данных, обновление версии базы данных.  1. Подключение к БД, отключение от БД.  2. Добавление, редактирование и удаление данных в таблицу БД.  3. Вывод данных из таблицы БД. |
| onCreate | Создание таблиц БД |
| onUpgrade | Удаление и создание таблиц БД заново |

Продолжение таблицы 3

|  |  |
| --- | --- |
| open | Открывание базы данных для записи и чтения |
| close | Закрывание базы данных |
| insertrecord | Добавление данных в таблицу БД |
| updaterecord | Редактирование данных таблицы БД |
| deleterecord | Удаление данных таблицы БД |
| displayrecord | Выборка данных таблицы БД |
| displaynow | Выборка записей таблицы БД, дата выполнения которых соответствует сегодняшнему дню |
| displayweek | Выборка записей таблицы БД, дата выполнения которых соответствует текущей неделе |
| getrecord | Выборка данных записи таблицы БД по идентификатору |

Выводы по второй главе

Во второй главе представленной работы было проведено обоснование средств разработки, в качестве среды разработки выбрана среда Android Studio, в качестве СУБД SQLite. Описаны основные этапы разработки интерфейса программы, разработанные layout и классы. С помощью MS Visio построены диаграммы вариантов использования для работы с заметками и для работы с будильником [7].

# 3 Тестирование и установка приложения

Для тестирования и отладки приложения было создано виртуальное устройство Galexy Nexus API 21, рисунок 11.

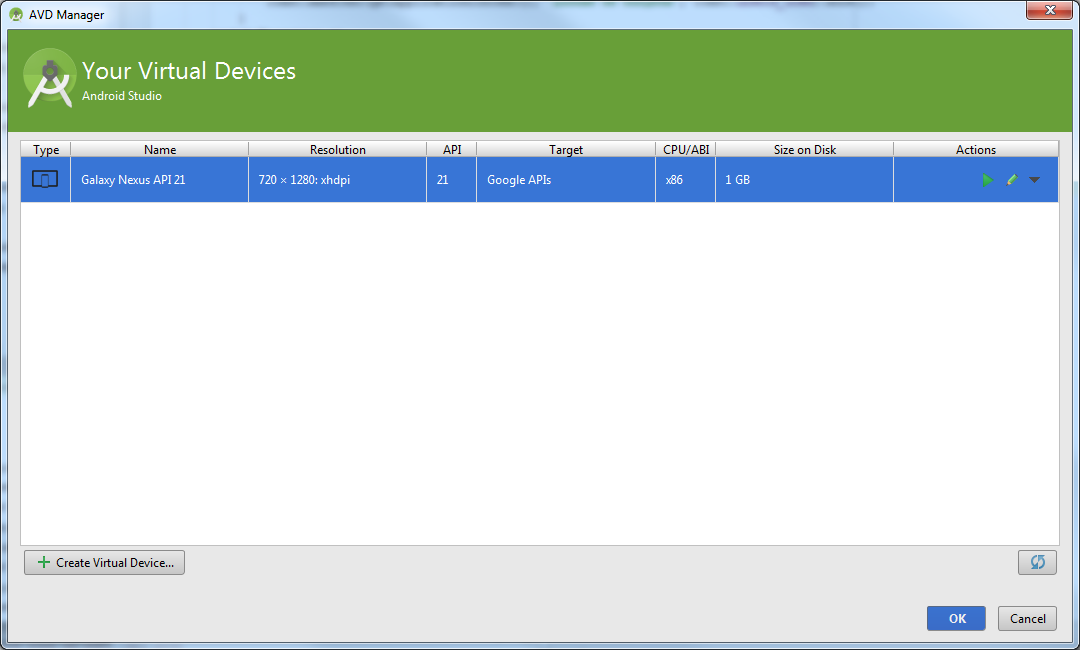


Рисунок 11 – Созданное виртуальное устройство

Параметры виртуального устройства представлены на рисунке 12.

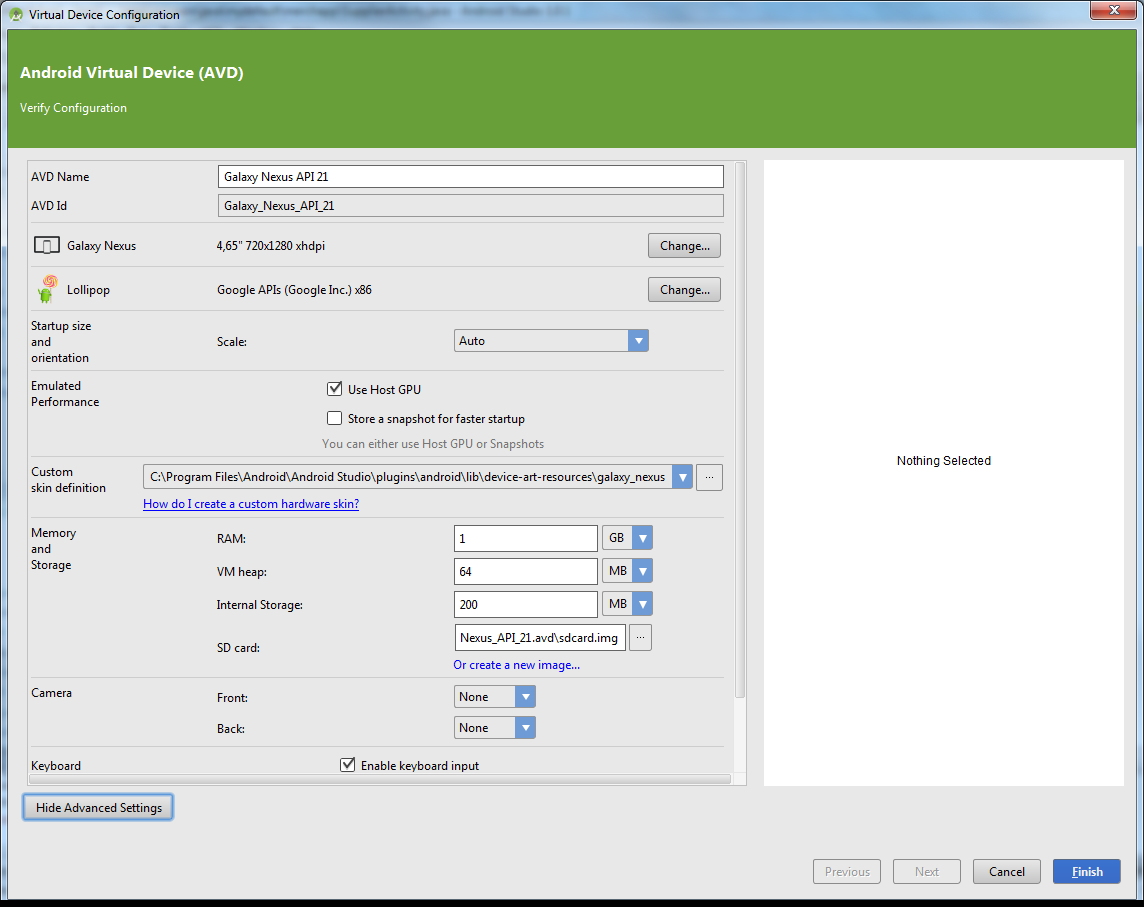


Рисунок 12 – Параметры виртуального устройства

Выбор виртуального устройства при запуске приложения представлена на рисунке 13.

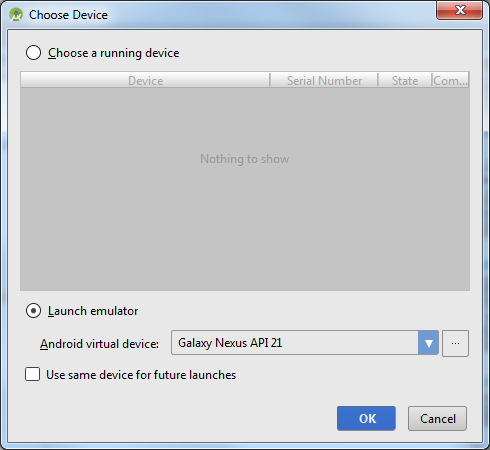


Рисунок 13 – Выбор виртуального устройства при запуске приложения

Загрузка операционной системы на эмуляторе представлена на рисунке 14. [8]

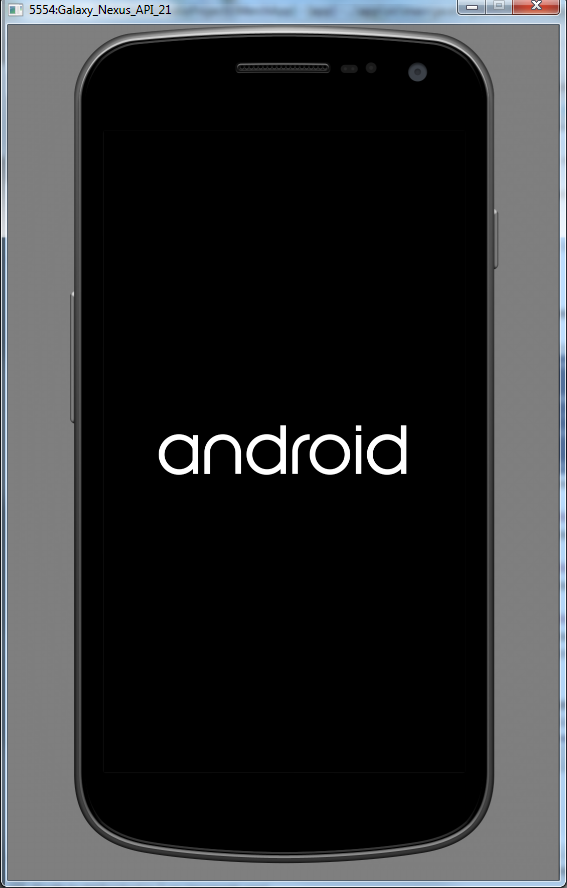


Рисунок 14 – Загрузка операционной системы на эмуляторе

После запуска приложения на эмуляторе необходимо разблокировать экран, рисунок 15.



Рисунок 15 – Разблокировка экрана эмулятора

Приложение запускается на эмуляторе автоматически, стартовая форма приложения представлена на рисунке 16.

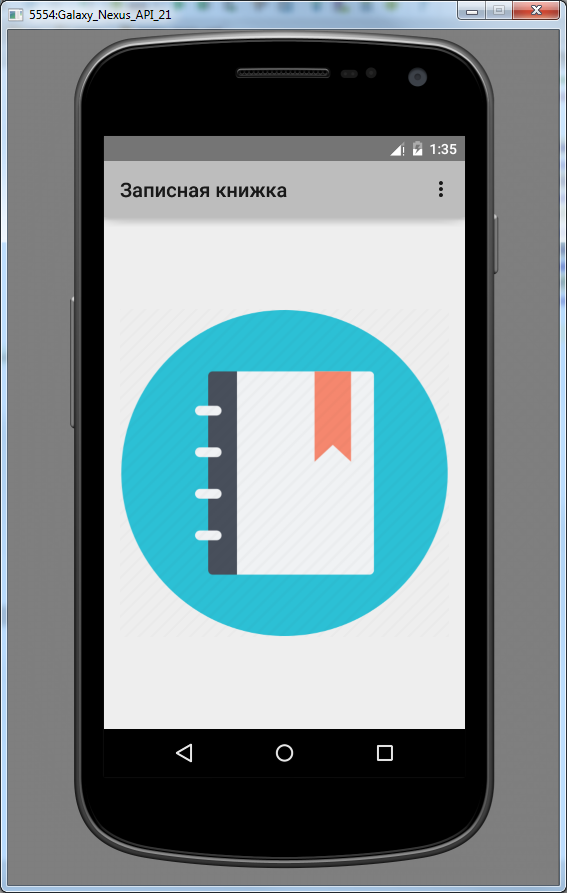


Рисунок 16 – Стартовая форма приложения

Меню приложения представлено на рисунке 17.

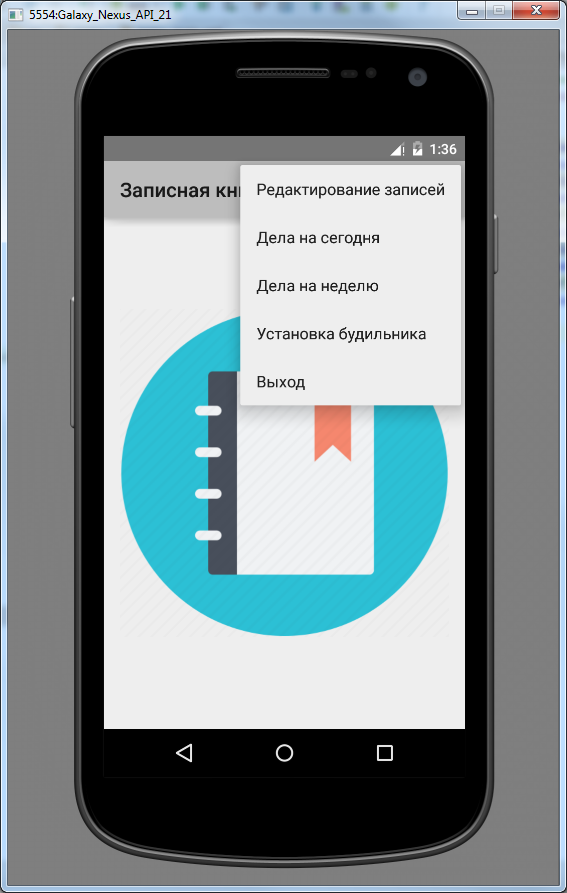


Рисунок 17 – Главное меню приложения

Форма для работы с заметками представлена на рисунке 18.

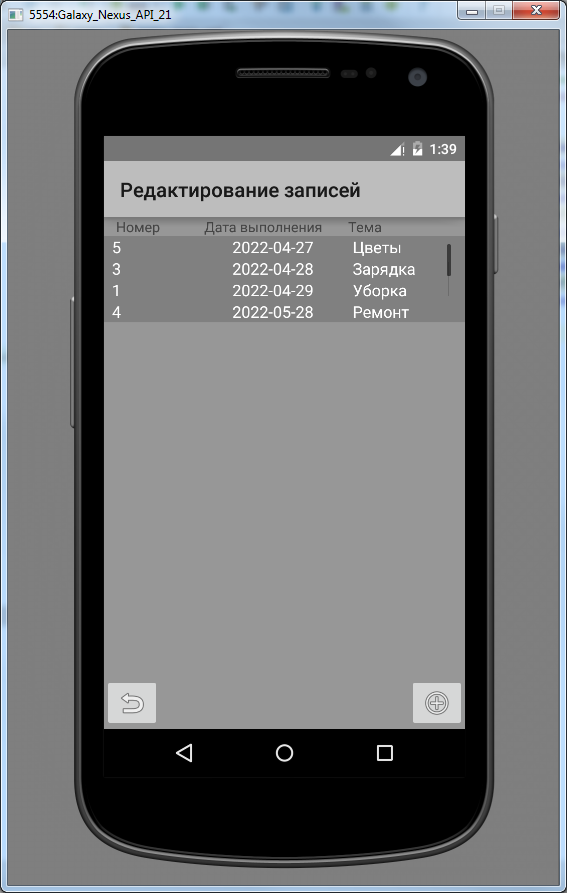


Рисунок 18 – Форма для работы с заметками

Для возращения на главную форму приложения необходимо нажать на кнопку . Для добавления данных в таблицу «record» необходимо нажать на кнопку . Форма добавления новой записи в таблицу «record» представлена на рисунке 19.

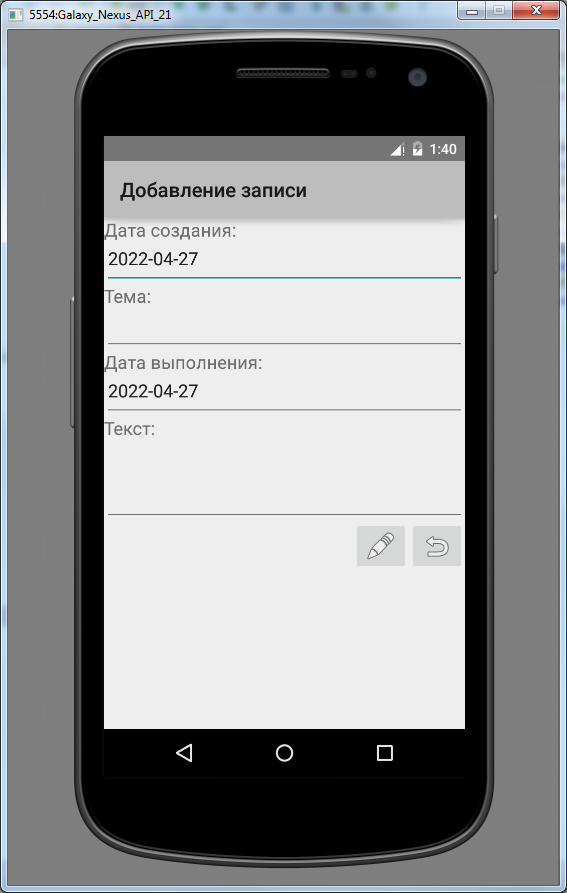


Рисунок 19 – Форма добавления новой записи в таблицу «record»

Для добавления данных необходимо нажать на кнопку . Для отмены внесенных изменений необходимо нажать на кнопку . [9]

Для редактирования данных таблицы «record» необходимо выбрать строку в таблице на форме заметок. Форма редактирования выбранной записи представлена на рисунке 20.

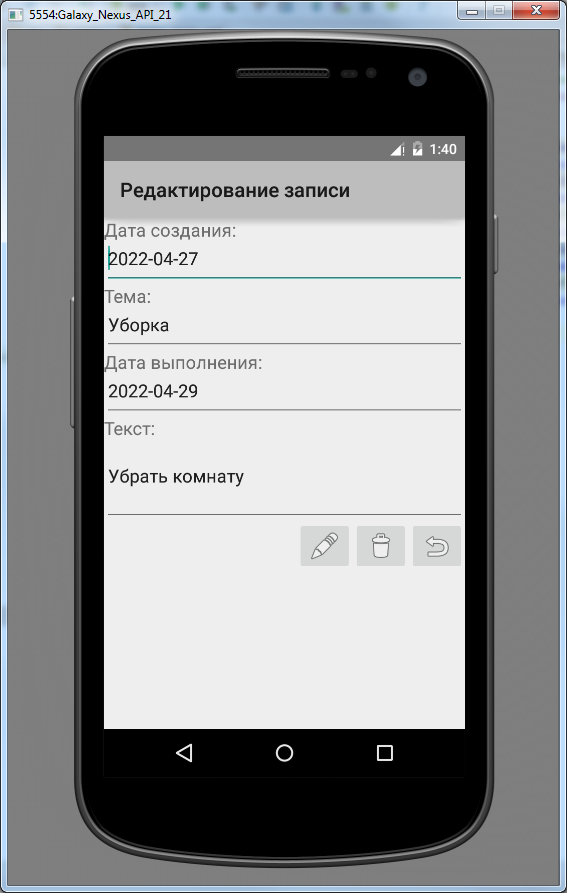


Рисунок 20 – Форма редактирования записи таблицы «record»

Для сохранения изменений необходимо нажать на кнопку . Для отмены внесенных изменений необходимо нажать на кнопку . Для удаления выбранной записи необходимо нажать на кнопку .

Форма для отображения дел, которые необходимо сделать сегодня, представлена на рисунке 21.



Рисунок 21 – Форма для отображения дел, которые необходимо сделать сегодня

Форма для отображения дел, которые необходимо сделать на этой неделе, представлена на рисунке 22.

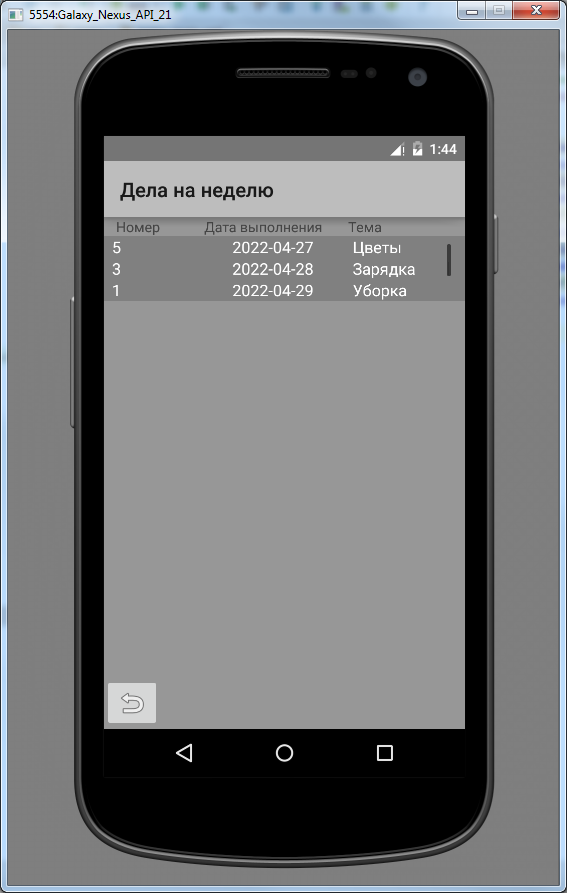


Рисунок 22 – Форма для отображения дел, которые необходимо сделать на этой неделе

Форма для установки будильника представлена на рисунке 23.

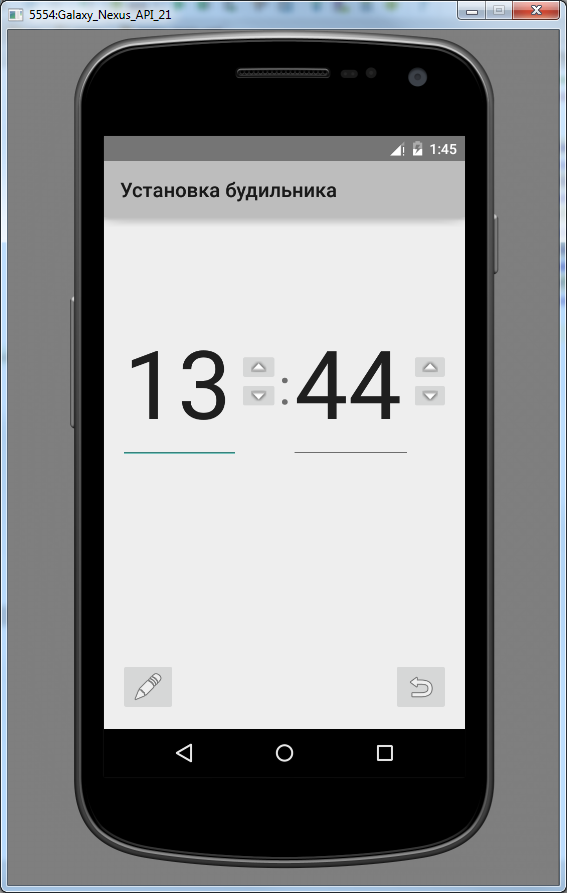


Рисунок 23 – Форма для установки будильника

После установки будильника на экран выводится соответствующее сообщение, рисунок 24.

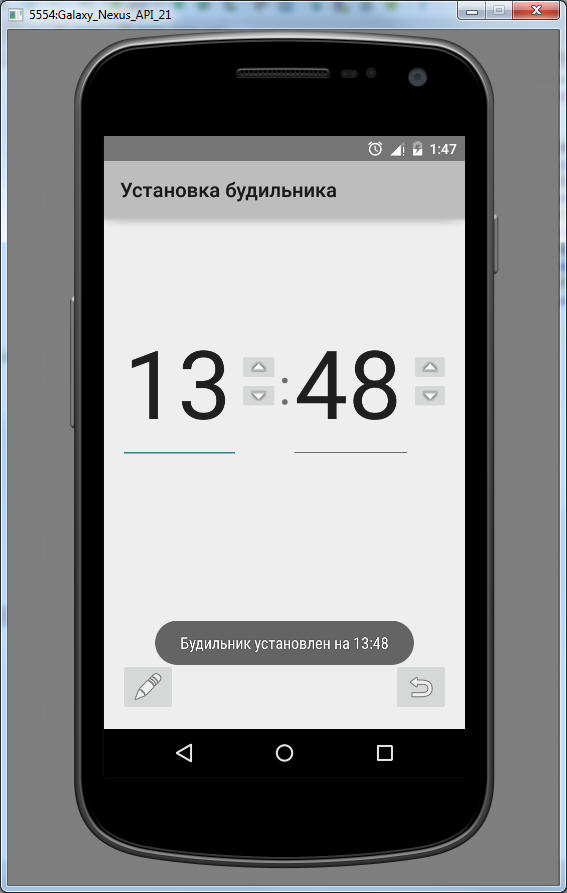


Рисунок 24 – Подтверждение установки будильника

Когда время будильника наступит, на экран будет выведена форма с соответствующим сообщением, рисунок 25.

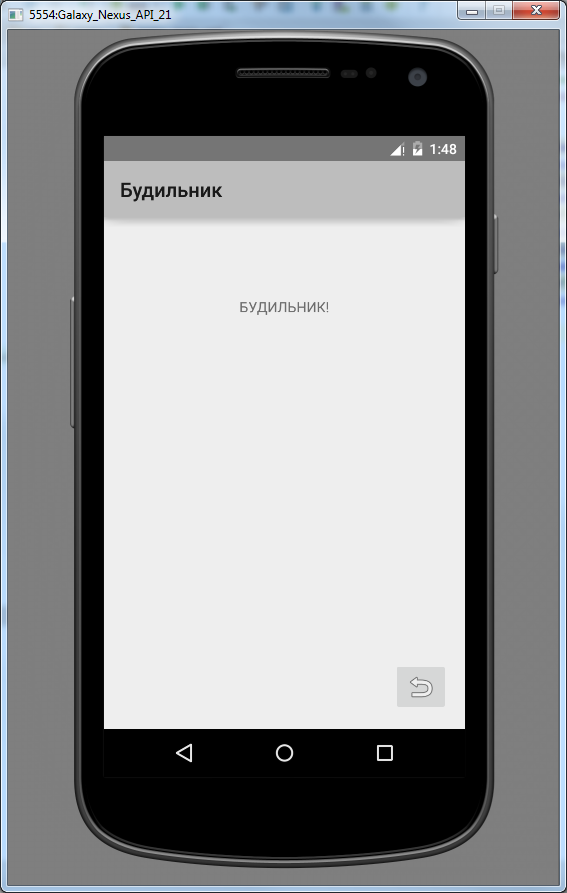


Рисунок 25 – Срабатывание будильника

Выводы по третьей главе

В третьей главе представленной работы было проведено тестирование приложения. Тестирование показало, что функционал, заложенный в приложение, работает в соответствии с предъявляемыми функциональными требованиями. [10]

# Заключение

В ходе выполнения представленной работы было разработано мобильное приложение под Android для работы с датой, временем и заметками.

Перечислим функции разработанного приложения:

1. Работа с заметками (просмотр редактирование данных).
2. Работа с будильником (установка времени).
3. Просмотр заметок (просмотр данных за текущий день и неделю).

В ходе выполнения работы был проведен обзор и анализ существующих информационных ресурсов предметной области, определены требования к ресурсам, выбраны наиболее подходящие средства реализации, спроектирована и реализована структура приложения. Было произведено тестирование приложения, в ходе которого было подтверждено, что приложения работает на операционных системах Android. [11]

Таким образом, задачи работы полностью решены, цели достигнуты. Разработанный проект – прост, и удобен в использовании, Данным программным продуктом могут пользоваться все, кто владеет элементарными навыками использования мобильным устройством. [12]

# Список использованной литературы

1. Седжвик Р. Алгоритмы на Java / Р. Седжвик, К. Уэйн . – М.: ООО “И.Д. Вильямс”, 2013. – 848 с
2. Кузин А. В., С. В. Левонисова, Базы данных, - Москва, Издательство: Академия, 2014 г., 320 стр.
3. Кумскова И. А., Базы данных, - Москва, Издательство: КноРус, 2014 г., 488 стр.
4. Вигерс, Битти: Разработка требований к программному обеспечению, - Москва, Издательство: BHV, 2014 г., 736 стр.
5. Брайан Хоган: HTML5 и CSS3. Веб-разработка по стандартам нового поколения, - Москва, Издательство: Питер, 2014 г., 320 стр.
6. Вадим Дунаев: HTML, скрипты и стили, - Москва, Издательство: BHV, 2015 г., 824 стр.
7. Гамма Э. Приемы объектно-ориентированного проектирования. Паттерны проектирования / Гамма Э., Хелм Р., Джонсон Р., Влиссидес Дж. - СПб.: Питер, 2015 г. – 368 с.
8. Грекул В.И. Проектирование информационных систем / Грекул В.И., Денищенко Г.Н., Коровкина Н.Л. – М.: Интернет-университет информационных технологий, 2018 г. - 321 с.
9. Дубейковский В.И. Эффективное моделирование с CA ErwinProcessModeler (BPwin; AllfusionProcessModeler)./В.И. Дубейковский. – М.: Диалог-МИФИ, 2015. – 384 с.
10. Илюшечкин В.М. Основы использования и проектирования баз данных. Учебник / В.М. Илюшечкин. - М.: Юрайт, 2015 г. - 214 с.
11. Кириллов В.В. Введение в реляционные базы данных. Введение в реляционные базы данных / В.В. Кириллов, Г.Ю. Громов - СПб.: БХВ-Петербург, 2015 г. - 464 c.
12. Коваленко В.В. Проектирование информационных систем / В.В. Коваленко. - М.: Форум, 2016 г. - 320с.

# Приложение А

# Код программы

package mydefault.androidapp;

import android.app.Activity;

import android.content.Intent;

import android.os.Bundle;

import android.view.Menu;

import android.view.MenuItem;

import android.widget.Button;

public class main\_activity extends Activity {

private Button enter\_button;

database\_handler handler;

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.main\_layout);

}

public boolean onCreateOptionsMenu(Menu menu) {

// Inflate the menu; this adds items to the action bar if it is present.

getMenuInflater().inflate(R.menu.menu\_main, menu);

return true;

}

public boolean onOptionsItemSelected(MenuItem item) {

// Handle action bar item clicks here. The action bar will

// automatically handle clicks on the Home/Up button, so long

// as you specify a parent activity in AndroidManifest.xml.

int id = item.getItemId();

if (id == R.id.action\_exit) {

finishAffinity();

System.exit(0);

return true;

}

if (id == R.id.action\_record) {

Intent i=new Intent(this,record\_activity.class);

startActivity(i);//start activity

return true;

}

if (id == R.id.action\_now) {

Intent i=new Intent(this,now\_activity.class);

startActivity(i);//start activity

return true;

}

if (id == R.id.action\_week) {

Intent i=new Intent(this,week\_activity.class);

startActivity(i);//start activity

return true;

}

if (id == R.id.action\_alarm) {

Intent i=new Intent(this,setalarm\_avtivity.class);

startActivity(i);//start activity

return true;

}

return super.onOptionsItemSelected(item);

}

}