

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМ. В. ДАЛЯ
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ТА ЕЛЕКТРОНІКИ
КАФЕДРА КОМП'ЮТЕРНИХ НАУК ТА ІНЖЕНЕРІЇ

До захисту допускається
Завідувач кафедри
_____ І. С. Скарга-Бандурова
« ____ » _____ 2020 р.

ДИПЛОМНИЙ ПРОЕКТ (РОБОТА) БАКАЛАВРА

ПОЯСНЮВАЛЬНА ЗАПИСКА

НА ТЕМУ:

Менеджер файлів для ОС Android

Освітній ступінь “бакалавр”

Спеціальність 122 Комп'ютерні науки
(шифр і назва спеціальності)

Керівник проекту:

_____ (підпис)

М.Є. Щербакова

_____ (ініціали, прізвище)

Консультант з охорони праці:

_____ (підпис)

Я.О. Критська

_____ (ініціали, прізвище)

Здобувач вищої освіти:

_____ (підпис)

В.О. Гростянський

_____ (ініціали, прізвище)

Група:

_____ КН-16з

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВОЛОДИМИРА ДАЛЯ

Факультет Інформаційних технологій та електроніки
Кафедра Комп'ютерних наук та інженерії
Освітній ступінь бакалавр
Спеціальність 122 Комп'ютерні науки
(шифр і назва)

ЗАТВЕРДЖУЮ:

Т.в.о. завідувача кафедри КНІ
С.О.Сафонова
« » 2020 р.

З А В Д А Н Н Я
НА ДИПЛОМНИЙ ПРОЕКТ (РОБОТУ) БАКАЛАВРА

Тростянського Владислава Олександровича

(прізвище, ім'я, по батькові)

1. Тема роботи Менеджер файлів для ОС Android

керівник проекту (роботи) Щербакова М.Є., к.т.н., доцент
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу від "30" 04 2020 р. № 77/15.15

2. Термін подання здобувачем вищої освіти роботи 10.06.2020

3. Вихідні дані до роботи ОС Android, мова програмування Java,
комплект розробника додатків Java Development Kit

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) 1) Аналіз та постановка задачі

2) Апаратні та програмні засоби розробки додатку

3) Розробка і тестування менеджера файлів

4) Охорона праці

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)
Електронні плакати

6. Консультанти розділів проекту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Охорона праці	Критська Я.О., ст.викл.		

7. Дата видачі завдання _____

Керівник _____
(підпис)

Завдання прийняв до виконання _____
(підпис)

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів проекту (роботи)	Примітка
1	Пошук літературних джерел	27.04.20-29.04.20	
2	Вивчення літературних джерел	29.04.20-04.05.20	
3	Розробка технічного завдання	04.05.20-05.05.20	
4	Проектування структури додатку	06.05.20-08.05.20	
5	Розробка менеджера файлів	08.05.20-25.05.20	
6	Розробка заходів по безпечному веденню робіт	26.05.20-30.05.20	
7	Оформлення пояснювальної записки	30.05.20-09.05.20	

Здобувач вищої освіти _____
(підпис)

Керівник _____

В.О. Тростянський _____
(ініціали, прізвище)

М.Є. Щербакова _____

РЕФЕРАТ

Пояснювальна записка до дипломного проекту (роботи) бакалавра: 75 с., 27 рис., 4 табл., 30 бібліографічних джерел, 2 додатки.

Об'єкт розробки: інформаційна система для перегляду файлів в операційній системі Android

Мета роботи: розробка інформаційної системи для перегляду і управління файлами в операційній системі Android

В проекті виконано:

1. Аналіз аналогів та визначення їх недоліків.
2. Вибір платформи і програмних засобів реалізації додатку.
3. Визначення структури додатку.
4. Розробка програмних засобів перегляду файлів для мобільної платформи.

Практичне значення, галузь застосування роботи: мобільні пристрої.

МОБІЛЬНА ПЛАТФОРМА, JAVA, ІНТЕРНЕТ, ІНТЕРФЕЙС,
АНДРОЇД-ДОДАТОК

Умови одержання дипломного проекту: СНУ ім. В. Даля, пр.
Центральний 59-А, м. Сєвєродонецьк, 93400

ЗМІСТ

ВСТУП	7
1 АНАЛІЗ ТА ПОСТАНОВКА ЗАДАЧІ	8
1.1 Аналіз предметної області.....	8
1.2 Аналіз та основні характеристики аналогів.....	11
1.2.1 ASTRO File Manager	12
1.2.2 ES File Explorer	12
1.2.3 AntTek Explorer.....	13
1.2.4 Порівняльний аналіз аналогів	15
1.3 Постановка задачі	16
1.4 Висновки до розділу 1	17
1.5 Перелік джерел посилань до розділу 1	17
2 АПАРАТНІ ТА ПРОГРАМНІ ЗАСОБИ РОЗРОБКИ ДОДАТКУ.....	19
2.1 Вибір апаратної платформи.....	19
2.2 Вибір операційної системи.....	20
2.2.1 Особливості архітектури ОС Android	23
2.3 Вибір засобів розробки та тестування.....	26
2.3.1 Мова Java.....	26
2.3.2 Java Development Kit	27
2.3.3 Інтегроване середовище розробки Eclipse.....	28
2.3.4 Android SDK.....	28
2.4 Висновки до розділу 2	29
2.5 Перелік джерел посилань до розділу 2	29
3 РОЗРОБКА І ТЕСТУВАННЯ МЕНЕДЖЕРУ ФАЙЛІВ	30
3.1 Опис програмного продукту	30
3.1.1 Функціональне призначення	30
3.1.2 Вхідні та вихідні дані.....	34
3.1.3 Логічна структура програми та її модулів.....	35
3.1.4 Організація інтерфейсу з користувачем	37
3.5 Тестування та аналіз результатів	38

3.6	Встановлення та налаштування програми	41
3.7	Висновки до розділу 3	44
3.8	Перелік джерел посилань до розділу 3	44
4	ОХОРОНА ПРАЦІ	45
4.1	Поняття охорони праці.....	45
4.2.	Охорона праці при роботі з комп'ютером	46
4.2.1	Загальні вимоги безпеки.....	46
4.2.1.1	Розрахунок рівня шуму в приміщенні.....	47
4.2.1.2	Розрахунок потрібного повітрообміну в приміщенні	50
4.2.2.	Вимоги безпеки перед початком роботи	53
4.2.3	Вимоги безпеки під час роботи.....	54
4.2.4	Вимоги безпеки в аварійних ситуаціях.....	56
4.2.5	Вимоги безпеки після закінчення роботи	57
4.3	Висновки до розділу 4	57
4.4	Перелік використаних джерел посилань до розділу 4	58
	ВИСНОВКИ.....	59
	Додаток А Лістинг програмного коду	60
	Додаток Б Комп'ютерна презентація	69

ВСТУП

У наш час практично всі галузі діяльності людини автоматизовані; крім цього, всі завдання повинні вирішуватися з точки зору оптимізації наявних ресурсів. Так, на сьогодні, до основних сфер діяльності людини варто віднести освітянську галузь. Освітянська галузь України представлена великою кількістю різноманітних закладів, які відрізняються спрямуванням діяльності та формою власності, проте майже кожен з них в процесі своєї роботи надає користувачеві низку файлів для навчання та самостійного розвитку.

Саме тут велике практичне значення мають сучасні інформаційні технології, котрі увійшли у життя кожної сучасної людини. Дуже важливим є питання перегляду файлів різного складу та формату на власному мобільному телефоні, планшеті, іншому гаджеті у будь-якому місці. Зникає потреба постійної наявності ПК. Автоматизуються реальні потреби користувачів у різних галузях. Виходячи з того, що майже кожна сучасна людина володіє мобільними технологіями, актуальність розробки не здається сумнівною.

Мета дипломного проекту – розробити мобільний додаток під ОС Андроїд для перегляду файлів. Має бути забезпечена можливість перегляду різних форматів файлів, вивід переліку файлів, їх структуризація, переміщення та видалення. Для досягнення поставленої мети потрібно буде дати наступні визначення: мобільні технології, протоколи, режими, мобільні додатки, охарактеризувати мобільні сервіси. Також необхідно розкрити технології розробки мобільного додатку під ОС Андроїд і здійснити розробку інформаційних програмних засобів.

1 АНАЛІЗ ТА ПОСТАНОВКА ЗАДАЧІ

1.1 Аналіз предметної області

Сьогодні завдяки технічному прогресу мало не щодня випускається маса інноваційних розробок в області мобільних технологій. Часто це відбувається настільки швидко, що часом ми навіть не встигаємо встежити за всіма новинками. Якщо раніше подзвонити можна було тільки по стаціонарному телефону, а вийти в Інтернет тільки з персонального домашнього комп'ютера, то зараз навряд чи кого здивуєш мобільним телефоном з бездротовим Інтернетом, Wi-Fi модулем та GPS-навігацією [1]. Нові мобільні технології поширюються по землі неймовірно швидкими темпами. Сьогодні до мобільних протоколів зв'язку можна віднести WiMAX Mobile, WAP, GPS, GSM, GPRS, UMTS, EDGE, Wi-Fi. Дані аббревіатури трапляються нам на очі постійно, як розшифровуються деякі з них, нам приблизно відомо, а про інші маємо лише смутні асоціації. Більшість з нас знають, що UMTS і GSM - це стандарти зв'язку, EDGE і GPRS - протоколи передачі даних, WAP - протокол для отримання доступу до мережі Інтернет з мобільного телефону. Деяких навіть не дивують такі нові мобільні технології, як Wi-Fi і GPS, але лише деякі можуть виразно пояснити, що таке WiMAX.

А між іншим, WiMAX - технологія передачі даних, що працює за принципом її попередника Wi-Fi і дозволяє отримувати доступ в Інтернет. Розроблена вона була ще в 2001 році.

Так само у нас перед очима відбувається не менш дивовижна річ – стрімко розвивається нове покоління зв'язку – мережа 4G. Мобільні пристрої, що працюють в сімействі мереж 4G, дозволяють здійснювати передачу даних на швидкостях понад 100 Мбіт на секунду. В теорії швидкість може перевищувати кордон у 1Гбіт в секунду. Для порівняння, в 3G, які

функціонують на великих частинах США, Європи та Азії, швидкість не перевищує 14 Мбіт в сек (рис. 1.1).

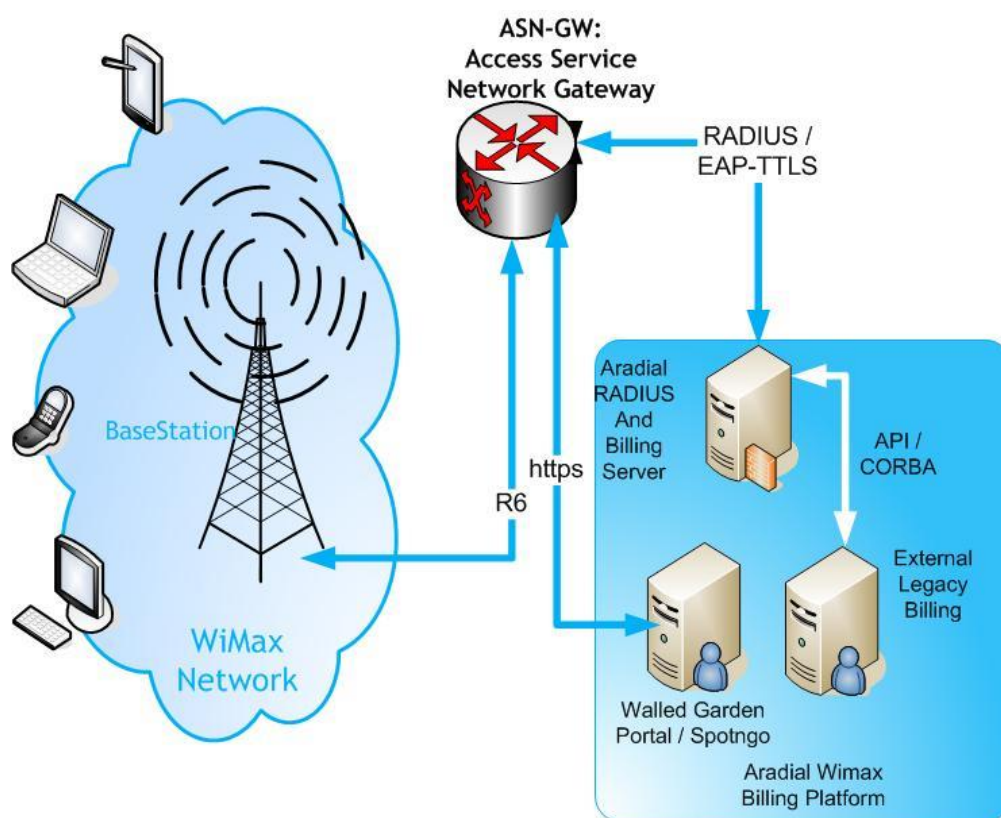


Рисунок 1.1 – Структурна схема бездротових мобільних технологій [2]

Детальніше розглянемо мобільну технологію 4G.

4G (від англ. Fourth generation – четверте покоління) – покоління мобільного зв'язку з підвищеними вимогами. До четвертого покоління прийнято відносити перспективні технології, що дозволяють здійснювати передачу даних зі швидкістю, що перевищує 100 Мбіт/с - мобільним (з високою мобільністю) і 1 Гбіт/с – стаціонарним абонентам (з низькою мобільністю).

Технології LTE Advanced (LTE-A) і WiMAX 2 (WMAN-Advanced, IEEE 802.16m) (SIM-карта не потрібна) були офіційно визнані бездротовими стандартами зв'язку четвертого покоління 4G (IMT-Advanced) Міжнародним союзом електрозв'язку на конференції в Женеві в 2012 році.

Стандарти 4-го покоління (4G) є подальшим розвитком світових телекомунікаційних технологій в галузі мобільного зв'язку. Вони дозволяють забезпечити ще більші швидкості передачі даних, що означає і підвищення якості послуг, що поставляються, і швидке поширення сучасних мультимедійних сервісів (соціальні мережі, розраховані на багато користувачів ігри, інтерактивні онлайн-додатки, відеоконференції, відеодзвінки, послуги позиціонування і багато інших) [3].

Стандарт LTE (Long Term Evolution – довгострокова еволюція) вважається логічним розвитком технологій 3G [4], його впровадження є перспективним напрямком розвитку мереж. Основними цілями створення стандарту LTE можна назвати нарощування можливостей високошвидкісних систем мобільного зв'язку, зменшення вартості передачі даних, можливість надання широкого спектра недорогих послуг, підвищення гнучкості використання вже існуючих систем. LTE відрізняється від 3G підвищеною ємністю, кращим використанням частотного спектра і меншою затримкою при передачі пакетів даних (рис. 1.2).

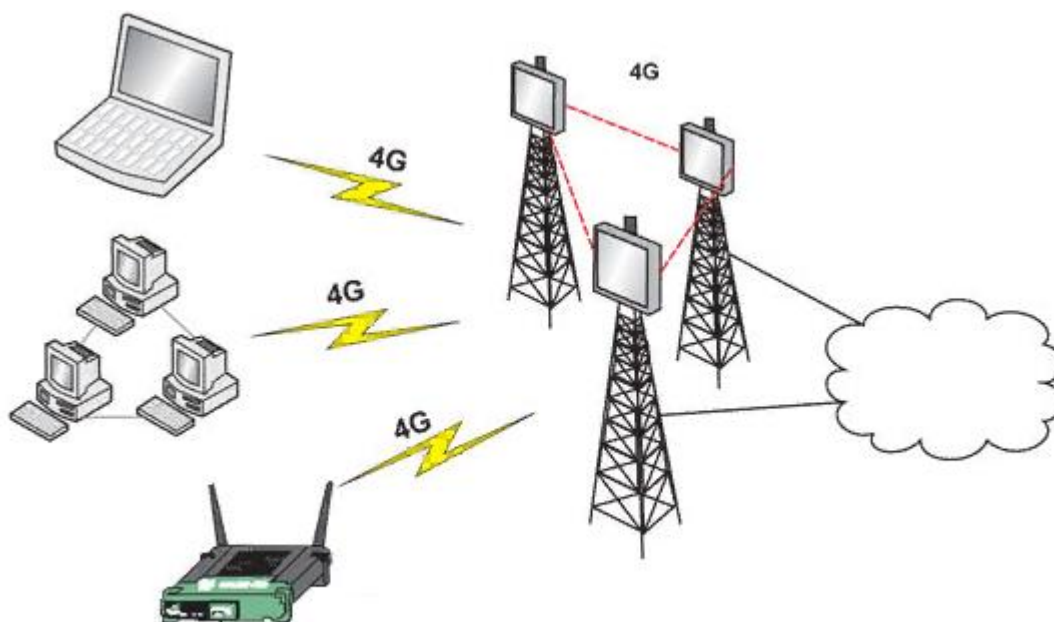


Рисунок 1.2 – Структурна схема технології 4G [5]

До мобільних технологій можна віднести супутникові VSAT-технології (рис. 1.3) [6]. Розквіт цих технологій у світі припав на 2000 роки, в ті часи мобільний зв'язок не забезпечував передачу даних по своїх каналах [7]. В Україні ринок супутникового зв'язку має специфічний розвиток, оскільки стимулюється в основному державою.

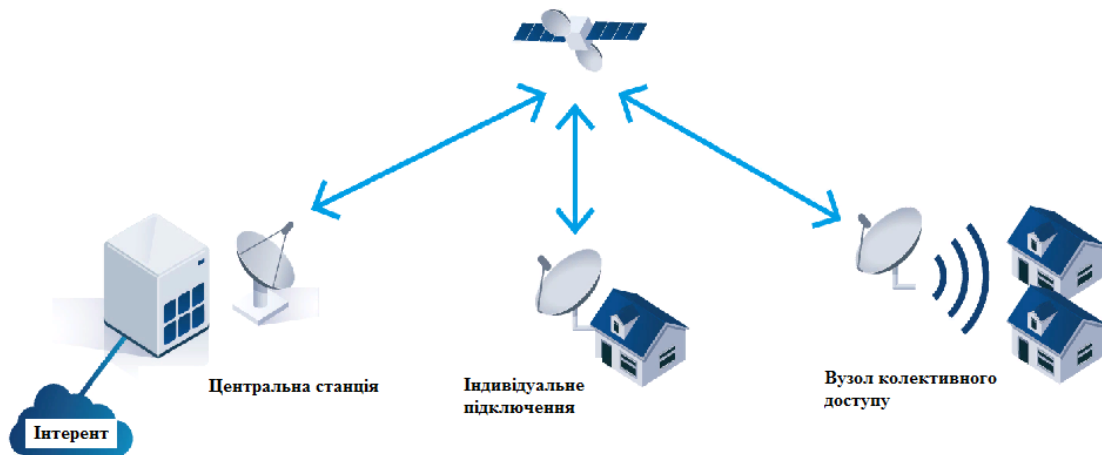


Рисунок 1.3 – Супутникові VSAT-технології [8]

В рамках дипломного проекту заплановано розробити менеджер файлів для перегляду і запуску файлів різних типів на мобільному пристрої.

1.2 Аналіз та основні характеристики аналогів

Мобільні сервіси широко використовуються в усіх сферах людського життя з високою конкуренцією. Розглянемо сучасні аналоги, що пропонуються на ринку.

1.2.1 ASTRO File Manager

ASTRO File Manager - це досить просунутий файловий менеджер для операційної системи Андроїд, в ньому є можливість бекапа програм, підтримка ярликів і zip архівів. Серед ключових можливостей цієї програми варто відзначити наступні:

- управління файлами, включаючи стиснення подібно zip і tar;
- бекап додатків;
- менеджер запущених додатків;
- відправка файлів як вкладення;
- перегляд ескізів і зображень;
- огляд і створення стислих файлів подібно Zip і Tar (рис. 1.4).



Рисунок 1.4 – Мобільний додаток ASTRO File Manager

1.2.2 ES File Explorer

Наступна програма, яка привернула увагу - ES File Explorer. На мій погляд, це зручний файловий менеджер. Він може відображати файли і папки у вигляді списку і ярликів. Так само є можливість підключення до віддалених машин.

Програма дозволяє:

- копіювати, переміщати, видаляти файли;
- виводити прив'ю;
- використовувати мультिवибір файлів;
- редагувати текстові файли;
- шукати файли;
- розпакувати zip-архіви;
- керувати файлами на FTP;
- отримувати доступ до файлів комп'ютера;
- керувати файлами через bluetooth;
- відсилати файли через наявні клієнти (Facebook, Twitter, BFT, Hancend SMS, GMail і т.д.) (рис. 1.5).

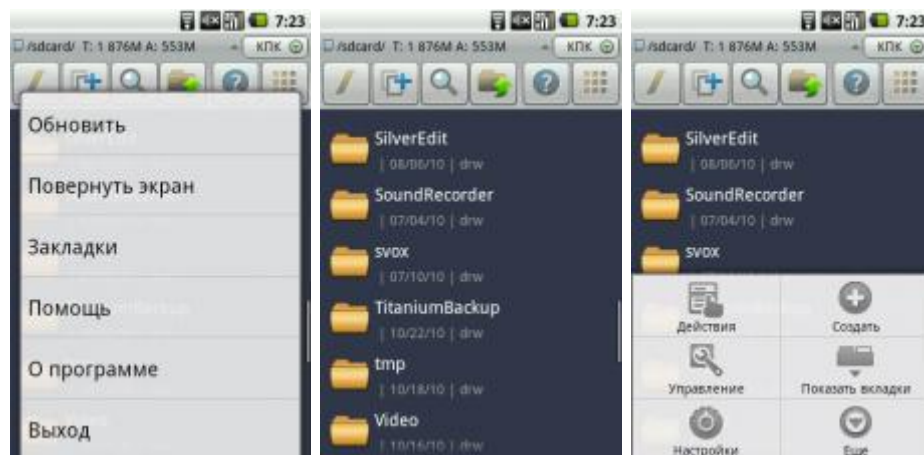


Рисунок 1.5 – ES File Explorer

1.2.3 AntTek Explorer

Нові файлові менеджери часто виглядають не доопрацьованими, мають дуже простий і не цікавий інтерфейс, проте AntTek Explorer вирізняється з поміж аналогів [9]. У менеджера дуже професійна настройка загального вигляду, можна докорінно змінити вид або включити додаткову панель (це

зручно, якщо на смартфоні/планшеті є SD-накопичувач і внутрішня пам'ять), це буде нагадувати Total Comander на PC. Завдяки закладкам можна забезпечити швидкий доступ до потрібних файлів, є зручний віджет (рис. 1.6).



Рисунок 1.6 – AntTek Explorer

Додаток не став тільки одним "браузером файлів", там є і медіа-плеєр, і менеджер переглядів зображень [10]. За допомогою додаткових розширень можна побачити підтримку .pdf та інших файлів. Дані можна швидко відправити через Bluetooth, Wi-Fi, на пошту або в хмарне сховище Dropbox і Box, використовуючи свій акаунт, більше того, програма має свій менеджер передачі [11]. Є і вбудований архіватор/разархіватор .zip і .rar файлів, підтримує кілька типів стиснення. Крім усього, можна активувати ROOT режим для доступу до системних файлів (рис. 1.7).



Рисунок 1.7 – AntTek Explorer

1.2.4 Порівняльний аналіз аналогів

Здійснимо порівняльний аналіз наведених аналогів, задля цього зведемо у таблицю основні характеристики, та охарактеризуємо кожен з аналогів у рамках табл. 1.1.

Таблиця 1.1 – Порівняльна характеристика аналогів

Назва Характеристика	ASTRO File Manager	ES File Explorer	AntTek Explorer
Каталог всіх файлів	+	+	+
Формат файлів txt ; doc; docx	+	+	+
Відтворення музики	-	-	+
gif-анімація	-	-	+
Структуризація файлів	+	-	+
Інформація про всі файли	+	+	+
Необхідність підключення до Інтернету	+	-	-
ОС	Android 5 і вище	Android 4.2 і вище	Android 4 і вище
Розмір	11.84 М	95 М	4,6 М
Вартість	умовно безкоштовна	умовно безкоштовна	умовно безкоштовна

Необхідні функції, які повинні бути в файловому менеджері: перегляд директорій і файлів на мобільному пристрої; копіювання, переміщення, перейменування і видалення директорій і файлів. Ці функції є у всіх перерахованих вище додатках. У файлових менеджерах часто реалізуються також групування медіафайлів, робота з різними розташуваннями внутрішньої пам'яті, попередній перегляд зображень, вбудовані засоби роботи з архівами, робота з хмарними сховищами (Google Диск, Drobbox, OneDrive та інші), підтримка протоколу FTP і підключення по локальній мережі.

Незважаючи на те, що розглянуті файлові менеджери є одними з найбільш популярних в даному сегменті додатків, за відгуками користувачів вони мають істотні недоліки.

В додатку ES File Explorer є практично все, що може знадобитися від файлового менеджера на ОС Android. Функціональність більша, ніж в інших подібних програмах. Однак останні його версії стали сприйматися користувачами не дуже однозначно: повідомляється про спливаючі повідомлення, погіршення інтерфейсу (з точки зору деяких користувачів) та інші зміни, які говорять на користь пошуку іншої програми [12].

За відгуками користувачів ASTRO File Manager про його останні версії, в ньому також погіршився користувацький інтерфейс у порівнянні з попередніми версіями, не всі заявлені функції працюють, додаток не може запускати музичні файли, відсутній доступ до системних папок [13].

AntTek Explorer має багато налаштувань інтерфейсу і досить велику функціональність. В останніх версіях є платною програмою, на відміну від аналогів. За відгуками користувачів, має незручний перегляд фотографій, не працююче сортування файлів за датою та часом зміни [14]. Для використання всіх функцій необхідно завантажувати та встановлювати додаткові плагіни.

У розроблюваному файловому менеджері, крім необхідного набору функцій, треба реалізувати:

- запуск файлів всіх часто використовуваних форматів;
- зручний та інтуїтивно зрозумілий інтерфейс користувача;
- можливість перегляду детальної інструкції з використання програми;
- доступ по протоколу FTP до сайтів для скачування файлів.

1.3 Постановка задачі

Метою роботи є удосконалення інформаційної технології використання мобільного терміналу для оперативного перегляду файлів різного формату.

Цей додаток має бути «електронним помічником» користувача, використовуючи який він зміг би завжди переглядати файли різного формату.

Система оперативного обслуговування користувача по перегляду файлів різного формату повинна мати простий і зрозумілий інтерфейс, в той же час володіти широкими функціональними можливостями.

Грунтуючись на дослідженнях, варто відзначити, що додаток повинен мати такі особливості:

- мобільність – можна скористатися додатком, де б користувач не знаходився і в будь-який час;
- можливість налаштування зручного перегляду будь-якого файлу в будь-який час;
- візуалізація усіх файлів в окремій папці з повним збереженням;
- можливість сортування файлів за значимістю.

1.4 Висновки до розділу 1

Виконано огляд сучасних технологій бездротової передачі даних, розглянуті можливості сучасних мобільних пристроїв. Проаналізовані характеристики існуючих програм перегляду файлів на мобільних пристроях, визначені їх недоліки та переваги. Сформульовано задачу розробки менеджера файлів для мобільної операційної системи.

1.5 Перелік джерел посилань до розділу 1

1. Орлов С. А. Теория и практика языков программирования: учебник для вузов. 2-е изд. Стандарт третьего поколения. СПб.: Питер, 2017. 688 с.
2. Мобильные технологии [Електронний ресурс] URL: <https://gsmvolna.com.ua/archives/1018> (дата звернення: 27.04.2020)

3. The Developer's Guide [Електронний ресурс] URL: <http://developer.android.com/guide/index.html> (дата звернення: 27.04.2020).
4. Global Positioning System [Електронний ресурс] URL: <http://www.navcen.uscg.gov> (дата звернення: 27.04.2020).
5. Чим 3G відрізняється від 4G, в чому різниця і що краще [Електронний ресурс] URL: <https://mobiz.com.ua/chym-3g-vidrizniaietsia-vid-4g-v-chomu-riznytsia-i-shcho-krashche.html> (дата звернення: 27.04.2020).
6. Філіпс Б., Стюарт К., Марсикано К. Android. Програмування для професіоналів. 3-е изд. СПб.: Питер, 2017. 688 с.
7. Буч Г. UML. Руководство пользования / Г. Буч, Д. Рамбо, А. Джекобсон. СПб: Питер, 2007. 630 с.
8. Спутниковый интернет Eutelsat [Електронний ресурс] URL: <https://smolsat.com/eutelsat/> (дата звернення: 27.04.2020).
9. Яценков В. С. Java за неделю: Вводный курс. Выдавна система: Ridero, 2018. 312 с.
10. Грехем И. Объектно-ориентированные методы. Принципы и практика / И. Грехем. М. : Вильямс, 2004. 880 с.
11. Описание операционной системы Android. [Електронний ресурс] URL: <http://androidu.ru/android> (дата звернення: 28.04.2020).
12. Лучшие файловые менеджеры для Android [Електронний ресурс] URL: <https://remontka.pro/android-file-managers/> (дата звернення: 28.04.2020).
13. Файловый менеджер АСТРО [Електронний ресурс] URL: <https://play.google.com/store/apps/details?id=com.metago.astro> (дата звернення: 28.04.2020).
14. AntTek Explorer [Електронний ресурс] URL: <https://play.google.com/store/apps/details?id=com.anttek.explorer.keyapp> (дата звернення: 28.04.2020).

2 АПАРАТНІ ТА ПРОГРАМНІ ЗАСОБИ РОЗРОБКИ ДОДАТКУ

2.1 Вибір апаратної платформи

Для початку необхідно визначитися, на яких пристроях буде функціонувати система. Для цього доцільно висунути ряд вимог до апаратної платформи і знайти рішення, яке найбільшою мірою їм задовольняє.

По-перше, пристрій повинен надавати оперативний і надійний доступ до додатку.

По-друге, необхідна наявність досить потужного процесора і оперативної пам'яті для здійснення необхідних обчислень і досягнення необхідної швидкодії. Вбудована пам'ять, використовувана для зберігання операційної системи, програмного комплексу і користувальницьких даних повинна бути досить великою. Бажана наявність слоту для підключення карти пам'яті з метою збільшення обсягу пам'яті.

По-третє, пристрій повинен так само володіти хорошим дисплеєм для відображення інформації в денний і нічний час. Батарея повинна тримати заряд досить тривалий час.

Всім перерахованим вимогам достатньою мірою задовольняють цифрові пристрої, звані комунікаторами або смартфонами.

Смартфони відрізняються наявністю досить розвиненої операційної системи, відкритої для розробки програмного забезпечення сторонніми розробниками. В даний час на ринку представлено цілий ряд операційних систем для мобільних пристроїв. З нього необхідно вибрати найбільш надійний та динамічно розвиваючийся варіант. Крім того, для вибраної операційної системи повинні існувати зручні і розвинені інструменти розробки додатків.

Про вибір операційної системи мова піде далі.

2.2 Вибір операційної системи

Операційна система Symbian, яка беззастережно лідирувала на ринку до 2010 року, в даний час майже повністю втратила свої позиції. Спершу Sony Ericsson оголосив про припинення використання Symbian, щоб зосередитися на випуску пристроїв під управлінням Android. Слідом за ним Samsung припиняє підтримку Symbian і переходить на використання Android, Windows Phone 7 і Bada, ОС власної розробки.

Навесні 2011 року сама Nokia оголосила про те, що Windows Phone 7 стане ключовою платформою для смартфонів, проте компанія не планує відмовлятися від платформи Symbian і стане підтримувати мобільну платформу Symbian «щонайменше» до 2022 року.

Незважаючи на це, перспектив у якісному розвитку даної операційної системи не передбачається. І, незважаючи на те, що розробка автоматизованої системи можлива і в рамках даної платформи, від цього варіанту варто відмовитися.

Apple iOS (до 2010 року відома як iPhone OS) мобільна операційна система, розроблена американською компанією Apple на основі Mac OS X спочатку для iPhone, а потім розширена для підтримки таких мобільних пристроїв, як Apple iPod, iPad і Apple TV і в даний час динамічно розвивається.

Розробка додатків для iOS ведеться в середовищі XCode, який підтримує мови C, C ++, Objective-C, Objective-C ++, Java, AppleScript, Python і Ruby з різними моделями програмування.

Apple не ліцензує iOS для установки на стороннє обладнання, а ціна комунікатора iPhone в Україні досить велика. Цей факт безумовно сильно ускладнює впровадження автоматизованої системи, розробленої для iOS з фінансової точки зору. Тому дана ОС так само не була обрана.

Набрала деяку популярність операційна система Windows Phone 10, проте вона має низку недоліків, серед яких необхідно виділити відсутність

вбудованої підтримки баз даних. SQL Server Compact не включено до складу WP10. Також не включені SQLite або будь-які інші сторонні СУБД. Відсутня так само підтримка взаємодії між процесами (IPC) і немає підтримки сокетів.

Продовжуючи аналізувати ринок операційних систем для смартфонів, можна прийти до доцільності використання ОС Android.

Операційна система Android – портативна (мережева) операційна система для комунікаторів, планшетних комп'ютерів, цифрових програвачів, наручних годинників. Спочатку розроблялася компанією Android Inc., яку потім купила Google. Згодом Google ініціювала створення альянсу Open Handset Alliance (ОНА), який зараз і займається підтримкою і подальшим розвитком платформи.

Частки ринку, які займали операційні системи в певні квартали, наведені на рис. 2.1 і в табл. 2.1.

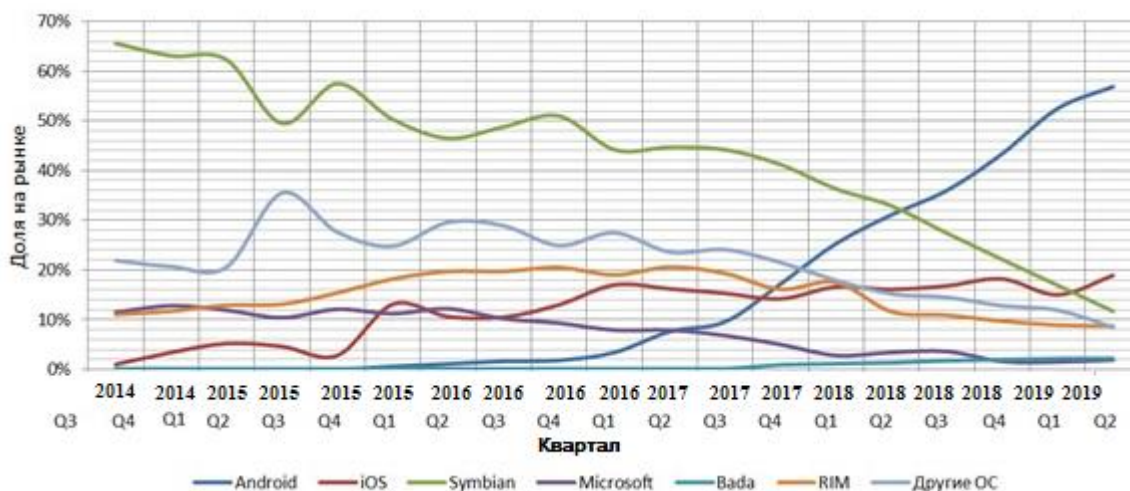


Рисунок 2.1 – Частки ринку операційних систем для смартфонів

Таблиця 2.1 – Частки ринку операційних систем для смартфонів

Квартал	Android	iOS	Symbian	Microsoft	Bada	RIM	Other
2015 Q2	0%	5,20%	62,30%	11,90%	0%	12,90%	20,60%
2015 Q3	0%	4,60%	49,50%	10,40%	0%	13,10%	35,50%
2015 Q4	0%	2,80%	57,50%	12,10%	0%	15,40%	27,60%
2016 Q1	0,60%	13,10%	50,30%	11,20%	0%	18,20%	24,80%
2016 Q2	1,10%	10,60%	46,50%	12,20%	0%	19,70%	29,60%
2016 Q3	1,60%	10,50%	48,80%	10,20%	0%	19,70%	28,90%
2016 Q4	1,80%	13,00%	51,00%	9,30%	0%	20,50%	24,90%
2017 Q1	3,40%	17,00%	44,20%	7,90%	0%	19,00%	27,50%
2017 Q2	7,60%	16,20%	44,70%	7,90%	0%	20,60%	23,60%
2017 Q3	9,60%	15,30%	44,20%	6,80%	0%	19,30%	24,10%
2017 Q4	17,20%	14,20%	41,20%	5,00%	0,90%	16,10%	21,50%
2018 Q1	25,30%	16,60%	36,30%	2,80%	1,10%	17,50%	17,90%
2018 Q2	31,10%	16,10%	32,90%	3,40%	1,30%	11,60%	15,20%
2018 Q3	36,00%	16,80%	27,40%	3,60%	1,70%	10,90%	14,50%
2018 Q4	43,40%	18,20%	22,10%	1,60%	1,90%	9,70%	12,80%
2019 Q1	52,50%	15,00%	16,90%	1,50%	2,20%	8,90%	11,90%
2019 Q2	56,90%	18,90%	11,70%	1,90%	2,10%	8,70%	8,50%

Операційна система Android має цілу низку переваг.

По-перше, це повна відкритість системи. Розробка додатків доступна всім бажаючим, і тому Android є однією з найбільш легко розширюваних платформ.

По-друге, система повністю універсальна. Android OS можна встановлювати і на смартфони, і на кишенькові комп'ютери, і навіть на телефони (у полегшеному вигляді).

Це одна з найбільш зручних і практичних мобільних операційних систем, що використовуються в сучасному світі, і її налаштування дуже просте.

Популярність операційної системи Android стала наслідком того, що провідні компанії мобільних телефонів, наприклад, HTC, Motorola, Samsung

Electronics і багато інших виробників стали використовувати Android на своїх смартфонах.

Перевагою Android від Google також є так само і те, що нова версія Android запускається майже кожні 3 місяці. Оновлення телефону новими версіями допомагає користувачам насолоджуватися більш якісними послугами.

2.2.1 Особливості архітектури ОС Android

Перед тим, як приступити до розробки програми для Android, необхідно ознайомитися з архітектурою системи та основними особливостями цієї платформи.

Система Android – це програмний стек для мобільних пристроїв, який включає операційну систему, програмне забезпечення проміжного шару (middleware), а також основні користувацькі додатки.

Архітектуру Android прийнято ділити на чотири рівні [1]:

- рівень ядра;
- рівень бібліотеки середовища виконання;
- рівень каркаса додатків;
- рівень додатків.

На рис. 2.2 показані основні компоненти операційної системи та їх взаємодія між собою.

Кожна програма в ОС Android [3] запускається у власному примірнику віртуальної машини Dalvik. Таким чином, всі працюючі процеси ізольовані від операційної системи і один від одного. І взагалі, архітектура Android Runtime така, що робота програм здійснюється строго в рамках оточення віртуальної машини. Завдяки цьому здійснюється захист ядра операційної системи від можливої шкоди з боку інших її складових. Тому код з помилками не зможе зіпсувати Android і пристрій на його базі, коли вони

спрацюють. Така захисна функція, разом з виконанням програмного коду, є однією з ключових для надбудови Android Runtime.

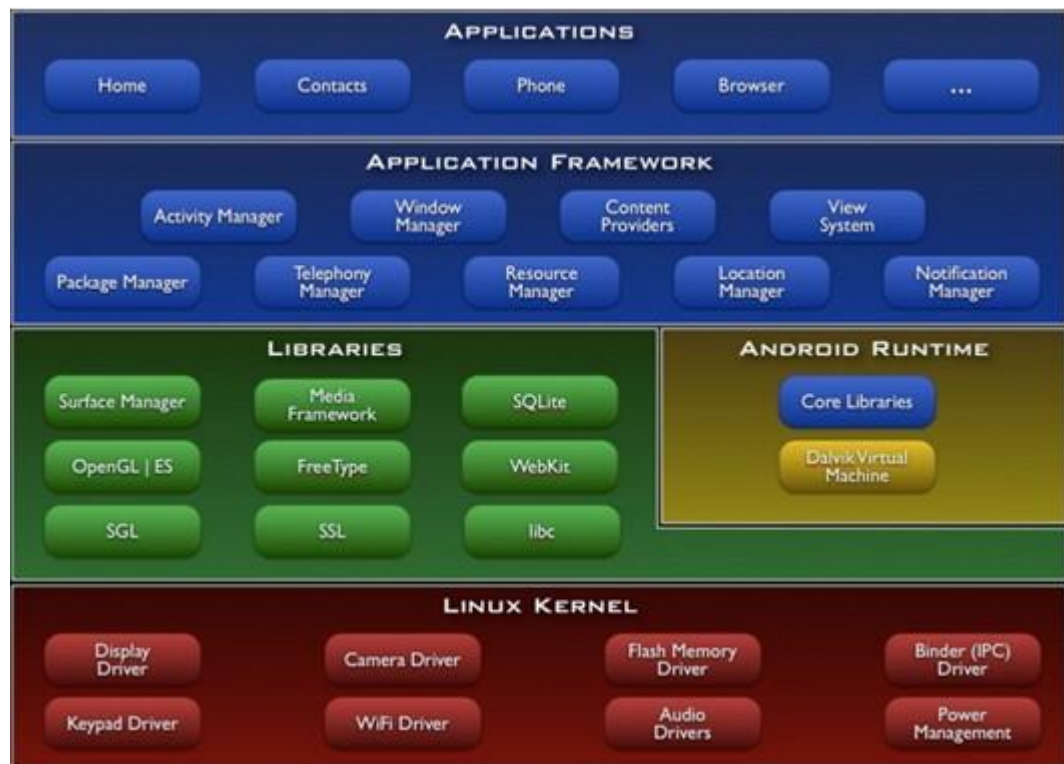


Рисунок 2.2 – Архітектура операційної системи Android

Рівнем вище розташовується Application Framework, іноді званий рівнем каркаса додатків. Саме через каркаси додатків розробники отримують доступ до API, що надаються компонентами системи, що лежать нижче рівнем. Крім того, завдяки архітектурі фреймворку, до будь-якого додатка надаються вже реалізовані можливості інших додатків, до яких дозволено одержувати доступ. У базовий набір сервісів і систем, що лежать в основі кожної програми і є частинами фреймворку, входять [2]:

- Багатий і розширюваний набір уявлень (Views), який може бути використаний для створення візуальних компонентів додатків, наприклад, списків, текстових полів, таблиць, кнопок або навіть вбудованого web-браузера.

- Контент-провайдери (Content Providers), керуючі даними, які одні програми відкривають для інших, щоб ті могли їх використовувати для своєї роботи.

- Менеджер Ресурсів (Resource Manager), що забезпечує доступ до ресурсів без функціональності (що не несуть коду), наприклад, до строкових даних, графіки, файлів та інших.

- Менеджер Оповіщень (Notification Manager), завдяки якому всі програми можуть відображати власні повідомлення для користувача в рядку стану.

- Менеджер Дій (Activity Manager), який управляє життєвими циклами додатків, зберігає дані про історію роботи з діями, а також надає систему навігації по них.

- Менеджер Місця розташування (Location Manager), що дозволяє додаткам періодично отримувати оновлені дані про поточний у географічному положенні пристрій.

Таким чином, завдяки Application Framework, додатки в ОС Android можуть отримувати в своє розпорядження допоміжний функціонал, завдяки чому реалізується принцип багаторазового використання компонентів додатків і операційної системи. Природно, в рамках політики безпеки.

На вершині програмного стека Android лежить рівень додатків (Applications). Сюди належить набір базових додатків, який встановлений на ОС Android. Наприклад, в нього входять браузер, поштовий клієнт, програма для відправки SMS, карти, календар, менеджер контактів та багато інших. Список інтегрованих програм може змінюватися в залежності від моделі пристрою та версії Android. І крім цього базового набору до рівня додатків відносяться в принципі всі програми під платформу Android, в тому числі і система, що розробляється.

2.3 Вибір засобів розробки та тестування

Вибір пристроїв-комунікаторів в якості апаратної основи для реалізації проекту і Android OS в якості операційної системи значною мірою зумовив і те, який інструментарій використовуватиметься для розробки і тестування додатків.

2.3.1 Мова Java

Система розроблялася на об'єктно-орієнтованій мові Java, розробленій компанією Sun Microsystems, яка в даний момент придбана корпорацією Oracle. Вибір даної мови був зроблений з низки міркувань.

Програми на мові Java транслюються в байт-код, що виконується віртуальною машиною (Java Virtual Mashine) - програмою, яка займається обробкою байт і передавальної інструкції обладнанню як інтерпретатор.

Гідність даного способу виконання програм полягає в повній незалежності байт-коду від операційної системи та обладнання, що дозволяє виконувати Java-додатки на будь-якому пристрої, для якого існує відповідна віртуальна машина. На комунікаторах з операційною системою Android таку функцію виконує віртуальна машина Dalvik.

Повний контроль виконання програми віртуальною машиною виливається в організацію гнучкої системи безпеки. Будь-які операції, які перевищують встановлені повноваження програми, викликають колізії при роботі з даними, викликають негайне переривання, яке може бути відповідно оброблено.

До недоліків концепції віртуальної машини можна віднести те, що виконання байт-коду віртуальною машиною в деякій мірі знижує продуктивність програм і алгоритмів, реалізованих на мові Java. Але і вплив цих особливостей зведено до мінімуму завдяки застосуванню технології трансляції байт-коду в машинний код безпосередньо під час роботи програми

(JIT-технологія) і широкому використанню переносного орієнтованого коду в стандартних бібліотеках.

Система Java створювалася об'єктно-орієнтованою з самого початку. Об'єктно-орієнтована парадигма в чомусь найбільш зручна при створенні програмного забезпечення.

Потрібно відзначити, що існує можливість розробляти програми і на C/C++ (за допомогою Native Development Kit), і на Basic (за допомогою Simple) і з використанням інших мов. Також можна створювати власні програми за допомогою конструкторів додатків, таких як App Inventor.

2.3.2 Java Development Kit

Для розробки програм на мові Java необхідно спеціальне програмне забезпечення.

Java Development Kit (JDK) - безкоштовно розповсюджуваний корпорацією Oracle Corporation комплект розробника додатків мовою Java, що включає в себе стандартні бібліотеки класів Java, приклади, документацію, різні утиліти і виконавчу систему Java (JRE).

Варто відзначити такий важливий факт, що до складу JDK не входить інтегроване середовище розробки, яка встановлюється окремо [4].

Всі сучасні інтегровані середовища розробки додатків на Java, такі, як NetBeans IDE, Sun Java Studio Creator, IntelliJ IDEA, Borland JBuilder, Eclipse, спираються на сервіси, що надаються JDK. Більшість з них для компіляції Java-програм використовують компілятор з комплекту JDK. Тому ці середовища розробки або включають в комплект поставки одну з версій JDK, або вимагають для своєї роботи попередньої інсталяції JDK на машині розробника [5].

2.3.3 Інтегроване середовище розробки Eclipse

Незважаючи на те, що для розробки можна використовувати цілий ряд сучасних інтегрованих засобів розробки (IDE), список яких наведено в попередньому підрозділі, вибір припав на Eclipse. Це відбулося з наступних причин:

Eclipse - найбільш повно документоване, вільне і доступне інтегроване середовище розробки для Java. Eclipse також дуже простий у вивченні. Це робить Eclipse дуже привабливим IDE для розробки додатків під Android.

Компанія Google випустила плагін для Eclipse - Android Development Tools, який дозволяє створювати Android-проекти, компілювати їх, і, що дуже важливо, використовувати емулятор мобільного пристрою для запуску та налагодження додатків [6].

Плагін Android Development Tools для Eclipse автоматично створює необхідну структуру Android-проекту і встановлює необхідні параметри налаштування компілятора.

2.3.4 Android SDK

Android SDK включає в себе різноманітні інструменти, бібліотеки та документацію, які допомагають розробляти мобільні додатки для платформи Android. У їх число входять:

Емулятор Android – віртуальний мобільний пристрій, який запускається на звичайному комп'ютері. Емулятор використовується для проектування, налагодження і тестування програм в реальному середовищі виконання Android.

Dalvik Debug Monitor Service (DDMS) - інтегрований з Dalvik, стандартною віртуальною машиною платформи Android, цей інструмент дозволяє керувати процесами на емуляторі або пристрої, а також допомагає у налагодженні додатків.

2.4 Висновки до розділу 2

Були проаналізовані характеристики і можливості популярних мобільних операційних систем. Для власної розробки обрано ОС Android. Її перевагою є велика популярність, а також повноцінний доступ до файлової системи, що спрощує розробку менеджера файлів. Мовою програмування обрано Java, а середовищем розробки додатку - IDE Eclipse.

2.5 Перелік джерел посилань до розділу 2

1. Дейтел П., Дейтел Х., Уолд А. Android для разработчиков. 3-е изд. СПб.:Питер, 2016. 512 с.
2. Майер Р. Android. Программирование приложений для планшетных компьютеров и смартфонов / Р. Майер. М. : Эксмо, 2011. 672 с.
3. Enabling GPS programatically in Android. [Електронний ресурс] URL: <http://stackoverflow.com/questions/5715257/enabling-gps-programatically-in-android> (дата звернення: 29.04.2020)
4. General Information on GPS [Електронний ресурс] URL: <http://www.navcen.uscg.gov> (дата звернення: 29.04.2020)
5. Кьюу Дж. Объектно-ориентированное программирование / Дж. Кьюу, М. Джеанини. Учебный курс. СПб. : Питер, 2005. 238 с.
6. Создание графических пользовательских интерфейсов. [Електронний ресурс] URL: <http://cpu.h17.ru/net/6/> (дата звернення: 29.04.2020)

3 РОЗРОБКА І ТЕСТУВАННЯ МЕНЕДЖЕРУ ФАЙЛІВ

3.1 Опис програмного продукту

3.1.1 Функціональне призначення

Додаток складається з декількох активностей, головна з яких є основною панеллю файлового менеджера. У головній активності додатку користувачеві надається вибір директорії, після чого відбувається виклик функції відображення вмісту обраної директорії. Додаток зберігає список файлів поточної і всіх вкладених директорій на випадок, якщо користувач захоче перемістити або видалити директорію.

На рис. 3 показана діаграма взаємодії основних складових частин розроблюваного менеджера файлів.

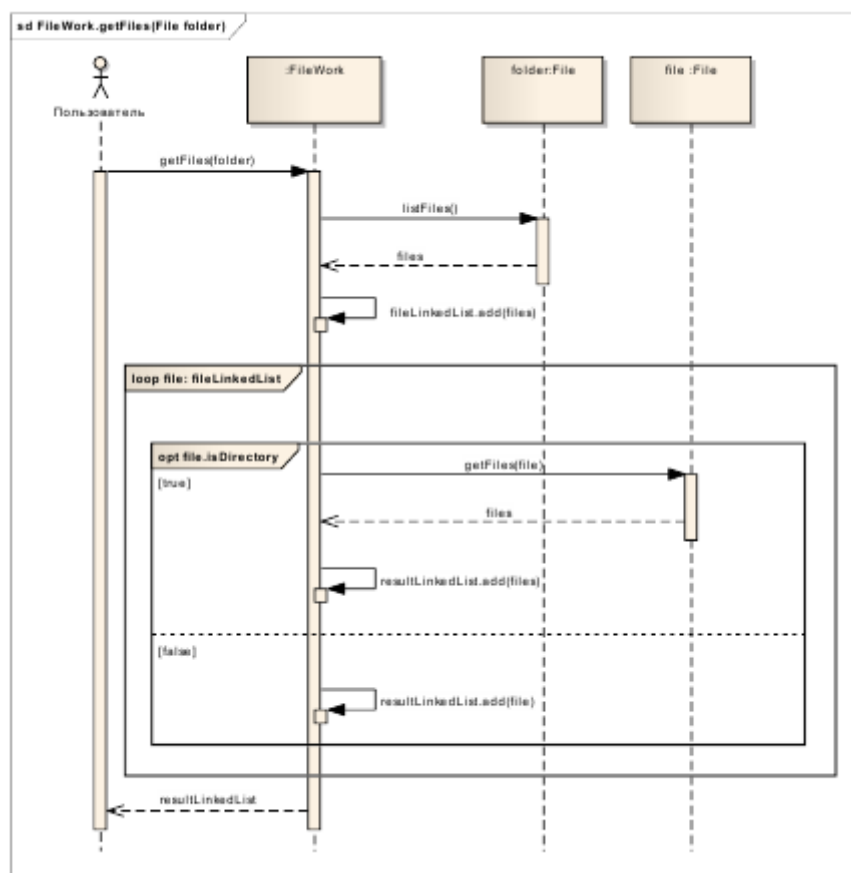


Рисунок 3.1 – Діаграма взаємодії для методу визначення дії

Перейдемо до визначення функцій, які повинен виконувати менеджер файлів, що розробляється. Для документування функціональних вимог будемо використовувати діаграми варіантів використання.

Розробка діаграми переслідує такі цілі:

- визначити загальні межі і контекст модельованої предметної області;
- сформулювати загальні вимоги до функціональної поведінки проектованої системи;
- розробити вихідну концептуальну модель системи для її подальшої деталізації у формі логічних і фізичних моделей;
- підготувати вихідну документацію для взаємодії розробників системи з її замовниками і користувачами.

Менеджер файлів має бути здатним запускати зареєстровані в ОС Android додатки для перегляду файлів заданих форматів (рис. 3.2).

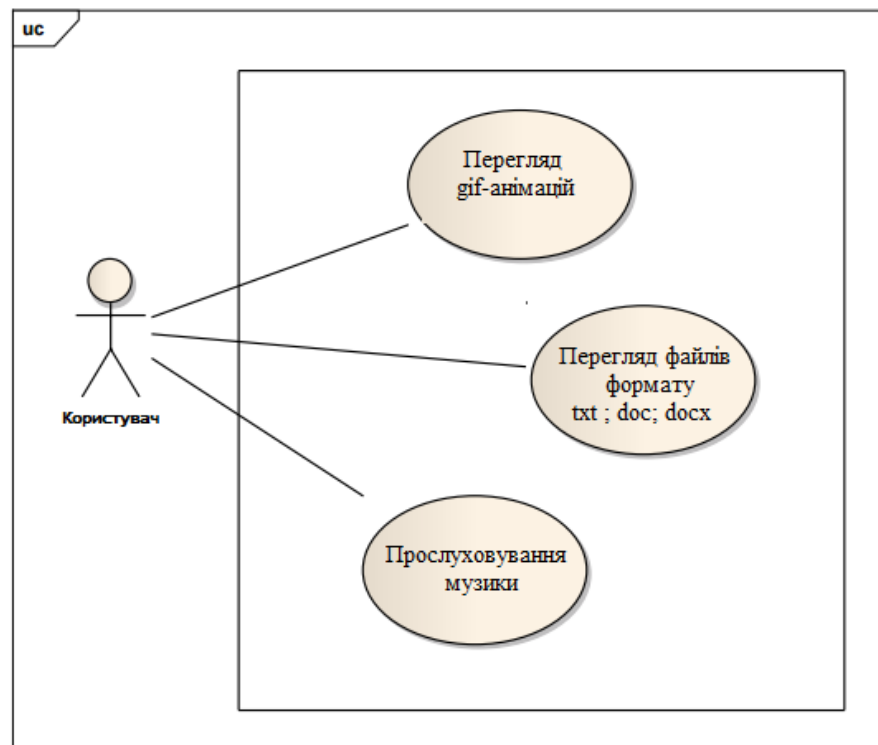


Рисунок 3.2 – Контекстна діаграма варіантів використання

Далі детально розкриємо кожну окрему дію, що має виконувати додаток, для більш повного уявлення про функціональні можливості менеджера файлів, що розробляється (рис. 3.3).

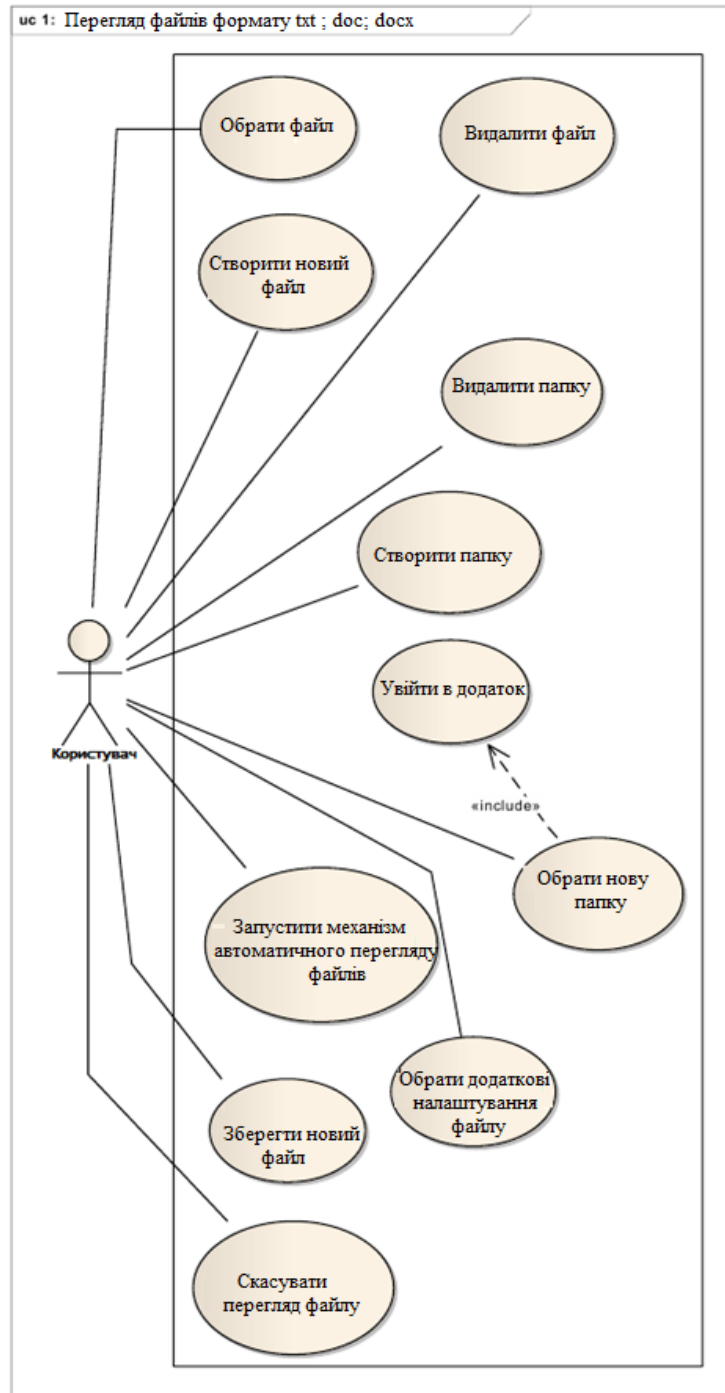


Рисунок 3.3 – Діаграма для варіанту використання «2Перегляд файлів формату txt ; doc; docx»

Використовуючи файловий менеджер, користувач повинен мати можливість переглядати файли зображень, а також gif-анімацію (рис. 3.4).

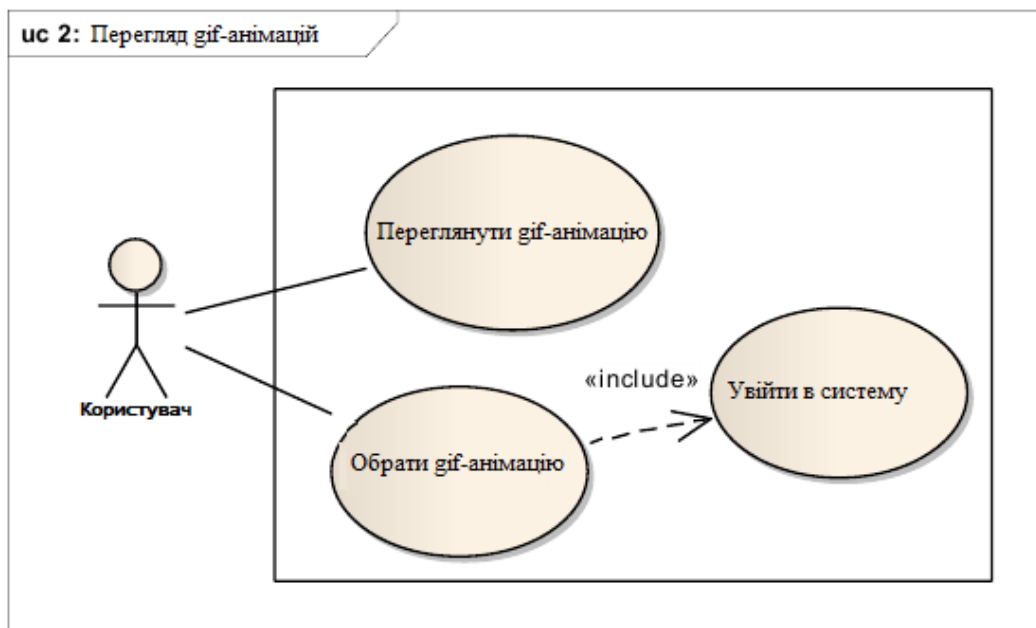


Рисунок 3.4 – Діаграма для варіанту використання «Перегляд gif-анімацій»

Окремий варіант використання позначається на діаграмі еліпсом, усередині якого міститься його коротка назва або ім'я у формі дієслова з пояснювальними словами.

Мета варіанту використання полягає в тому, щоб визначити закінчений аспект або фрагмент поведінки деякої сутності без розкриття її внутрішньої структури. В якості такої сутності може виступати система або будь-який елемент моделі, який володіє власною поведінкою.

Кожен варіант використання відповідає окремому сервісу, який надає сутність, що моделюється за запитом актора, тобто визначає спосіб застосування цієї сутності. Сервіс, який ініціалізується за запитом актора, є закінченою неподільною послідовністю дій.

При відкритті будь-якого файлу запускається додаток зі списку «Програми за замовчуванням» ОС Андроїд, зокрема, встановлений в ОС аудіоплеєр (рис. 3.5).

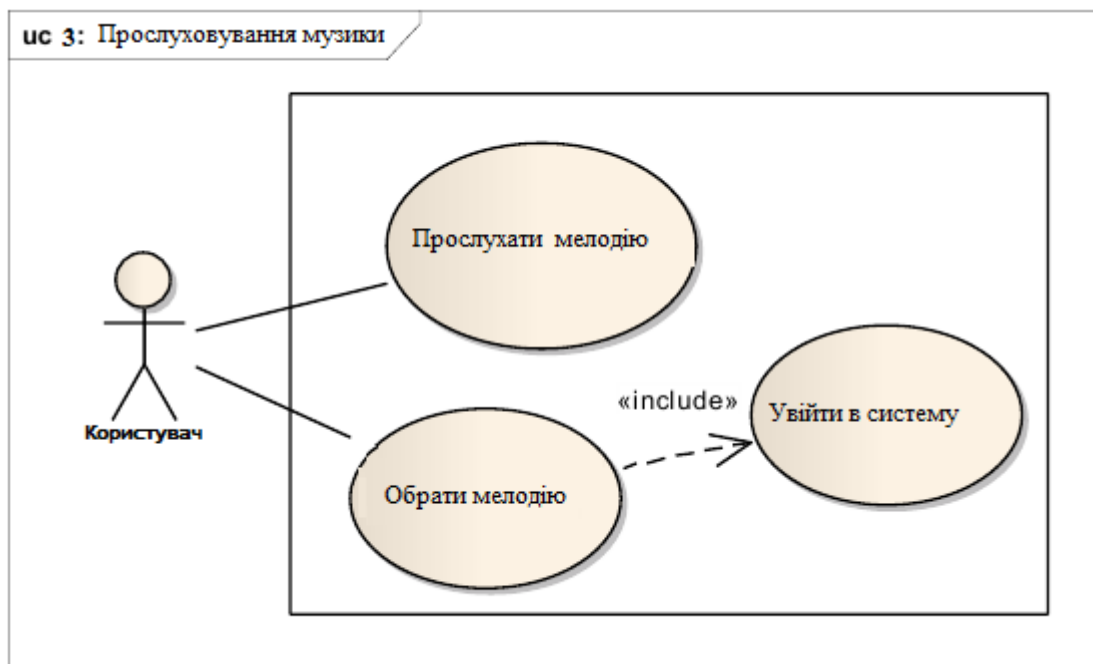


Рисунок 3.5 – Діаграма для варіанту використання «Прослуховування музики»

3.1.2 Вхідні та вихідні дані

Програми для Android складаються з компонентів, які система може запускати і керувати так, як їй необхідно [1]. Для цього система повинна бути в змозі запустити процес для програми, в якому знаходяться необхідні компоненти, і ініціалізувати потрібні їй об'єкти. Одним з компонентів Android-додатку є діяльність (activity).

Activity являє собою візуальний інтерфейс (окремий екран) для однієї дії, яку користувач може зробити [2].

Додаток може складатися з одного activity або з декількох. Це залежить від типу додатка і його дизайну. Одне activity може викликати інше.

Кожне activity задає вікно для відображення, яке, зазвичай, займає весь екран, але може бути менше і плавати поверх інших вікон. Activity може використовувати додаткові вікна, наприклад, спливаючий діалог, який

вимагає проміжної відповіді користувача, або вікно, яке відображає користувачам важливу інформацію при виборі елемента, вартого особливої уваги.

Візуальний інтерфейс будується на основі ієрархії візуальних компонентів – об'єктів, похідних від базового класу View. Android має ряд готових до використання компонентів, включаючи кнопки, текстові поля, смуги прокрутки, меню, прапорці і багато іншого [3].

3.1.3 Логічна структура програми та її модулів

Загальна структура головно вікна [4] з виводом основних функцій описується так (рис. 3.6):

```
<?xml version="1.0" encoding="utf-8" ?>
<TableLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent" >

    <TableRow >
    <TextView android:id="@+id/titleManager"
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:padding="5dip"
    />
    </TableRow >
    <TableRow >
    <ListView android:id="@id/android:list"
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:layout_weight="2"
        android:drawSelectorOnTop="false" />
    </TableRow >
</TableLayout >
```

Рисунок 3.6 – xml-розмітка головної активності додатка

Тут задається розмітка для основного Layout'a форми [5]. `TableLayout` тут означає, що елементи у нас збудовані у вигляді таблиці. Далі у верхній комірці таблиці розміщується елемент `TextView` (текстове поле), а в нижній клітинці - `ListView` (список). Обидва елементи мають `id`, використовуючи який, ми можемо змінювати вміст елементів. Наприклад, використовуючи `R.id.titleManager` для нашого текстового поля `TextView` (рис. 3.7).

```
<?xml version="1.0" encoding="utf-8" ?>
<TextView
xmlns:android="http://schemas.android.com/apk/res/android"
android:layout_width="fill_parent"
android:layout_height="40sp"
android:padding="5dip"
android:gravity="center_vertical"
/>
```

Рисунок 3.7 – Елемент для позначення окремого файлу

Далі задаємо розмітку для кожного елемента нашого `ListView`, тобто безпосередньо для кожної окремо взятої папки або кожного файлу. В даному коді у нас задається ширина кожного елемента, висота, відступ (`padding`) і вирівнювання `center_vertical` - тобто центрування по вертикалі (лістинг коду в додатку). На початку вказується назва пакета (`package name`) (рис. 3.8).

```
package ru.alwake.filemanager;
import java.io. File;
import java.util. ArrayList;
import java.util. List;

// browse to file or directory
private void browseTo (final File aDirectory) {
// if we want to browse directory
if (aDirectory.isDirectory ()) {
// fill list with files from this directory
this .currentDirectory = aDirectory;
fill (aDirectory.listFiles ());
```

Рисунок 3.8 – Відображення файлів заданної директорії

Рядки 2-4 відповідають за імпорт потрібних нам бібліотек. Важливо відзначити, що імпорт бібліотек Eclipse може виробляти автоматично, як тільки зустрінеться що-небудь невідоме.

3.1.4 Організація інтерфейсу з користувачем

Оскільки, згідно технічному завданню, інтерфейс користувача має бути максимально зрозумілим, легким та простим у використанні, було вирішено розбити головне вікно на дві однакові панелі для відображення з правої сторони тільки папок, що зберігаються на пристрої, а з лівої сторони файлів, що містяться у вибраній папці; навігаційну панель розміщено внизу.

Результати проектування інтерфейсу додатку, що розробляється, наведено на рис. 3.9.

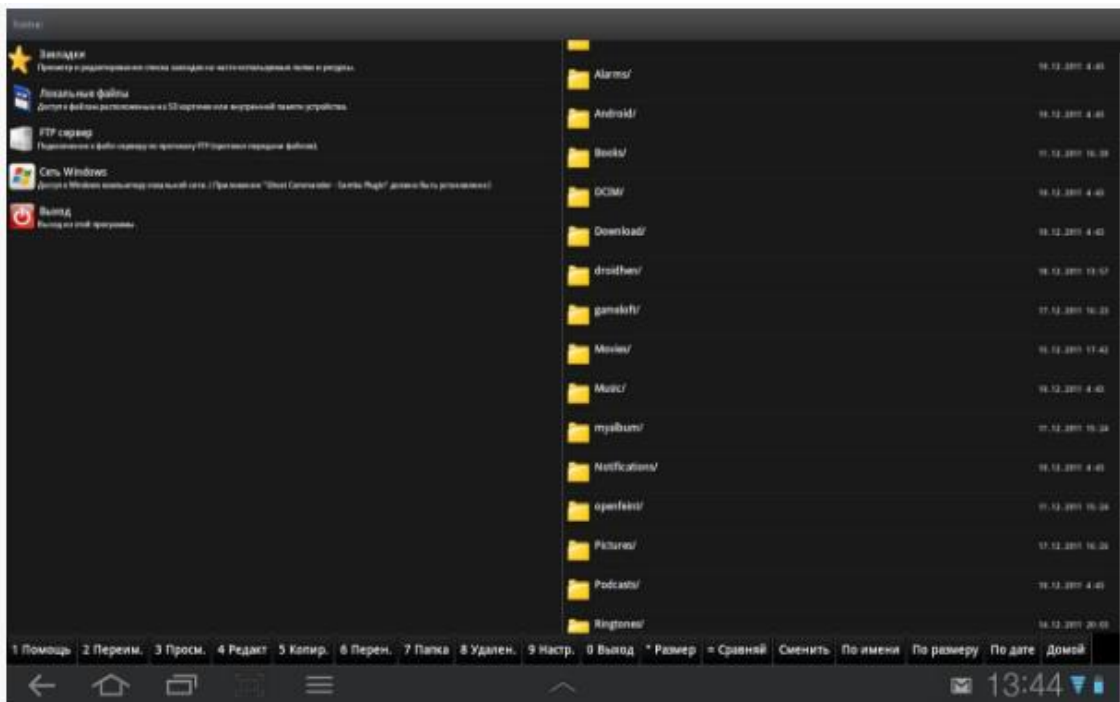


Рисунок 3.9 – Інтерфейс менеджера файлів

3.5 Тестування та аналіз результатів

Проведемо тестування розробленого додатку з метою виявлення його слабких сторін та усунення можливих недоліків, якщо такі будуть виявлені.

Тестування проводимо за всіма можливими рівнями використання додатку. Результати тестування наведені на рис. 3.10-3.13. На рис. 3.10 показаний перегляд основної інформації про обраний файл.

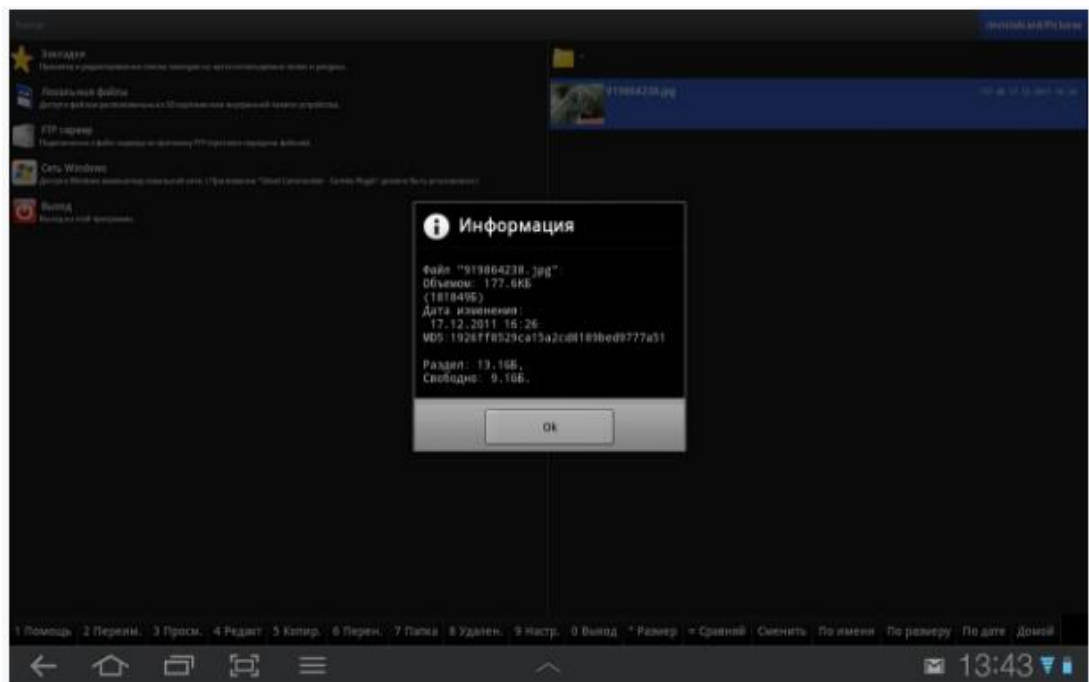


Рисунок 3.10 – Основні параметри файлу

На рис. 3.11 показана коротка інструкція по використанню програми. Приведено перелік дій, що виконуються розробленим менеджером файлів:

- виділити поточний файл;
- перейти на іншу панель;
- додати поточну директорію в закладки;
- показати в іншій панелі цю ж директорію;
- перейменувати поточний файл;
- редагувати поточний файл;

- скопіювати файли або директорії;
- перенести файли директорії;
- створити директорію;
- видалити виділене.

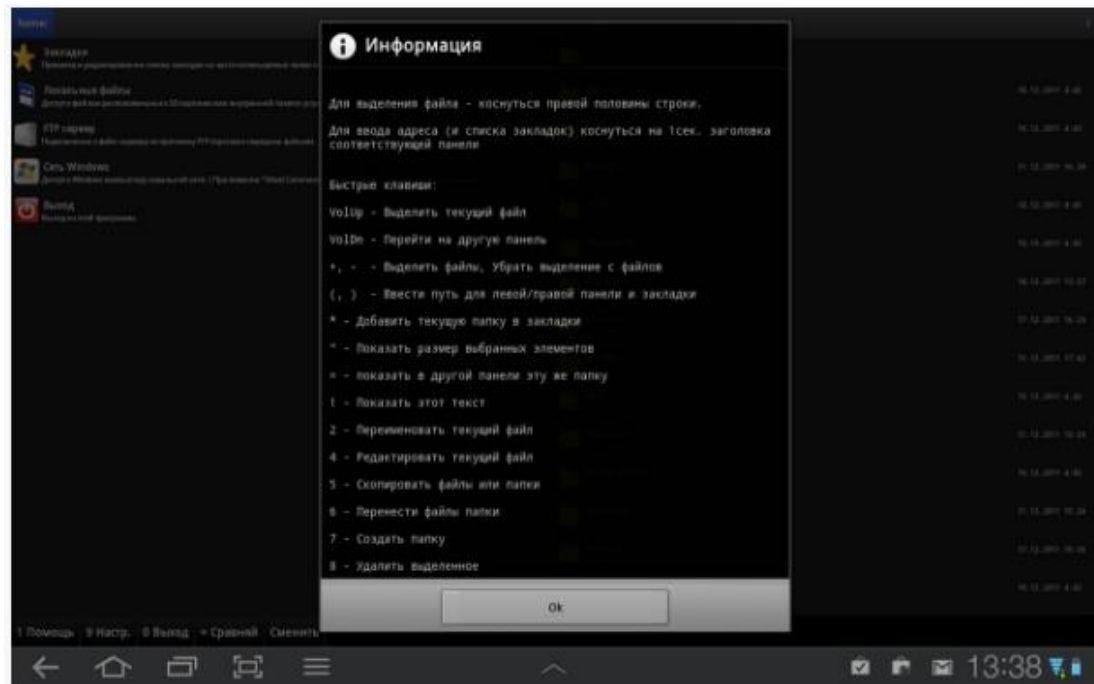


Рисунок 3.11 – Інструкція користувача

На рис. 3.12 показаний діалог вибору програми для запуску файла. Таке відбувається, коли в операційній системі для файлів даного формату відсутня програма для запуску за замовчуванням.

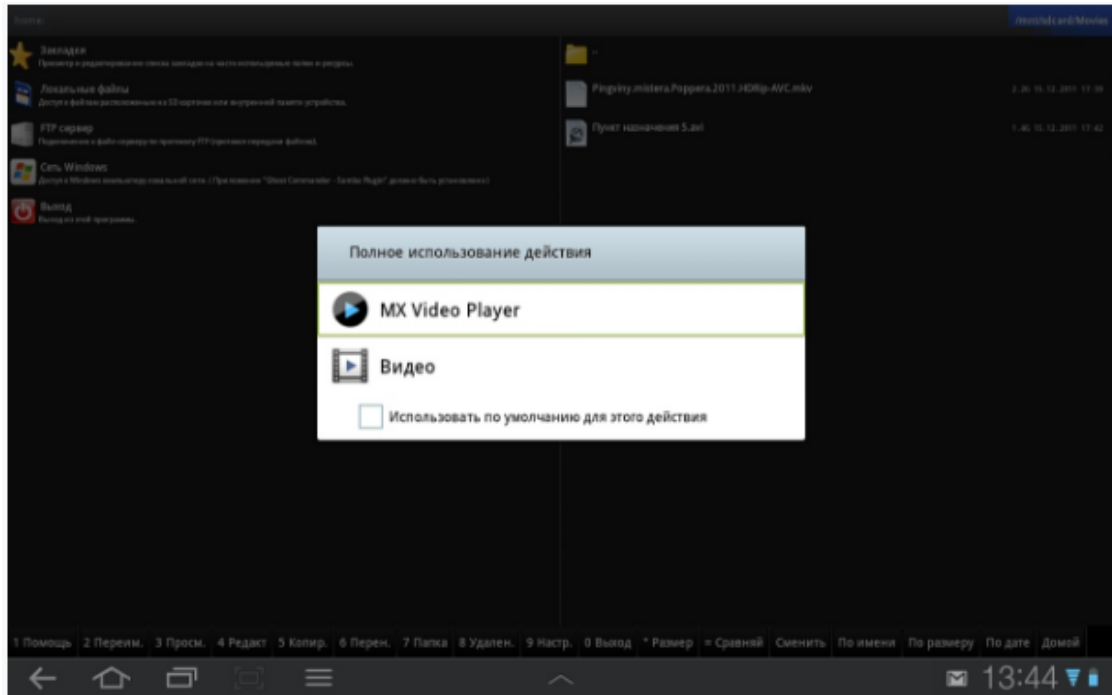


Рисунок 3.12 – Вибір програми для запуску

Далі продемонстрована можливість підключення до ftp-серверу (рис. 3.13) і контекстне меню для вибраного файлу або директорії (рис. 3.14).

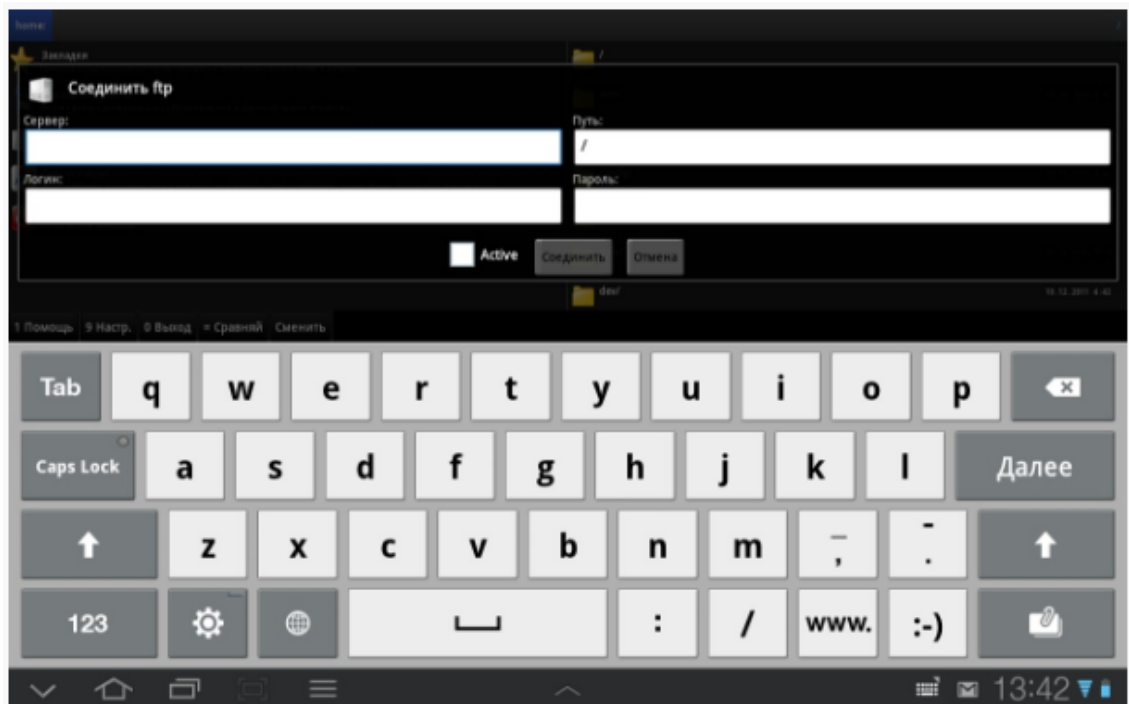


Рисунок 3.13 – Тестування програмного продукту

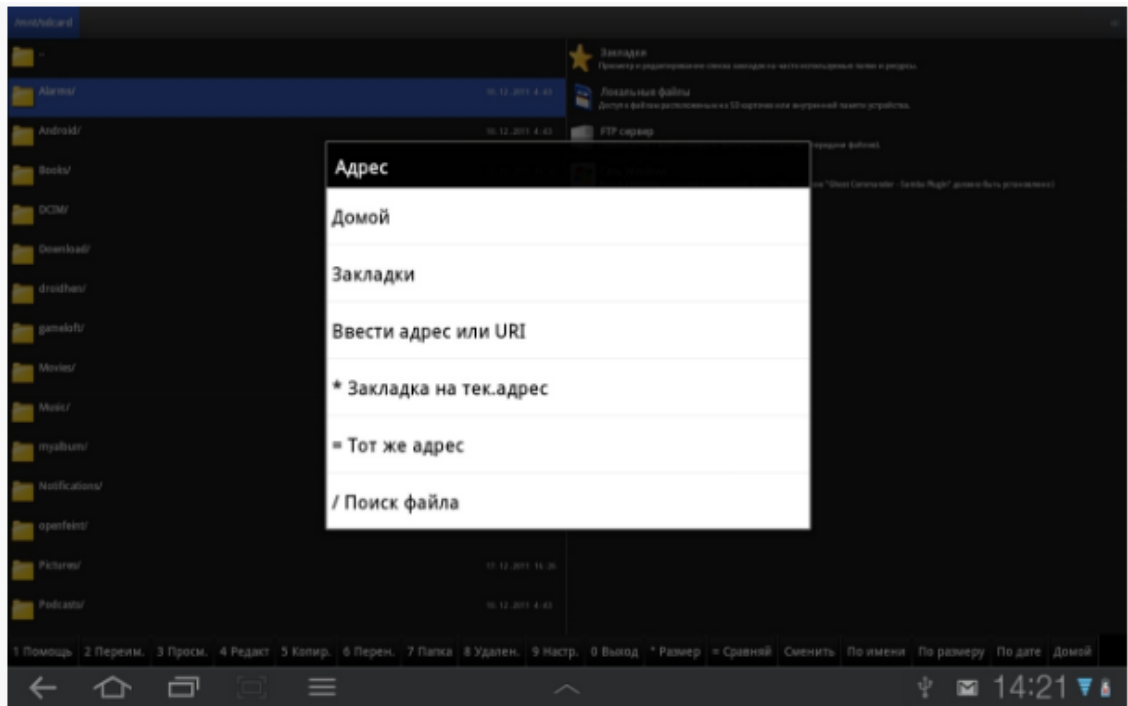


Рисунок 3.14 – Тестування програмного продукту

Під час тестування збоїв та недоліків у роботі додатку не виявлено, що говорить про достатню якість розробки та можливість впровадження її у використання.

3.6 Встановлення та налаштування програми

Для запуску програми треба встановити файл .apk та натиснути на відповідному значку. На екрані з'явиться головне вікно програми (рис. 3.12).

Потім необхідно за допомогою пошуку знайти файл, який необхідно переглянути (рис. 3.15), на екрані з'явиться перелік всіх файлів даного формату (рис. 3.16)

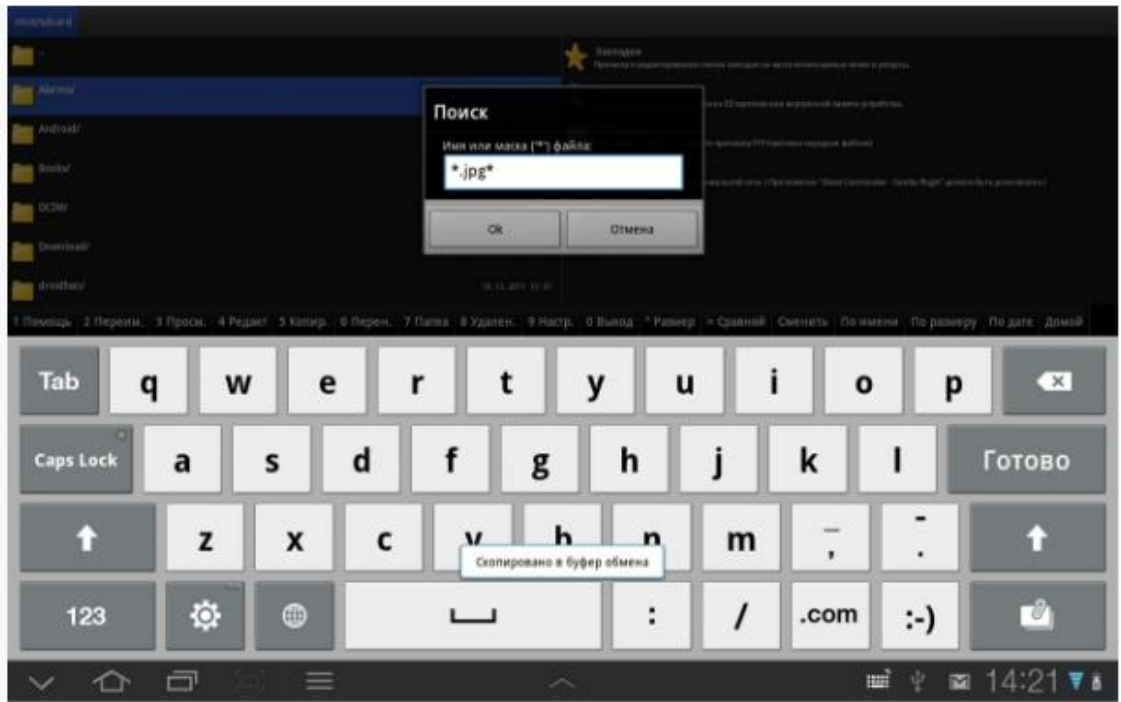


Рисунок 3.15 – Пошук файлу

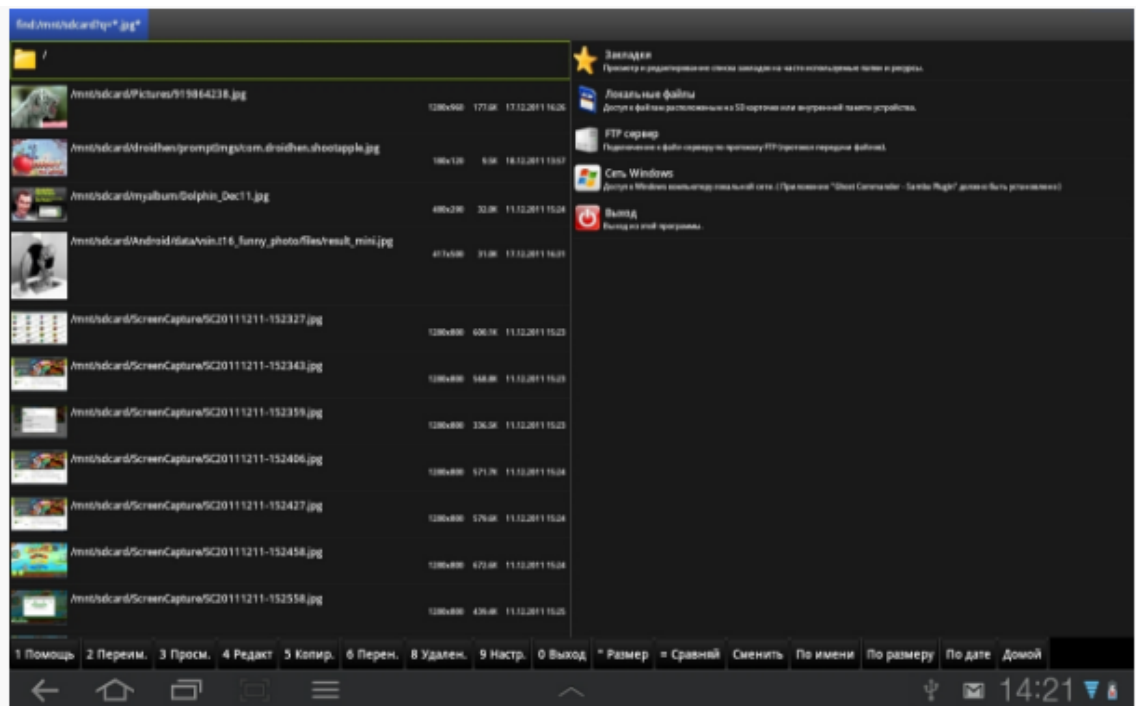


Рисунок 3.16 – Результат пошуку файлу

Налаштування параметрів виводу необхідної інформації відбувається за вказаними налаштуваннями (рис. 3.17, рис. 3.18).

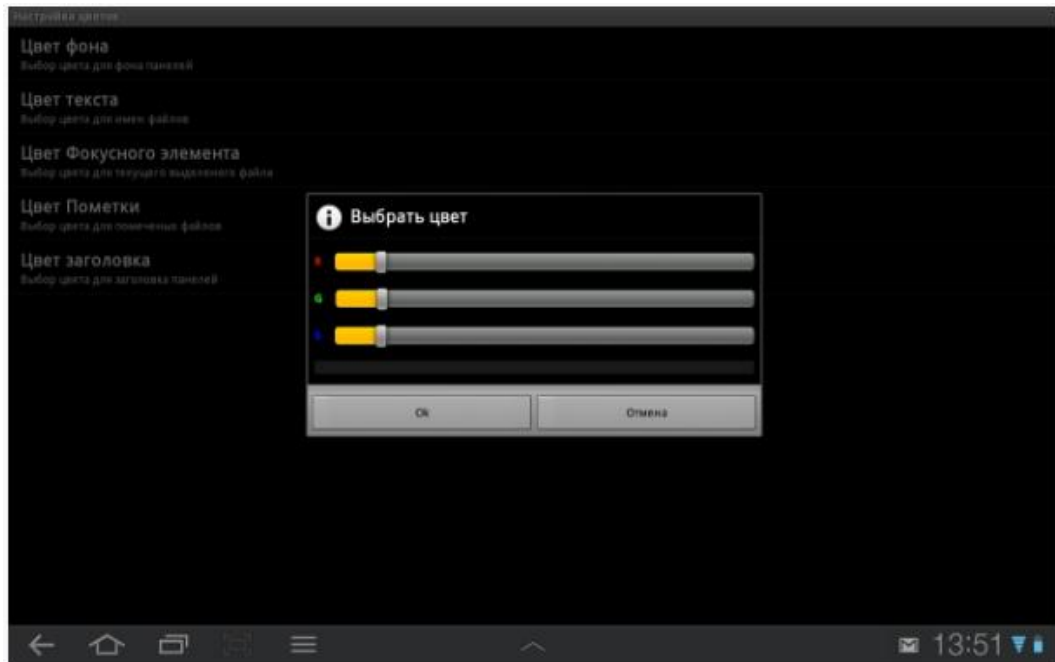


Рисунок 3.17 – Результат налаштування показу



Рисунок 2.18 – Результат налаштування показу

3.7 Висновки до розділу 3

Був спроектований і реалізований додаток для перегляду файлів в ОС Android. Він має функції перегляду директорій і файлів на мобільному пристрої, запуску файлів різних типів за допомогою встановлених програм. Додаток легко встановити і налаштувати. Розроблений менеджер файлів має інтуїтивно зрозумілий інтерфейс. В додатку передбачена інструкція з використання з переліком "швидких клавіш".

3.8 Перелік джерел посилань до розділу 3

1. Орлов С. А. Теория и практика языков программирования: учебник для вузов. 2-е изд. Стандарт третьего поколения. СПб.: Питер, 2017. 688 с.
2. Грехем И. Объектно-ориентированные методы. Принципы и практика / И. Грехем. М. : Вильямс, 2004. 880 с.
3. Описание операционной системы Android. [Электронный ресурс] URL: <http://androidu.ru/android> (дата звернення: 05.05.2020)
4. Томас Х. Алгоритмы: построение и анализ / Х. Томас, Ч. И. Лейзерсон, Р. Л. Ривест, К. Штайн. – М. : Вильямс, 2006. – 1296 с.
5. Левитин А. В. Алгоритмы: введение в разработку и анализ / А. В. Левитин. – М. : Вильямс, 2006. – 1296 с.
6. Аргерих Л. Профессиональное PHP программирование / Л. Аргерих, В.Чой, Д. Коггсхол. – СПб. : Символ-Плюс, 2003 г. – 1048 с.

4 ОХОРОНА ПРАЦІ

4.1 Поняття охорони праці

Сучасні наукові дослідження розглядають охорону праці як економічну, соціальну та правову категорії. На сьогодні, охорону праці як правову категорію розглядають у широкому та вузькому значенні. При вживанні терміну «охорона праці» відповідно до його значення, тобто в широкому розумінні, до його поняття відносять ті гарантії для працівників, що передбачають усі норми трудового законодавства, наприклад норми, що забороняють власнику або уповноваженому ним органу звільнити працівників з роботи тоді, коли немає підстав, передбачених ст.ст. 40, 41 КЗпП [1]. Ці норми становлять надзвичайно важливу гарантію, спрямовану на охорону трудових прав працюючих. Або така гарантія, як можливість розірвання трудового договору з ініціативи власника або уповноваженого ним органу тільки за попередньою згодою профспілкового комітету (ст. 43 КЗпП).

Проте термін «охорона праці» в чинному трудовому законодавстві вживається не в такому широкому, тобто буквальному, значенні цих слів, а в більш вузькому. У вузькому значенні під охороною праці розуміється сукупність заходів щодо створення безпосередньо в процесі роботи нормальних і безпечних технічних і санітарно-гігієнічних умов для всіх працюючих.

Основними нормативними документами, що забезпечують діяльність кожного працівника, можна поділити на декілька частин, а саме:

- законодавчі нормативні документи;
- внутрішні нормативні документи.

Тобто використання цього комплексу законодавчих та внутрішніх нормативних документів забезпечують працівникам ефективну систему охорони праці.

Законодавчу діяльність з охорони праці забезпечується Кодексом законів про працю, у якому чітко та послідовно викладені основні права працівників, а саме на відпочинок, на умови праці, на винагорода за роботу тощо. Окрім КзПП є ще Закон «Про охорону праці» [2], Закону України «Про загальнообов'язкове державне соціальне страхування від нещасного випадку на виробництві та професійного захворювання, які спричинили втрату працездатності» [3], Конституція [4] та інші документи.

В Україні прийнято ряд нормативно-правових актів, які регулюють порядок опрацювання, прийняття, перегляду та скасування державних нормативних актів про охорону праці (далі — ДНАОП), принципи їх прийняття та побудови.

Тобто з точки зору законодавчої бази кожен працівник має право у своїй роботі використовувати ці законодавчі акти, які безперечно захищають працівників у їх трудовій діяльності.

4.2 Охорона праці при роботі з комп'ютером

4.2.1 Загальні вимоги безпеки

Вимоги до мікроклімату і іонізації повітряного середовища:

У виробничих приміщеннях, в яких робота на ПК є основною, повинні забезпечуватися оптимальні параметри мікроклімату. Для підтримки оптимальних параметрів мікроклімату рекомендується установка кондиціонера;

Рівні позитивних і негативних аероіонів в повітрі приміщення з ПК повинні відповідати ГОСТ 12.1.005-88. ССБТ. Воздух рабочей зоны. Общие требования. – Введ. 01.07.77 [5].

Вимоги до шуму:

У приміщеннях, обладнаних ПК, рівень шуму на робочому місці не повинен перевищувати 50 дБ «А».

Вимоги до випромінювання:

Всі термінали повинні мати гігієнічний сертифікат, що включає, у тому числі оцінку візуальних параметрів;

Основним джерелом електромагнітних випромінювань від моніторів ПК є трансформатор високої частоти рядкової розгортки;

Конструкція монітора ПК повинна забезпечувати захист від рентгенівського випромінювання, доза якого не повинна перевищувати 0,1 мбер/год (100 мкр/год) в будь-якій точці на відстані 5 см від екрану і корпусу монітора;

Вимоги до напруженості електромагнітного поля (на відстані 50 см навколо ПК) по електричному, до щільності магнітного потоку індукції; Поверхневий електростатичний потенціал не повинен перевищувати 500 В.

Вимоги електробезпеки:

До роботи на комп'ютері і периферійному обладнанні допускаються особи, які вивчили інструкцію з охорони праці при роботі на ПК, які пройшли інструктаж з охорони праці та поставили розпис в журналі інструкцій;

Електроживлення ГТК і друкуючих пристроїв здійснюється від однофазної мережі змінного струму напругою 220В і частоти коливань 50 Гц;

Комп'ютер повинен бути заземлений.

4.2.1.1 Розрахунок рівня шуму в приміщенні

Шум - це неправильне, хаотичне, аперіодичне поєднання різних звуків, відмінних за силою і частотою коливань. Як фізичне явище звук являє собою хвилеподібні механічні коливання частинок пружного середовища. При коливанні тіла в повітрі останнє поперемінно то густішає то розріджується,

тобто змінюється тиск, внаслідок чого виникає звукова хвиля, яка поширюється у просторі.

Залежно від шкідливості дії і частотного складу всі шуми поділяють на низько-, середньо-, й високочастотні. Низькочастотними вважають шуми, у спектрі яких переважають частоти 250-300 гц. Середньочастотний шум характеризується переважно частотою в діапазоні 300-800 Гц. Усі шуми з найбільшою інтенсивністю на частотах понад 800 Гц є високочастотними.

Численними експериментами і клінічними спостереженнями встановлено, що тривала дія шуму на організм людини несприятливо позначається не тільки на слуховому аналізаторі, а й на центральній нервовій системі, призводить до порушення функції серцево-судинної системи, травного каналу, органів чуття та ін. Шум підвищує нервову напруженість, знижує психічні й зорові реакції, є причиною швидкого стомлення людини, знижує продуктивність праці. При дії шуму інтенсивністю 90-100 дБ «А» знижуються зорова реакція (на 25%), пильність, затримуються психічні реакції, з'являється головний біль і запаморочення, прискорюється процес стомлення, порушуються пульс і дихання. Усе це може бути причиною появи виробничого травматизму.

Інтенсивний шум несприятливо діє на вищі відділи центральної і периферичної нервової системи. При цьому слід зауважити, що шум негативно впливає на нервову систему навіть тоді, коли не спостерігається помітного впливу на органи слуху і він не є суб'єктивно неприємним.

Санітарними й гігієнічними нормами ДСН 3.3.6.037-99 [6] встановлено гранично допустимий рівень звуку і рівень звукового тиску при середньгеометричних частотах октавних смуг. Граничний шум не повинен заважати нормальній роботі. У таблиці 4.1 наведено гранично допустимі рівні для шуму з безперервним спектром, який діє на людину більше 4 годин в зміну, та називається рівнем звуку і вимірюється за шкалою шумоміра «А» у децибелах «А».

Таблиця 4.1 – Гранично допустимий рівень звуку на робочих місцях

Найменування	Рівень звуку, дБА
Приміщення для конструкторських бюро, програмістів, лабораторій теоретичних робіт	50
Приміщення управління, робочі кімнати	60
Приміщення лабораторії, для проведення експериментальних робіт, приміщення для розміщення шумних машин	80
Постійні робочі місця і робочі зони у виробничих приміщеннях та на території виробництва	85

В приміщенні, де розробляється програмне забезпечення, встановлено 5 однакових комп'ютерів, які є джерелами шуму з рівнем 30 дБА кожний. Також встановлено два вентилятора з рівнем шуму 45 дБА.

Як відомо, сумарний рівень шуму однакових джерел, може бути розрахований за формулою:

$$P_{\text{сум}} = P + 10 \lg N \quad (4.1)$$

де P – рівень шуму від одного джерела, дБА;

N – кількість однакових джерел шуму, шт.

Але в нашому випадку потрібно розрахувати сумарний рівень шуму під час роботи джерел з різним рівнем шуму кожного (точніше групи однакових джерел). Розрахунок можна провести за такою формулою:

$$L_y = 10 \lg(N * 10^{0.1P} + L * 10^{0.1D}) \quad (4.2)$$

де P – рівень шуму початкових джерел шуму, дБА;

N – кількість однакових джерел шуму, шт;

D – рівень шуму від додаткових джерел, дБА;

L – додатково встановлені джерела шуму, шт.

Отже, сумарний рівень шуму в приміщенні буде складати:

$$L_y = 10 \lg(5 \cdot 10^{0.1 \cdot 30} + 2 \cdot 10^{0.1 \cdot 45}) = 48.34 \text{ дБА}$$

Як бачимо з таблиці 4.1 рівень шуму не перевищує прийнятої норми для даного класу приміщення, отже шумоподавлюючі засоби використовувати не потрібно.

4.2.1.2 Розрахунок потрібного повітрообміну в приміщенні

Вентиляція являє собою сукупність заходів та засобів призначених для забезпечення на постійних робочих місцях та зонах обслуговування виробничих приміщень метеорологічних умов та чистоти повітряного середовища, що відповідають гігієнічним та технічним вимогам. Основне завдання вентиляції – вилучити із приміщення забруднене або нагріте повітря та подати свіже.

У відповідності з санітарними нормами в приміщенні з об'ємом менше 20 м³/чол. повітрообмін повинен бути не менше 30 м³/год. на кожного робітника. Це є допустимою нормою.

Оптимальна норма повітрообміну визначається наступним розрахунком:

1. Для вилучення вуглекислоти, що виділяється при диханні

$$L_{CO_2} = \frac{A \cdot q_e}{q_1 - q_2}, \text{ м}^3/\text{ГОД} \quad (4.3)$$

2. Для вилучення вологи

$$L_{H_2O} = \frac{A \cdot B}{0.01 \cdot P_{max} (O_A - O_C)}, \text{ м}^3/\text{ГОД} \quad (4.4)$$

3. Для вилучення тепла

- а) при умові переходу від організму людини до повітря:

$$L'_a = \frac{A \cdot Q_{\text{я}}}{0.24 \cdot \gamma (t_{\text{пш}} - t_{\text{п}})}, \text{ м}^3/\text{ГОД} \quad (4.5)$$

б) при умові вилучення тепла з приміщення

$$L''_a = \frac{A(Q_{\text{я}} + Q_{\text{пр}})}{\gamma(t_3 - t_{\text{н}})}, \text{ м}^3/\text{ГОД} \quad (4.6)$$

У цих формулах:

A – кількість одночасно працюючих людей;

q_e, q_1, q_2 – кількість CO_2 виділяемого людиною, допускаемого у приміщенні, яке є у складі припливі повітря;

P_{max} – максимальний склад водяного пару у повітрі;

B – кількість вологи, виділяемої організмом людини;

O_A – допустима вологість у приміщенні;

O_C – вологість зовнішнього повітря;

$t_{\text{пш}}, t_{\text{п}}$ – температура поверхні шкіри, повітря;

t_3, t_b - тепло складова повітря, яке залишає приміщення і яке надходить у приміщення;

$Q_{\text{я}}$ – явне тепло (яке виділяється людиною за відрахуванням тепла, витраченого на випарювання)

$Q_{\text{пр}}$ тепло від електроприладів

$$Q_{\text{пр}} = \varphi \cdot W, \text{ Вт} \quad (4.7)$$

Де φ – коефіцієнт для електроприладів – 0,9;

W - потужність електроприладів, кВт

Визначимо необхідний повітрообмін для підтримання оптимального режиму в приміщенні операторів ПК. Кількість працюючих у приміщенні – 5 чоловік. Умови праці – легкі. Район місцевості – м. Київ, період року – літо.

Використовуючи нормативні дані вибираємо:

$$q_1 = 1 \text{ л/м}^3; q_2 = 0,5 \text{ л/м}^3; q_e = 25 \text{ л/м}^3;$$

$$Q_{\text{я}} = 84 \text{ Вт}; B = 110 \text{ г/год.}; P_{\text{max}} = 19,66 \text{ г/м}^3$$

$$t_3 = 29,4^\circ\text{C}; t_{\text{в}} = 20^\circ\text{C}; t_{\text{пш}} = 31^\circ\text{C}; t_{\text{п}} = 22^\circ\text{C}$$

Визначаємо потрібний повітрообмін:

а) для вилучення надмірної вологи, щоб склад водяного пару не перевищував 60%:

$$L_{\text{H}_2\text{O}} = \frac{110 \cdot 5}{0,01 \cdot 19,66 (60 - 40)} = 139,87, \text{ м}^3/\text{год}$$

б) для вилучення надмірної кількості CO₂:

$$L_{\text{CO}_2} = \frac{5 \cdot 25}{1 - 0,5} = 250, \text{ м}^3/\text{год}$$

в) для вилучення тепла з приміщення при умові переходу від організму людини до повітря:

$$L'_a = \frac{5 \cdot 84}{0,24 \cdot (31 - 22)} = 194,44, \text{ м}^3/\text{год}$$

г) для вилучення тепла з приміщення

$$L''_a = \frac{5 (84 + (0,9 \cdot 430))}{29,4 - 20} = 250,53 \text{ м}^3/\text{год}$$

4.2.2 Вимоги безпеки перед початком роботи

1. Перед початком роботи з ПК працівник зобов'язаний:

- провітрити робоче приміщення;
- перевірити стійкість положення обладнання на робочому столі, відсутність видимих пошкоджень обладнання, дискет, справність живильних і сполучних кабелів, роз'ємів і штепсельних з'єднань, захисного заземлення, справність меблів;

- відрегулювати положення стільця, підставки для ніг, клавіатури, екрану монітора, оцінити освітленість на робочому місці. При необхідності включити місцеве освітлення.

2. Протерти поверхню екрана монітора, захисного фільтра сухою м'якою тканинною серветкою;

3. Переконатися у відсутності відбиттів на екрані монітора, зустрічного світлового потоку.

4. Включити устаткування ПК в електричну мережу, дотримуючись такої послідовності:

- блок безперебійного живлення;
- периферійні пристрої (принтер, монітор, сканер і інші пристрої);
- системний блок.

5. Забороняється приступати до роботи при:

- виражене тремтіння зображення на моніторі;
- виявлення несправності обладнання;
- наявність пошкоджених кабелів або проводів, роз'ємів, штепсельних з'єднань;
- відсутності або несправності захисного заземлення обладнання.

4.2.3 Вимоги безпеки під час роботи

1. Під час роботи з ПК працівник зобов'язаний:

- дотримуватися вимог охорони праці, встановлені цією інструкцією;
- містити в порядку і чистоти своє робоче місце;
- тримати відкритими вентиляційні отвори обладнання;
- дотримуватися оптимальна відстань від екрана до очей.

2. Роботу за екраном монітора слід періодично переривати на регламентовані перерви, які встановлюються для забезпечення працездатності та збереження здоров'я, або замінити іншою роботою з метою скорочення робочого навантаження у екрану.

3. Час регламентованих перерв протягом робочого дня встановлюється в залежності від його тривалості, виду і категорії трудової діяльності (табл. 3.2).

Група А - робота з зчитування інформації з екрану ПК з попереднім запитом;

Група Б - робота з введення інформації;

Група В - творча робота в режимі діалогу з ПК.

При виконанні протягом робочого дня робіт, що відносяться до різних груп, за основну роботу з ПК слід приймати таку, яка займає не менше 50 % часу протягом робочого дня (зміни)

Таблиця 4.2 – Основні вимоги щодо роботи з ПК

Категорія роботи з ПК	Рівень навантаження за робочу зміну при види робіт з ПК			Сумарний час регламентованих перерв, хвилин	
	група А кількість знаків	група В, кількість знаків	група В, годин	при 8- часовій зміні	при 12- часовій зміні
I	До 20000	До 15000	До 2,0	30	70
II	До 40000	До 30000	До 4,0	50	90
III	До 60000	До 40000	До 6,0	70	120

При 8-часовій робочій зміні і роботі з ПК регламентовані перерви встановлюються:

для I категорії робіт через 2 години від початку робочої зміні і через 2 години після обідньої перерви тривалістю 15 хвилин кожний;

для II категорії робіт через 2 години від початку робочої зміні і через 1,5-2 години після обідньої перерви тривалістю 15 хвилин кожен або тривалістю 10 хвилин через кожну годину роботи;

для III категорії робіт через 1,5-2 години від початку робочої зміні і через 1,5-2 години після обідньої перерви тривалістю 20 хвилин кожен або тривалістю 15 хвилин через кожну годину роботи.

При 12-часовій робочій зміні і роботі з ПК регламентовані перерви встановлюються в перші 8 годин роботи аналогічно перерв у 8 - годинному робочому зміні, а протягом останніх 4 годин роботи незалежно від категорії і виду робіт, щогодини тривалістю 15 хвилин.

4. Тривалість безперервної роботи з ПК без регламентованого перерви не повинна перевищувати 2 годин.

5. Під час перерв для зниження нервово-емоційної напруги, стомлення зорового аналізатора, поліпшення стану м'язів плечового поясу, рук, спини, шиї і ніг доцільно виконувати комплекси вправ.

6. З метою зменшення негативного впливу монотонності необхідно застосовувати чергування операцій.

7. Не слід залишати обладнання включеним без нагляду. При необхідності припинення на деякий час роботи коректно закриваються всі активні завдання й устаткування вимикається.

8. При роботі з ПК не дозволяється:

- при ввімкненому живленні доторкатися до панелей з роз'ємами устаткування, роз'ємів живлення і сполучних кабелів, екрану монітора;
- захарашувати верхні панелі обладнання, робоче місце паперами, сторонніми предметами;

- здійснювати перемикання, відключення живлення під час виконання поточної задачі;
- допускати попадання вологи на поверхню обладнання;
- включати сильно охолоджене (принесене з вулиці в зимовий час) обладнання;
- проводити самостійно розтин і ремонт обладнання;
- витирати пил на включеному обладнанні;
- допускати знаходження поблизу обладнання сторонніх осіб.

4.2.4 Вимоги безпеки в аварійних ситуаціях

1. В аварійних ситуаціях необхідно:

- при пошкодженні обладнання, кабелів, проводів, несправності заземлення, появи запах гару, виникненні незвичайного шуму та інших несправності негайно відключити електроживлення обладнання і повідомити про це безпосереднього керівника і особі, що здійснює технічне обслуговування обладнання;
- у разі збою в роботі устаткування ПК або програмного забезпечення викликати спеціаліста організації, що здійснює технічне обслуговування даного обладнання, для усунення неполадок;
- при загоряння електропроводки, устаткування негайно відключити електроживлення і прийняти заходи з гасіння пожежі з допомогою наявних первинних засобів пожежогасіння, повідомити про подію безпосереднього керівника. Застосування води і пінних вогнегасників для знаходиться під напругою електроустаткування неприпустимо. Для цієї мети використовуються вогнегасники вуглекислотні;
- у разі раптового погіршення здоров'я (посилення серцебиття, появи головного болю та ін.) припинити роботу, вимкнути обладнання, повідомити про це керівника і при необхідності звернутися до лікаря.

у разі нещасного випадку на виробництві необхідно:

- швидко вжити заходів щодо запобігання впливу на потерпілого травмуючих факторів, надання першої допомоги потерпілому, виклик на місце події медичних працівників медпункту по телефону або доставки потерпілого в лікувальну установу;
- повідомити про подію керівників.

4.2.5 Вимоги безпеки після закінчення роботи

1. По закінченні роботи працівник зобов'язаний:

- коректно закрити всі активні завдання;
- при наявності дискети в дисководі витягти її;
- вимкнути живлення системного блоку;
- вимкнути живлення всіх периферійних пристроїв;
- відключити блок безперебійного живлення;
- відключити кабель живлення від мережі;
- оглянути і привести в порядок робоче місце;
- про несправності обладнання та інших зауваження по роботі з ПК повідомити безпосереднього керівника або особі, що здійснює технічне обслуговування обладнання;
- при необхідності вимити руки з милом

4.3 Висновки до розділу 4

У межах третього розділу розкрито питання безпеки життєдіяльності та охорони праці. Здійснено розрахунок повітрообміну, отримані дані відповідають встановленим нормам, також здійснено розрахунок шуму, рівень шуму не перевищує прийнятої норми для даного класу приміщення, отже шумоподавлюючі засоби використовувати не потрібно.

4.4 Перелік джерел посилань до розділу 4

1. Кодекс законів про працю: Закон України // Відомості верховної ради від 10.12.1971. – № 50. – ст. 375 – Редакція від 02.04.2020, підстава – 540-IX. – Режим доступу. – <https://zakon.rada.gov.ua/laws/show/322-08>.
2. Про охорону праці: Закон України // Відомості верховної ради від 14.10.1992. – № 49. – ст.668 – Редакція від 27.12.2019, підстава – 341-IX. – Режим доступу. – <https://zakon.rada.gov.ua/laws/show/2694-12>.
3. Про загальнообов'язкове державне соціальне страхування: Закон України // Відомості верховної ради 01.01.2001. – № 46-47. – ст.403 – Редакція від 18.03.2020, підстава – 533-IX. – Режим доступу. – <https://zakon.rada.gov.ua/laws/show/1105-14>.
4. Конституція України: Закон України // Відомості верховної ради від 28.06.1996. – № 30. – ст. 141 – Редакція від 01.01.2020, підстава – 27-IX. – Режим доступу. – <https://zakon.rada.gov.ua/laws/show/254%D0%BA/96-%D0%B2%D1%80>.
5. ГОСТ 12.1.005-88 Общие санитарно-гигиенические требования к воздуху рабочей зоны от 01.01.1989 года. – Режим доступа. – http://online.budstandart.com/ua/catalog/doc-page.html?id_doc=6264.
6. ДСН 3.3.6.037-99 Санитарные нормы производственного шума, ультразвука и инфразвука от 01.12.1999 года. – Режим доступа. – http://online.budstandart.com/ru/catalog/doc-page?id_doc=48147.
7. Івах Р., Основи охорони праці / Івах Р., Бедрій Я., Білінський Б., Козир М. // Навчальний посібник. – К.: Кондор, 2011. – 464 с.
8. Атаманчук П.С., Безпека життєдіяльності та охорона праці (Практичний курс) / П.С. Атаманчук, В.В. Мендерецький, О.П. Панчук, О.Г. Чорна // Навчальний посібник. – Кам'янець-Подільський: "Думка", 2010. – 152 с.

ВИСНОВКИ

У межах дипломного проекту проведено розробку мобільного додатку під ОС Андроїд для перегляду файлів.

Для досягнення поставленої мети вирішено такі завдання:

- розглянуто інформаційні технології використання мобільних додатків;
- дано визначення: мобільні технології, протоколи, режими, мобільні додатки;
- проаналізовані технології розробки мобільних додатків під ОС Android;
- здійснено розробку інформаційних програмних засобів для мобільного пристрою;

Розроблена система для перегляду файлів має простий і зрозумілий інтерфейс, в той же час має широкі функціональні можливості.

Додаток має такі особливості:

- мобільність – можна скористатися додатком, де б користувач не знаходився і в будь-який час;
- можливість налаштування зручного перегляду файлів різного формату;
- збереження інформації про всі файли, їх структурування.

Таким чином, була вивчена предметна область, визначено ряд проблем, які необхідно вирішити, виявлені необхідні складові інформаційної системи, обрана апаратна і програмна основа для реалізації проекту.

Сильні сторони проекту: проект розроблений спеціально для ОС Android, що динамічно розвивається; функція перегляду файлів зручна у використанні. Бурхливий розвиток в області бездротової комунікації та систем обробки даних створює численні шляхи розвитку проекту в майбутньому паралельно вдосконаленню технологій.

Додаток А

Лістинг програмного коду

```
1 package com.nexes.manager;
2 public final class R {
3     public static final class attr {
4     }
5     public static final class color {
6         public static final int black=0x7f050002;
7         public static final int brightBlue=0x7f050007;
8         public static final int brightGreen=0x7f050006;
9         public static final int brightRed=0x7f050005;
10        public static final int darkBlue=0x7f050008;
11        public static final int darkGray=0x7f050003;
12        public static final int lightGray=0x7f050004;
13        public static final int offWhite=0x7f050001;
14        public static final int transGray=0x7f05000a;
15        public static final int transparent=0x7f050009;
16        public static final int white=0x7f050000;
17    }
18    public static final class drawable {
19        public static final int appicon=0x7f020000;
20        public static final int back=0x7f020001;
21        public static final int color=0x7f020002;
22        public static final int config32=0x7f020003;
23        public static final int excel=0x7f020004;
24        public static final int filter=0x7f020005;
25        public static final int folder=0x7f020006;
26        public static final int folder_full=0x7f020007;
27        public static final int help=0x7f020008;
28        public static final int home=0x7f020009;
29        public static final int html32=0x7f02000a;
30        public static final int icon=0x7f02000b;
31        public static final int icon64=0x7f02000c;
32        public static final int image=0x7f02000d;
33        public static final int info=0x7f02000e;
34        public static final int jar32=0x7f02000f;
35        public static final int logout=0x7f020010;
36        public static final int more=0x7f020011;
37        public static final int movies=0x7f020012;
38        public static final int multiselect_38=0x7f020013;
39        public static final int multiselectbox_38=0x7f020014;
40        public static final int music=0x7f020015;
41        public static final int newfolder=0x7f020016;
42        public static final int pdf=0x7f020017;
43        public static final int ppt=0x7f020018;
44        public static final int processinfo=0x7f020019;
45        public static final int rename=0x7f02001a;
46        public static final int search=0x7f02001b;
47        public static final int setting=0x7f02001c;
48        public static final int singlecheck=0x7f02001d;
49        public static final int space=0x7f02001e;
50        public static final int text=0x7f02001f;
51        public static final int toolbox=0x7f020020;
52        public static final int warning=0x7f020021;
53        public static final int widget_folder=0x7f020022;
54        public static final int widget_frame1=0x7f020023;
55        public static final int wireless=0x7f020024;
56        public static final int word=0x7f020025;
57        public static final int xml32=0x7f020026;
```

```
58 public static final int zip=0x7f020027;
59 }
60 public static final class id {
61 public static final int available_mem_label=0x7f07002e;
62 public static final int back_button=0x7f07001b;
63 public static final int backup_button_all=0x7f070008;
64 public static final int backup_label=0x7f070007;
65 public static final int bottom_view=0x7f070038;
66 public static final int detail_label=0x7f070028;
67 public static final int dirs_label=0x7f070012;
68 public static final int edit_view_label=0x7f070009;
69 public static final int edit_view_text=0x7f07000a;
70 public static final int fifth_title=0x7f070019;
71 public static final int files_label=0x7f070014;
72 public static final int first_title=0x7f070011;
73 public static final int fourth_title=0x7f070017;
74 public static final int free_space=0x7f07001a;
75 public static final int help_button=0x7f070027;
76 public static final int help_email_bt=0x7f07000c;
77 public static final int help_top_label=0x7f07000b;
78 public static final int help_website_bt=0x7f07000d;
79 public static final int hidden_attach=0x7f07002a;
80 public static final int hidden_buttons=0x7f070029;
81 public static final int hidden_copy=0x7f07002c;
82 public static final int hidden_delete=0x7f07002b;
83 public static final int hidden_move=0x7f07002d;
84 public static final int home_button=0x7f070023;
85 public static final int info_button=0x7f070026;
86 public static final int info_icon=0x7f07000e;
87 public static final int input_cancel_b=0x7f070021;
88 public static final int input_create_b=0x7f070020;
89 public static final int input_icon=0x7f07001e;
90 public static final int input_inputText=0x7f07001f;
91 public static final int input_label=0x7f07001d;
92 public static final int manage_button=0x7f070024;
93 public static final int media_close_button=0x7f070006;
94 public static final int media_play_button=0x7f070005;
95 public static final int multiselect_button=0x7f070025;
96 public static final int multiselect_icon=0x7f070036;
97 public static final int music_label=0x7f070004;
98 public static final int name_label=0x7f07000f;
99 public static final int num_processes_label=0x7f07002f;
100 public static final int path_label=0x7f070010;
101 public static final int row_image=0x7f070035;
102 public static final int second_title=0x7f070013;
103 public static final int setting_hidden_box=0x7f070031;
104 public static final int setting_storage_box=0x7f070033;
105 public static final int setting_text_color_button=0x7f070030;
106 public static final int setting_thumbnail_box=0x7f070032;
107 public static final int settings_sort_button=0x7f070034;
108 public static final int storage_label=0x7f070022;
109 public static final int third_title=0x7f070015;
110 public static final int time_stamp=0x7f070016;
111 public static final int top_view=0x7f070037;
112 public static final int total_size=0x7f070018;
113 public static final int widget_cancel_button=0x7f07003b;
114 public static final int widget_filelabel=0x7f070003;
115 public static final int widget_folderlabel=0x7f070002;
116 public static final int widget_icon=0x7f070000;
117 public static final int widget_label=0x7f070039;
118 public static final int widget_namelabel=0x7f070001;
119 public static final int widget_ok_button=0x7f07003a;
120 public static final int zip_button=0x7f07001c;
```

```

121     }
122     public static final class layout {
123     public static final int appwidget=0x7f030000;
124     public static final int audio_layout=0x7f030001;
125     public static final int backup_layout=0x7f030002;
126     public static final int edit_layout=0x7f030003;
127     public static final int help_layout=0x7f030004;
128     public static final int info_layout=0x7f030005;
129     public static final int input_layout=0x7f030006;
130     public static final int main=0x7f030007;
131     public static final int manage_layout=0x7f030008;
132     public static final int settings=0x7f030009;
133     public static final int tablerow=0x7f03000a;
134     public static final int widget_config_layout=0x7f03000b;
135     }
136     public static final class string {
137     public static final int app_name=0x7f060000;
138     public static final int dir_info=0x7f060002;
139     public static final int manager_act=0x7f060003;
140     public static final int media_dialog=0x7f060001;
141     }
142     public static final class xml {
143     public static final int folder_widget_info=0x7f040000;
144     }
145     }

146     import android.app.ListActivity;
147     import android.content.DialogInterface;
148     import android.content.DialogInterface.OnClickListener;
149     import android.content.Intent;
150     import android.graphics.drawable.Drawable;
151     import android.os.Bundle;
152     import android.view.View;
153     import android.widget.ListView;
154     import android.widget.Toast;
155     import com.mobeegal.android.R;
156     import com.mobeegal.android.model.IconifiedText;
157     import com.mobeegal.android.view.IconifiedTextListAdapter;

158     import java.io.File;
159     import java.util.ArrayList;
160     import java.util.Collections;
161     import java.util.List;
162     import java.util.logging.Logger;

163     public class AndroidBrowser
164     extends ListActivity
165     {

166     private static Logger logger = Logger.getLogger("Testcatalogs");

167     private enum DISPLAYMODE
168     {

169     ABSOLUTE, RELATIVE;
170     }

171     private int position;
172     protected static final int SUB_ACTIVITY_REQUEST_CODE = 1337;
173     public String str1, str2;
174     private final DISPLAYMODE displayMode = DISPLAYMODE.RELATIVE;
175     private List<IconifiedText> directoryEntries =

```

```
176     new ArrayList<IconifiedText>();
177     private File currentDirectory = new File("/");
178     private String ch;
179     private int ch1;
180     private Intent myIntent;

181     /**
182     Called when the activity is first created.
183     */
184     public void onCreate(Bundle icle)
185     {
186     super.onCreate(icle);
187     setTheme(android.R.style.Theme_Black);

188     browseToRoot();
189     this.setSelection(0);

190     Bundle bundleobj = this getIntent().getExtras();
191     if (bundleobj != null)
192     {

193     str1 = bundleobj.getString("value1");

194     }

195     }

196     private void browseToRoot()
197     {

198     browseTo(new File("/"));
199     }

200     private void upOneLevel()
201     {
202     if (this.currentDirectory.getParent() != null)
203     {
204     this.browseTo(this.currentDirectory.getParentFile());
205     }
206     }

207     private void browseTo(final File aDirectory)
208     {

209     if (this.displayMode == DISPLAYMODE.RELATIVE)
210     {
211     this.setTitle(aDirectory.getAbsolutePath() + " :: ");
212     }
213     if (aDirectory.isDirectory())
214     {
215     this.currentDirectory = aDirectory;
216     fill(aDirectory.listFiles());
217     }
218     else
219     {
220     OnClickListener okButtonListener = new OnClickListener()
221     {

222     public void onClick(DialogInterface arg0, int arg1)
223     {
224     AndroidBrowser.this.openFile(aDirectory);
225     }
```

```

226     };
227     OnClickListener viewButtonListener = new OnClickListener()
228     {

229         public void onClick(DialogInterface arg0, int arg1)
230         {
231             AndroidBrowser.this.playFile(aDirectory);
232             // Intent playIntent = new
Intent(AndroidFileBrowser.this, PlayMedia.class);
233             //
startActivityForResult(playIntent, 0);
234         }
235     };

236     OnClickListener cancelButtonListener = new OnClickListener()
237     {
238         // @Override
239         public void onClick(DialogInterface arg0, int arg1)
240         {
241             // Do nothing ^^
242         }
243     };
244     // AlertDialog.show(this, "Upload", position, " Do you
want to Upload ?\n", "Upload", okButtonListener, "cancel",
cancelButtonListener, "view", viewButtonListener, false, null);

245     }
246     }

247     private void openFile(File aFile)
248     {
249         String filename = aFile.getName();

250         if (checkEndsWithInStringArray(filename,
251             getResources().getStringArray(R.array.fileEndingImage)))
252         {
253             String uploadingFile = aFile.getAbsolutePath();
254             Bundle uploadfile = new Bundle();
255             Intent myIntent =
256             new Intent(AndroidBrowser.this, Uploadmultimedia.class);
257             uploadfile.putString("key", uploadingFile);
258             uploadfile.putString("key1", str1);
259             logger.info("count = " + str1);
260             myIntent.putExtras(uploadfile);
261             startActivityForResult(myIntent, 0);
262         }
263         else if (checkEndsWithInStringArray(filename,
264             getResources().getStringArray(R.array.fileEndingVideo)))
265         {
266             String uploadingFile = aFile.getAbsolutePath();
267             Bundle uploadfile = new Bundle();
268             Intent myIntent =
269             new Intent(AndroidBrowser.this, Uploadmultimedia.class);
270             uploadfile.putString("key", uploadingFile);
271             uploadfile.putString("key1", str1);
272             logger.info("count = " + str1);
273             myIntent.putExtras(uploadfile);
274             startActivityForResult(myIntent, 0);
275         }
276         else if (checkEndsWithInStringArray(filename,
277             getResources().getStringArray(R.array.fileEndingAudio)))
278         {
279             String uploadingFile = aFile.getAbsolutePath();

```



```

280 Bundle uploadfile = new Bundle();
281 Intent myIntent =
282 new Intent(AndroidBrowser.this, Uploadmultimedia.class);
283 uploadfile.putString("key", uploadingFile);
284 uploadfile.putString("key1", str1);
285 logger.info("count = " + str1);
286 myIntent.putExtras(uploadfile);
287 startActivityForResult(myIntent, 0);
288 }
289 else
290 {
291 Toast.makeText(AndroidBrowser.this, "FileFormat not Supported",
292 Toast.LENGTH_SHORT).show();
293 }

294 }

295 private void playFile(File aFile)
296 {
297 String filename = aFile.getName();
298 if (checkEndsWithInStringArray(filename,
299 getResources().getStringArray(R.array.fileEndingImage)))
300 {
301 String uploadingimage = "file://" + aFile.getAbsolutePath();
302 Bundle uploadimage = new Bundle();
303 Intent myIntent1 =
304 new Intent(AndroidBrowser.this, UploadGallery.class);
305 uploadimage.putString("key", uploadingimage);
306 uploadimage.putString("key1", str1);
307 logger.info("count = " + str1);
308 myIntent1.putExtras(uploadimage);
309 startActivityForResult(myIntent1, 0);
310 }
311 else if (checkEndsWithInStringArray(filename,
312 getResources().getStringArray(R.array.fileEndingVideo)))
313 {
314 String uploadingFile = "file://" + aFile.getAbsolutePath();
315 Bundle uploadfile = new Bundle();
316 Intent myIntent1 = new Intent(AndroidBrowser.this, PlayMedia.class);
317 uploadfile.putString("key", uploadingFile);
318 uploadfile.putString("key1", "Video File");
319 myIntent1.putExtras(uploadfile);
320 startActivityForResult(myIntent1, 0);
321 }
322 else if (checkEndsWithInStringArray(filename,
323 getResources().getStringArray(R.array.fileEndingAudio)))
324 {
325 String uploadingFile = "file://" + aFile.getAbsolutePath();
326 Bundle uploadfile = new Bundle();
327 Intent myIntent1 = new Intent(AndroidBrowser.this, PlayMedia.class);
328 uploadfile.putString("key", uploadingFile);
329 uploadfile.putString("key1", "Audio File");
330 myIntent1.putExtras(uploadfile);
331 startActivityForResult(myIntent1, 0);
332 }
333 else
334 {
335 Toast.makeText(AndroidBrowser.this, "FileFormat not Supported",
336 Toast.LENGTH_SHORT).show();
337 }
338 }

```

```

339 private void fill(File[] files)
340 {
341     this.directoryEntries.clear();

342     // Add the "." == "current directory"
343     this.directoryEntries.add(new IconifiedText(
344         ".",
345         getResources().getDrawable(R.drawable.folder)));
346     // and the ".." == 'Up one level'
347     if (this.currentDirectory.getParent() != null)
348     {
349         this.directoryEntries.add(new IconifiedText(
350             "..",
351             getResources().getDrawable(R.drawable.uponelevel)));
352     }

353     Drawable currentIcon = null;
354     for (File currentFile : files)
355     {
356         if (currentFile.isDirectory())
357         {
358             currentIcon = getResources().getDrawable(R.drawable.folder);
359         }
360         else
361         {
362             String fileName = currentFile.getName();

363             if (checkEndsWithInStringArray(fileName, getResources().
364                 getStringArray(R.array.fileEndingImage)))
365             {
366                 currentIcon = getResources().getDrawable(R.drawable.image);
367             }
368             else if (checkEndsWithInStringArray(fileName, getResources().
369                 getStringArray(R.array.fileEndingWebText)))
370             {
371                 currentIcon =
372                 getResources().getDrawable(R.drawable.webtext);
373             }
374             else if (checkEndsWithInStringArray(fileName, getResources().
375                 getStringArray(R.array.fileEndingPackage)))
376             {
377                 currentIcon = getResources().getDrawable(R.drawable.packed);
378             }
379             else if (checkEndsWithInStringArray(fileName, getResources().
380                 getStringArray(R.array.fileEndingVideo)))
381             {
382                 currentIcon = getResources().getDrawable(R.drawable.video);
383             }
384             else if (checkEndsWithInStringArray(fileName, getResources().
385                 getStringArray(R.array.fileEndingAudio)))
386             {
387                 currentIcon = getResources().getDrawable(R.drawable.audio);
388             }
389             else
390             {
391                 currentIcon = getResources().getDrawable(R.drawable.text);
392             }
393         }
394         switch (this.displayMode)
395         {
396             case ABSOLUTE:
397                 /* On absolute Mode, we show the full path */
398                 this.directoryEntries.add(new IconifiedText(

```

```

399     currentFile.getPath(), currentIcon));
400     break;
401     case RELATIVE:
402         /* On relative Mode, we have to cut the
403         current-path at the beginning */
404         int currentPathStringLength =
405         this.currentDirectory.getAbsolutePath().length();
406         this.directoryEntries.add(new IconifiedText(
407         currentFile.getAbsolutePath().
408         substring(currentPathStringLength),
409         currentIcon));

410     break;
411     }
412     }
413     Collections.sort(this.directoryEntries);

414     IconifiedTextListAdapter itla = new IconifiedTextListAdapter(this);
415     itla.setListItems(this.directoryEntries);
416     this.setAdapter(itla);
417     }

418     @Override
419     protected void onItemClick(ListView l, View v, int position,
long id)
420     {
421         super.onItemClick(l, v, position, id);
422         //int selectionRowID = (int) this.getSelectionRowID();
423         String selectedFileString =
424         this.directoryEntries.get(position).getText();
425         if (selectedFileString.equals("."))
426         {
427             // Refresh
428             this.browseTo(this.currentDirectory);
429         }
430         else if (selectedFileString.equals(".."))
431         {
432             this.upOneLevel();
433         }
434         else if (selectedFileString.equals("data"))
435         {
436             this.browseTo(new File("/data/misc/"));
437         }
438         else
439         {
440             File clickedFile = null;
441             switch (this.displayMode)
442             {
443                 case RELATIVE:
444                     clickedFile = new File(this.currentDirectory
445                     .getAbsolutePath() +
446                     this.directoryEntries.get(position).getText());
447                     break;
448                 case ABSOLUTE:
449                     clickedFile = new File(
450                     this.directoryEntries.get(position).getText());
451                     break;
452             }
453             if (clickedFile != null)
454             {
455                 this.browseTo(clickedFile);
456             }
457         }

```

```
458     }

459     private boolean checkEndsWithInStringArray(String checkItsEnd,
460     String[] fileEndings)
461     {
462     for (String aEnd : fileEndings)
463     {
464     if (checkItsEnd.endsWith(aEnd))
465     {
466     return true;
467     }
468     }
469     return false;
470     }
471     //added image view
472     }
```

Додаток Б
Комп'ютерна презентація

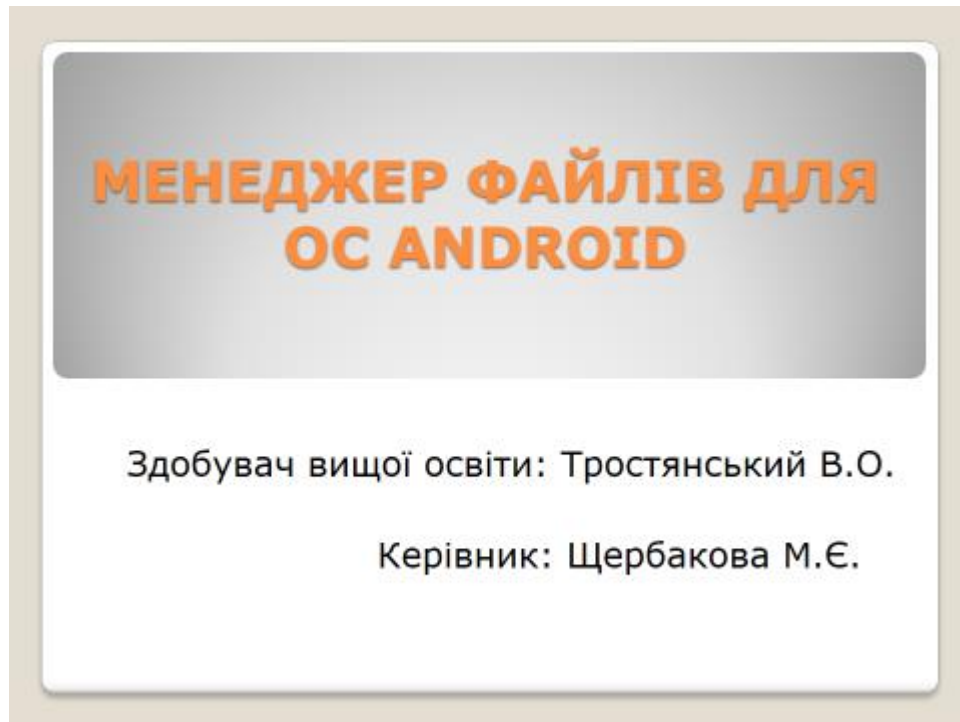


Рисунок Б.1 – Титульний слайд

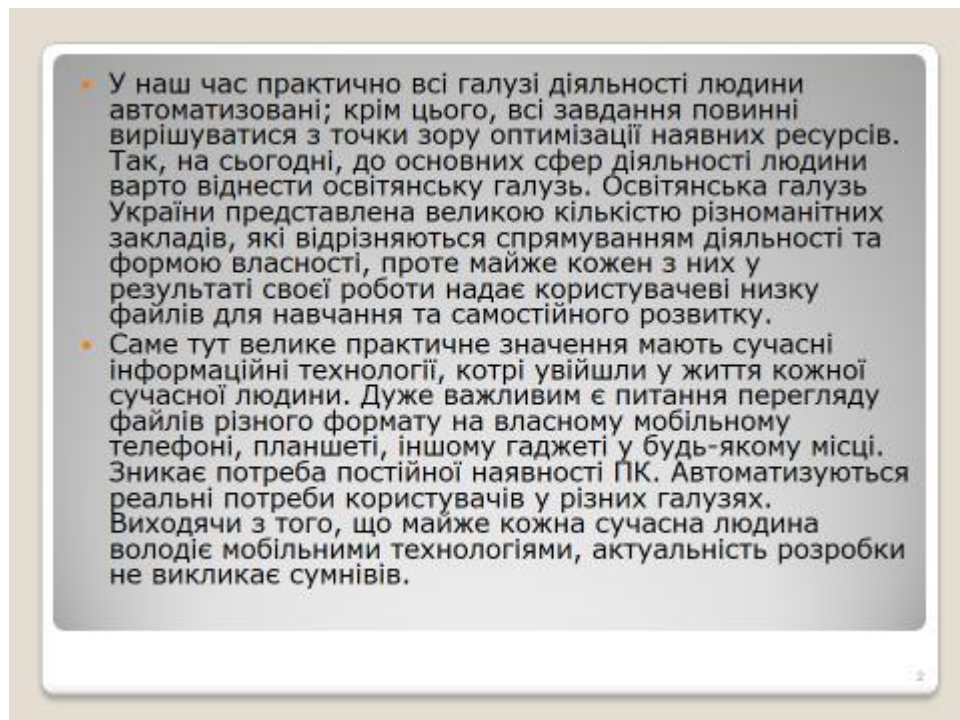


Рисунок Б.2 – Актуальність роботи

Мета дипломного проекту – розробити мобільний додаток під ОС Андроїд для перегляду файлів.

Додаток повинен мати такі особливості:

- перегляд різних форматів файлів, вивід переліку файлів, їх структуризація, переміщення та видалення;
- можливість налаштування зручного перегляду будь-якого файлу;
- візуалізація усіх файлів в окремій папці з повним збереженням;
- можливість сортування файлів за значимістю.

3

Рисунок Б.3 – Мета дипломного проекту

Програми-аналоги

ASTRO File Manager

ES File Explorer

AntTek Explorer

4

Рисунок Б.4 – Програмні продукти подібного призначення

Порівняльна характеристика аналогів

Назва	ASTRO File Manager	ES File Explorer	AntTek Explorer
Характеристика			
Каталог всіх файлів	+	+	+
Формат файлів txt ; doc; docx	+	+	+
Відтворення музики	-	-	+
gif-анімація	-	-	+
Структуризація файлів	+	-	+
Інформація про всі файли	+	+	+
Необхідність підключення до Інтернету	+	-	-
ОС	Android 5 і вище	Android 4.2 і вище	Android 4 і вище
Розмір	11.84 М	95 М	4,6 М
Вартість	умовно безкоштовна	умовно безкоштовна	умовно безкоштовна

Рисунок Б.5 – Характеристики програм подібного призначення



Рисунок Б.6 – Взаємодія частин програми



Рисунок Б.7 – Діаграма варіантів використання

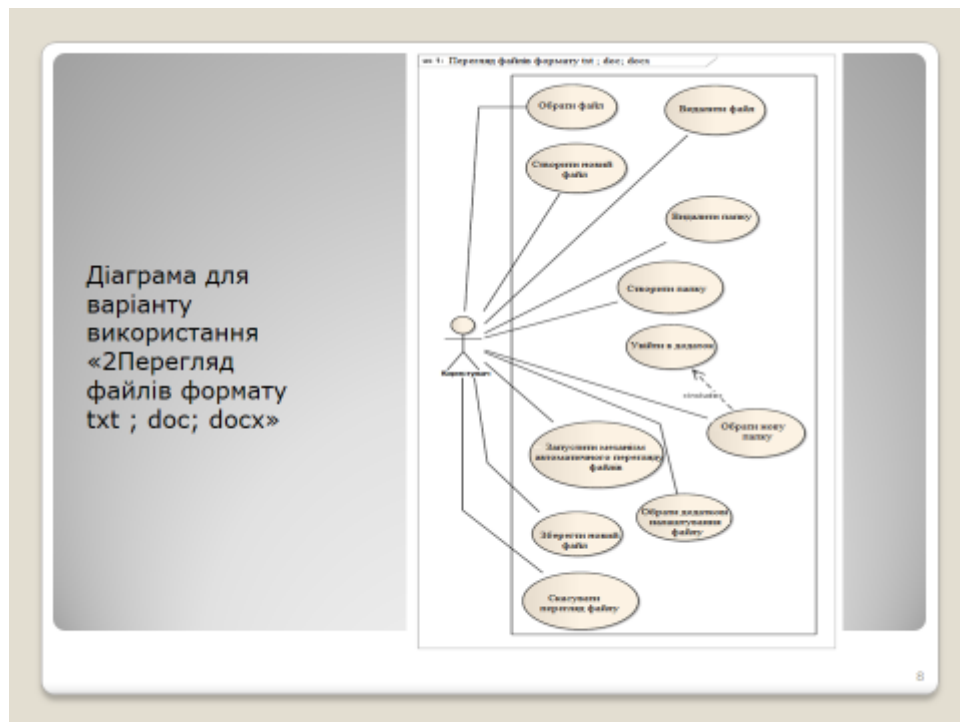


Рисунок Б.8 – Можливі дії з файлами

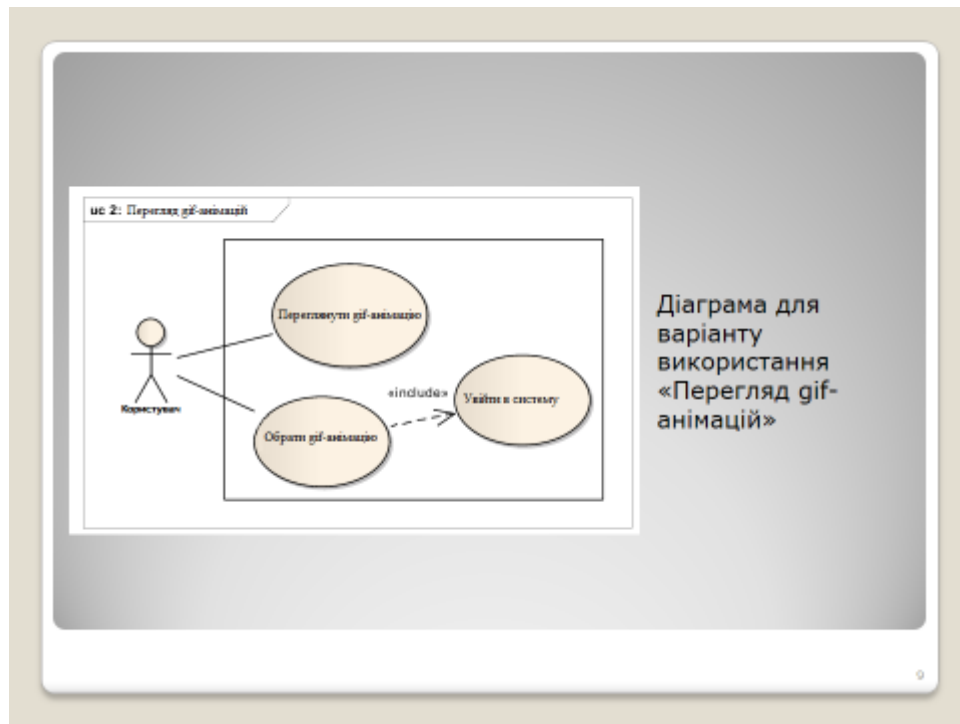


Рисунок Б.9 – Перегляд анімації за допомогою менеджера файлів

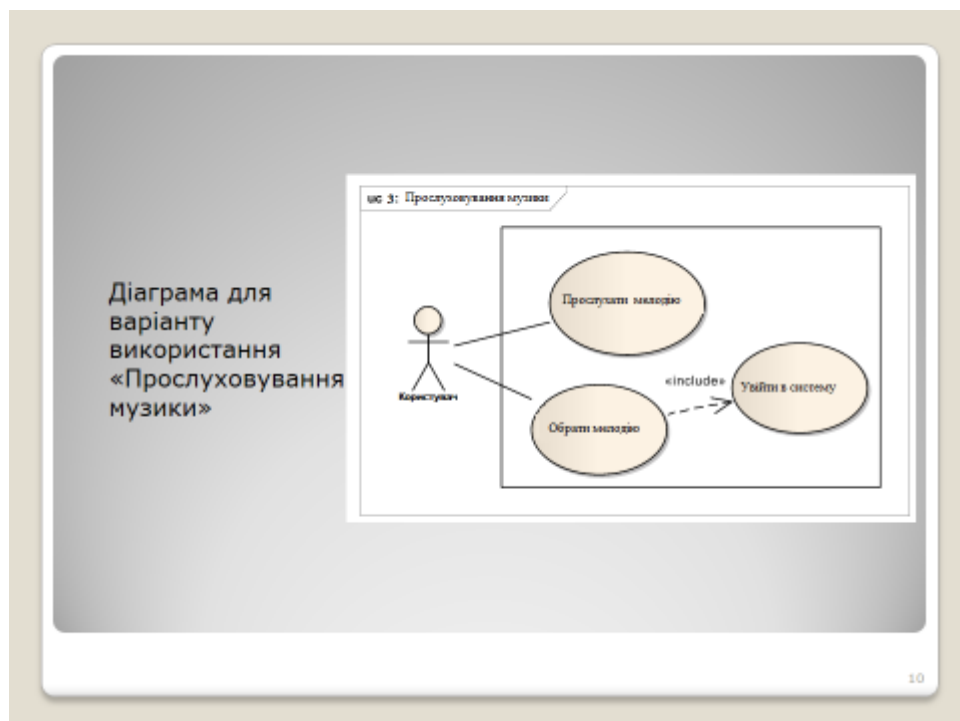


Рисунок Б.10 – Прослуховування музики за допомогою менеджера файлів

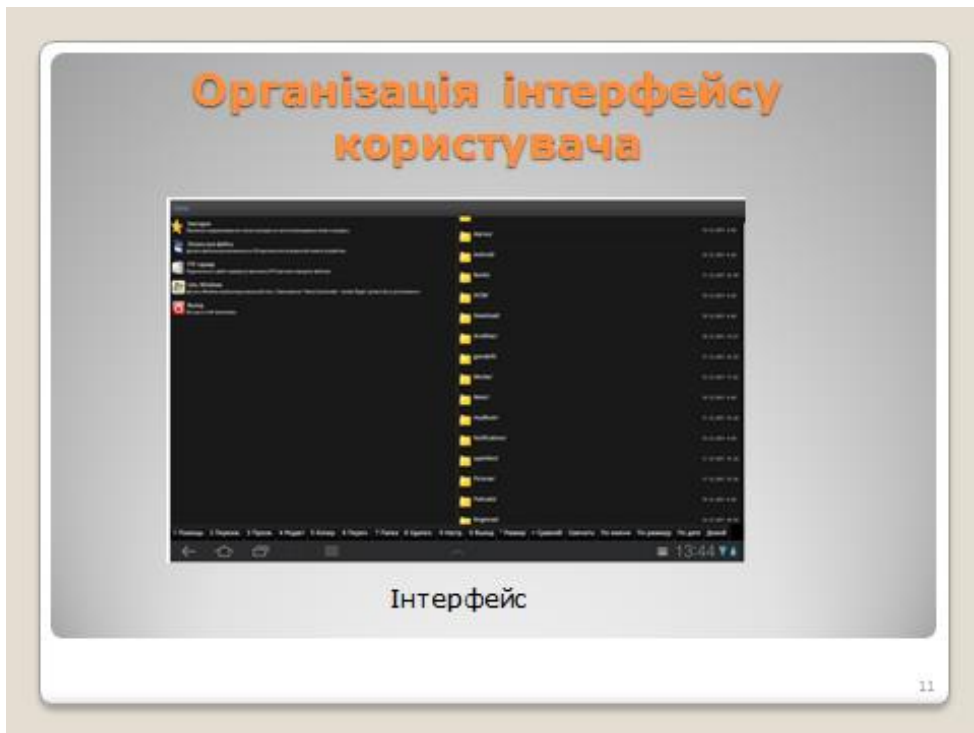


Рисунок Б.11 – Інтерфейс користувача

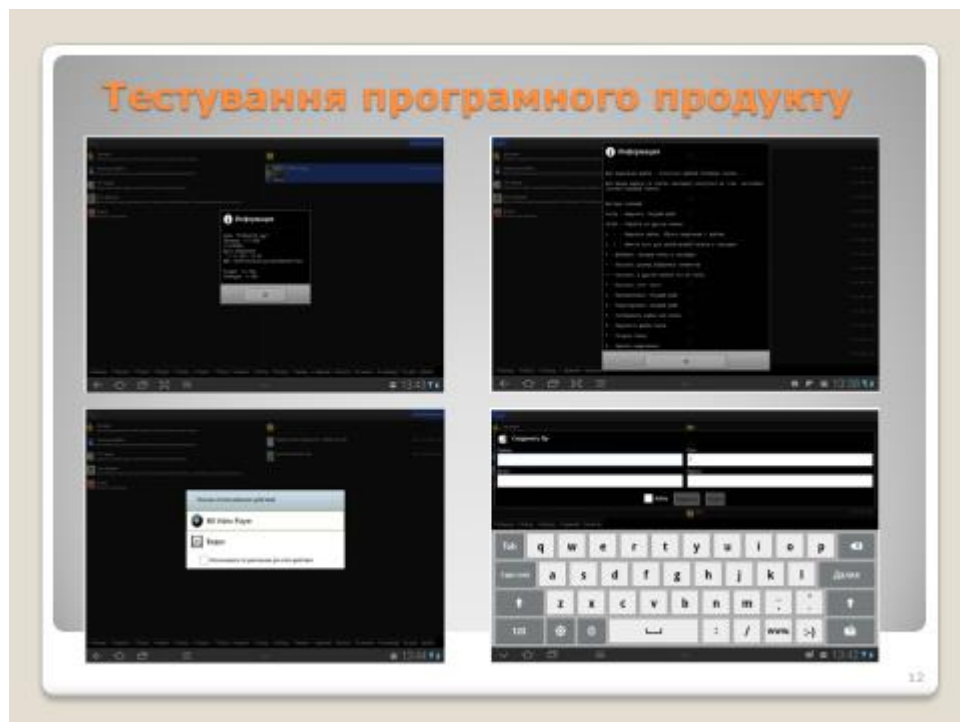


Рисунок Б.12 – Тестування мнеджера файлів

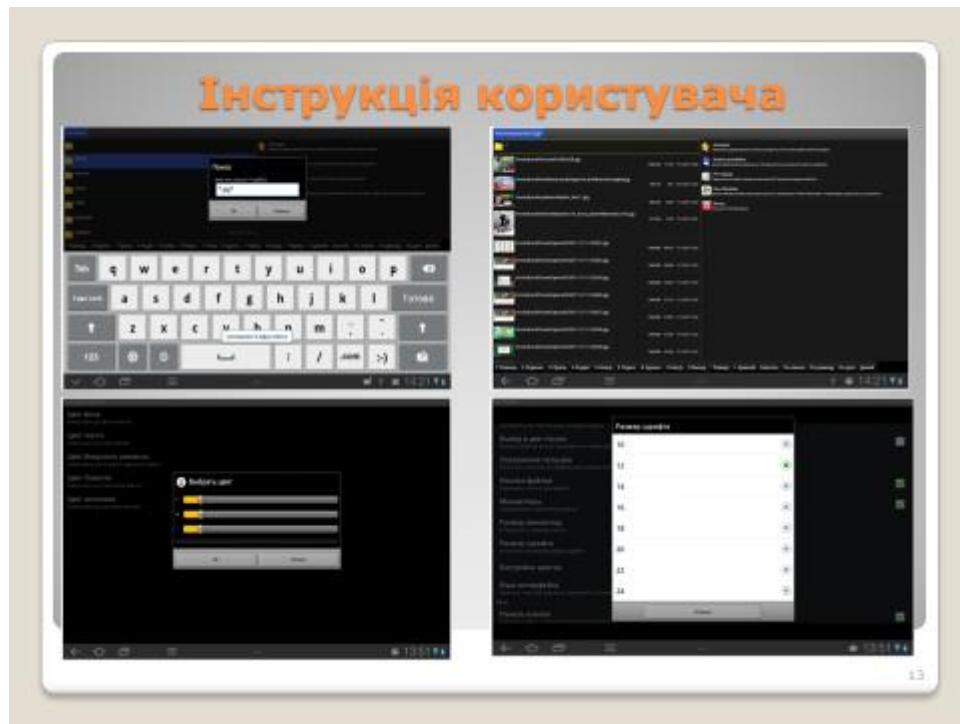


Рисунок Б.13 – Налаштування менеджера файлів

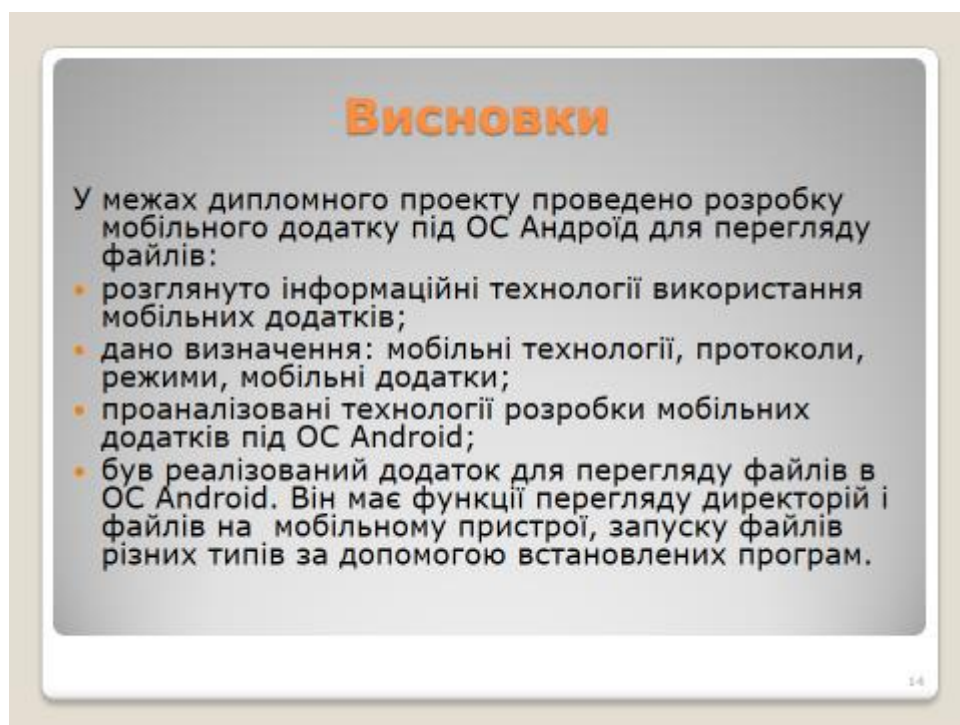


Рисунок Б.14 – Висновки