

Міністерство освіти і науки України
Криворізький національний університет
Кафедра моделювання та програмного забезпечення

КВАЛІФІКАЦІЙНА РОБОТА
на здобуття ступеня вищої освіти бакалавра
зі спеціальності 121 – Інженерія програмного забезпечення

На тему: Розробка програмного забезпечення комплексу віртуальних лабораторій з фізики

Засвідчую, що в цій
кваліфікаційній роботі немає
запозичень із праць інших авторів
без відповідних посилань.

Студент гр. ПЗ-21 ск
_____ /К. О. Павленко /

Керівник
кваліфікаційної роботи _____ / А. М. Стрюк /

В.о. завідувача кафедри _____ / А. М. Стрюк /

Кривий Ріг
2024

Криворізький національний університет

Факультет: Інформаційних технологій

Кафедра: Моделювання та програмного забезпечення

Ступінь вищої освіти: бакалавр

Спеціальність: 121 – Інженерія програмного забезпечення

ЗАТВЕРДЖУЮ:
В.о. зав. кафедри
А. М. Стрюк
« ____ » ____ 20__ р.

ЗАВДАННЯ

на кваліфікаційну роботу

студенту групи ІІЗ-21 ск Павленку Кирилу Олеговичу

1. Тема: Розробка програмного забезпечення комплексу віртуальних лабораторій з фізики
затверджено наказом по КНУ № 22 від «13» лютого 2024 р.
2. Термін подання студентом закінченої роботи: «15» червня 2024 р.
3. Вихідні дані по роботі: розроблювана система повинна працювати з інтерактивними анімаціями, створювати звіт виконаної роботи, виконувати резервне копіювання даних.
4. Зміст пояснювальної записки (перелік питань, що їх треба розробити): виконати аналіз існуючих методів розв'язання задачі, вибрати метод відтворення анімації на мобільних пристроях, спроектувати автоматизовану систему генерації звітів, здійснити програмну реалізацію розробленої системи, провести тестування розробленої системи.
5. Перелік ілюстративного матеріалу: блок-схеми розроблених алгоритмів, схема взаємодії модулів системи, знімки екранних форм.

Календарний план:

№	Найменування етапів кваліфікаційної роботи	Термін виконання етапів роботи
1	Огляд літератури за тематикою та збір даних	27.02.2024 – 29.02.2024
2	Проведення порівняльного аналізу програм аналогічного призначення	30.02.2024 – 09.03.2024
3	Проведення порівняльного аналізу бібліотек для реалізації функцій програми	10.03.2024– 18.03.2024
4	Підготовка матеріалів першого розділу роботи	19.03.2024 – 23.03.2024
5	Розробка візуального стилю програмного забезпечення	24.03.2024– 30.03.2024
6	Підготовка матеріалів другого розділу роботи	1.04.2024 – 04.04.2024
7	Підготовка матеріалів третього розділу роботи	05.04.2024– 13.05.2024
8	Розробка програмного забезпечення для розпізнавання комплексу віртуальних лабораторій з фізики	14.05.2024 – 20.05.2024
9	Підготовка матеріалів четвертого розділу роботи	21.05.2024 – 26.05.2024
10	Оформлення пояснювальної записки	27.05.2024 – 05.06.2024

Дата видачі завдання: «01» квітня 2024 р.

Студент _____ / К. О. Павленко /

Керівник роботи _____ / А. М. Стрюк /

РЕФЕРАТ

ІНТЕРАКТИВНА АНІМАЦІЯ, ЛАБОРАТОРНІ РОБОТИ З ФІЗИКИ, ШКІЛЬНА ПРОГРАМА, МОБІЛЬНІ ПРИСТРОЇ.

Пояснювальна записка: 63 с., 18 рис., 2 дод., 9 джерел.

Метою кваліфікаційної роботи є створення програмного забезпечення комплексу віртуальних лабораторій з фізики.

У роботі проведено аналіз предметної області указаної задачі – лабораторних робіт для учнів старшої школи 10-11 класів з предмету фізика. Виконано їх адаптування під мобільні пристрої.

Для кожної лабораторної роботи сформована інтерактивна анімація та інструкція до її виконання. Розроблено систему, що дозволяє генерувати звіт виконаної роботи та надавати рекомендації в обчисленнях.

ABSTRACT

**INTERACTIVE ANIMATION, LABORATORY WORK IN PHYSICS,
SCHOOL CURRICULUM, MOBILE DEVICES.**

Thesis in: 63 p., 18 fig., 2 app., 9 sources.

The purpose of the qualification work is to create software for a complex of virtual laboratories in physics.

The work analyses the subject area of this task - laboratory work for high school students of grades 10-11 in the subject of physics. Their adaptation to mobile devices has been carried out.

An interactive animation and instructions for each laboratory work have been created. A system was developed that allows generating a report of the work performed and providing recommendations in calculations.

ЗМІСТ

1 ТЕОРЕТИЧНИЙ РОЗДІЛ.....	7
1.1 Аналіз професійної області проблеми	7
1.2 Аналіз існуючих рішень	9
1.3 Формулювання актуальності і завдань роботи	11
2 ПРОЕКТНИЙ РОЗДІЛ.....	13
2.1 Проектування інформаційної системи.....	13
2.2 Розробка структурної та функціональної схеми програмного комплексу	14
2.3 Розробка блок-схем основних алгоритмів програмного комплексу	16
2.4 Розробка моделі візуального інтерфейсу програмного комплексу	19
3 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ	25
3.1 Аналіз обраного середовища програмування	25
3.2 Програмна реалізація основних функцій	27
3.3 Методика роботи користувача.....	29
ВИСНОВОК.....	31
ПЕРЕЛІК ПОСИЛАНЬ.....	32
Додаток А.....	33
Додаток Б	63

1 ТЕОРЕТИЧНИЙ РОЗДІЛ

1.1 Аналіз професійної області проблеми

Нові можливості, які відкривають інформаційні технології впливають на всі сфери життя людей: сім'ю, роботу, освіту, тощо. Їх стрімко зростаючий потенціал охоплює як суспільство, так і окремі корпорації, значно покращуючи та створюючи нові форми організації праці.

Сьогодні вноситься багато змін, до яких потрібно підлаштовуватись. Протягом останніх кількох років зріс попит на дистанційні форми організації праці, включаючи навчання, оскільки небезпека зустрічей у реальному житті значно зросла, тому актуальною постає проблема зберегти позитивний досвід від занять в аудиторії, та перенести навчання у домашні умови.

Вже існує багато рішень для подолання наявних проблем, але комп'ютеризація дійшла не до всіх сфер розвитку людини. До того ж, портативність та мобільність грають далеко не останню роль у сьогоднішніх реаліях.

На наш погляд, розділ фізики потребує більш досконалого рішення для організації учбового процесу, тому було прийняте рішення створити мобільний застосунок для виконання лабораторних робіт з фізики, орієнтованих на програму старшої школи та учнів 10-11 класів.

Навчання визнається ключовою складовою людського життя, оскільки воно сприяє нашому особистісному розвитку, відбудовуючи навички і якості потрібні кожній людині. Проте в умовах, коли якісне навчання стає недоступним, інтелектуальний рівень занепадає, особливо це стосується учнів, які ще не вміють ефективно розпоряджатися своїм часом.

Використання інформаційних технологій у навчанні супроводжується рядом переваг, це підтверджує наукова стаття під назвою «Use of Visual Learning Media to Increase Student Learning Motivation» яка опублікована у журналі «World Psychology»

Дослідження виконали Дженсен В., Ерландссон Н., Бурманн У., Опаріна Б. та ін. Воно має на меті дослідити вплив візуальних навчальних медіа на мотивацію учнів.

У дослідженні використовувався метод глибинних інтерв'ю та збору даних. Опитування проводилося за допомогою анкет з відповідями від викладачів та учнів про їхній досвід, та думки щодо візуальних навчальних медіа.

Дослідження показує, що електронні інструменти подачі матеріалу, включаючи зображення та відео, значно підвищують мотивацію і зацікавленість учнів. Навчання з використанням візуальних засобів допомагає краще зрозуміти матеріал. Вони також допомагають викладачам ефективніше подавати матеріал, тим самим підвищуючи загальну якість освіти.

До проблем такого способу можна віднести високу вартість обладнання, що не завжди можливо забезпечити в умовах обмеженого шкільного бюджету та недостатня підготовка викладачів.

Вчителі фізики часто стикаються з браком програмного забезпечення, адаптованого до навчальних планів і потреб учнів. Це ускладнює процес підготовки та проведення уроків з використанням інформаційних технологій.

Учні можуть мати різний рівень підготовки та володіння технікою, що створює додаткові труднощі у забезпеченні рівних можливостей для всіх. Частина школярів може відчувати страх перед новими технологіями або не мати доступу до них поза школою.

Для успішного впровадження інновацій у навчання фізики, необхідно подолати ці проблеми, забезпечивши належне програмне забезпечення, його підтримку та підвищення кваліфікації вчителів, а також враховувати психологічні та соціальні аспекти навчання.

За допомогою нашого застосунку з інтерактивними анімаціями, можна зробити навчальний процес захоплюючим, та зберегти ефективність навчання у складних умовах.

1.2 Аналіз існуючих рішень

Давайте проведемо аналіз ринку програмного забезпечення для виявлення наявності програм з аналогічними призначенням. У результаті пошуку було виявлено мобільний застосунок під назвою «PhysicsApp», який містить анімації з фізичними експериментами, створених для навчання.

В цій версії програми доступні такі розділи, як «Механіка», «Квантова механіка» та «Електродинаміка», в кожному з яких представлено кілька фізичних експериментів. Крім того, програма може проводити обчислення величин за допомогою вбудованого калькулятора.

Проведемо рецензування програмного продукту. Приклад інтерфейсу користувача зображений на рисунку 1.

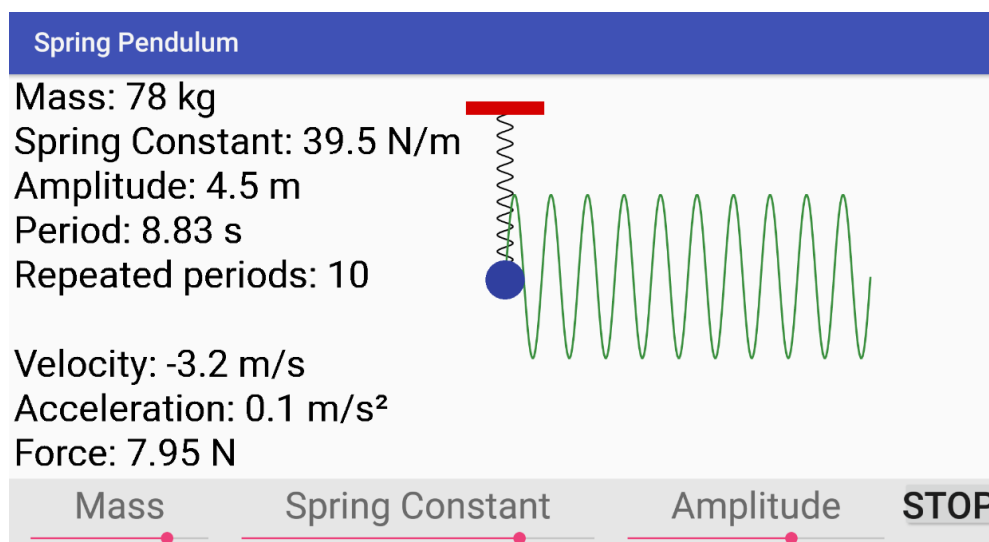


Рисунок 1.1 – Інтерфейс застосунку «PhysicsApp» [1].

Дане програмне забезпечення має наступні переваги:

- зручний зрозумілий інтерфейс;
- автоматизація обчислень;
- інтерактивні анімації.

Але цей застосунок також має ряд недоліків:

- не створює документів для друку;
- високий поріг входження;
- не має теоретичної бази;

Наступним проаналізуємо застосунок «Physics Lab». Приклад інтерфейсу користувача зображений на рисунку 1.2

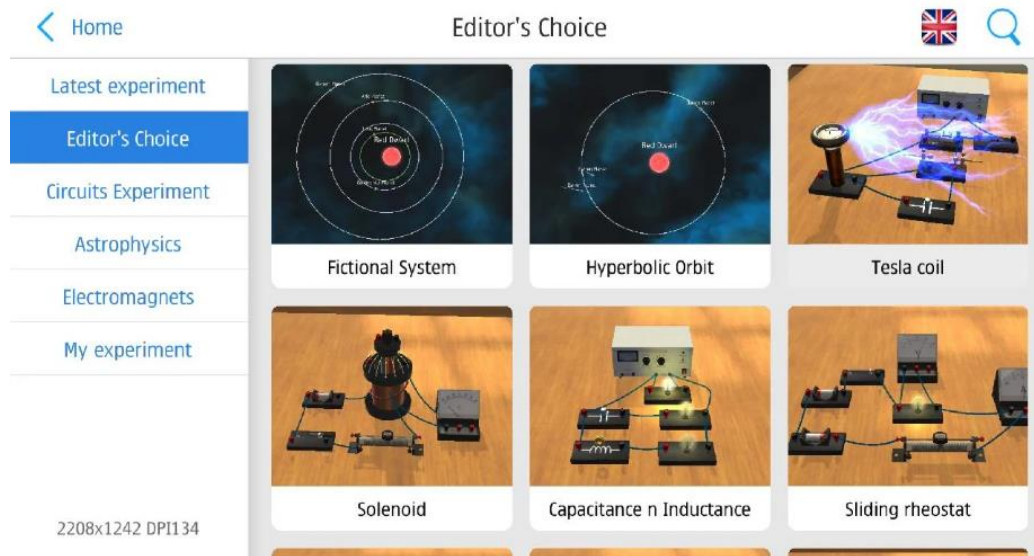


Рисунок 1.2 – Інтерфейс застосунку «Physics Lab» [1].

Проаналізувавши застосунок можна зробити висновок, що цей програмний продукт має ряд переваг, а саме:

- зручний та зрозумілий користувацький інтерфейс;
- інтерактивні 3D анімації.

Також можна виділити декілька недоліків:

- не створює документів для друку;
- високий поріг входження;
- не має довідки для використання стендів.

Отже, застосунків, які в повному обсязі можуть задовольнити потребу виконання лабораторних робіт з фізики нами не були виявлені, тому створення даного програмного продукту буде актуальним і матиме своє місце на ринку програмного забезпечення.

1.3 Формулювання актуальності і завдань роботи

Після аналізу наявних рішень, можна зробити висновок, що розробка системи для організації дистанційного навчального процесу з фізики є актуальною, оскільки жоден з розглянутих варіантів не відповідає в повному обсязі вимогам шкільної програми з фізики та можливостей, які потребують учні 10-11 класів. Розробка сучасного застосунку з інтерактивними анімаціями є затребуваною та дозволить учням комфортно навчатися у дистанційному режимі.

Головним завданням під час розробки програмного забезпечення є створення системи, за допомогою якої можна виконувати лабораторні роботи з фізики.

Метою дипломного проекту є збереження та підвищення якості навчання у галузі фізики у складних умовах.

Аналіз предметної області показав, що майбутній застосунок повинен мати наступні можливості:

- знайомити користувача з методичними рекомендаціями виконання лабораторних робіт;
- для різних видів лабораторних робіт мати унікальні інтерактивні анімації;
- надавати можливість користувачу переглядати його виконані роботи;
- зберігати дані користувача;
- ділитися результатами роботи;
- виконувати резервне збереження даних користувача;
- створювати документи для друку, а саме звіти з результатами обчислень.

Додатково програма повинна володіти наступними функціональними можливостями: перевіряти вхідні дані на коректність та відповідність умовам, мати зручний та зрозумілий інтерфейс, мати підказки на відповідних кроках розрахунків.

Виділені завдання:

1. Розробити систему для організації дистанційного навчального процесу.
2. Розробити блок-схеми алгоритмів для операцій:
 - 2.1. Реєстрації нового учня.
 - 2.2. Виконання та поширення лабораторних робіт.
 - 2.3. Перевірки лабораторних робіт.
 - 2.4. Резервне копіювання лабораторних робіт.
3. Розробити інтерфейс системи.
4. Розробити інструкцію користувача.

2 ПРОЕКТНИЙ РОЗДІЛ

2.1 Проектування інформаційної системи

Провівши аналіз предметної області, було вирішено розробити локальну архітектуру застосунку з профілем учня. На даному етапі проекту. Дані будуть зберігатися локально, на пристрої користувача.

Така архітектура має деякі переваги, наприклад відсутність потреби у підключенні до мережі та додаткова безпека даних, але має недоліки, такі як незручність розповсюдження результатів лабораторної роботи та неможливість відновити дані з хмари.

В подальшому наявне рішення буде доповнене файл-серверною архітектурою застосунку, за допомогою якої з'явиться можливість створити багато-користувацький режим роботи та додати профіль вчителя.

Діаграма варіантів використання зображена на рисунку 2.1.

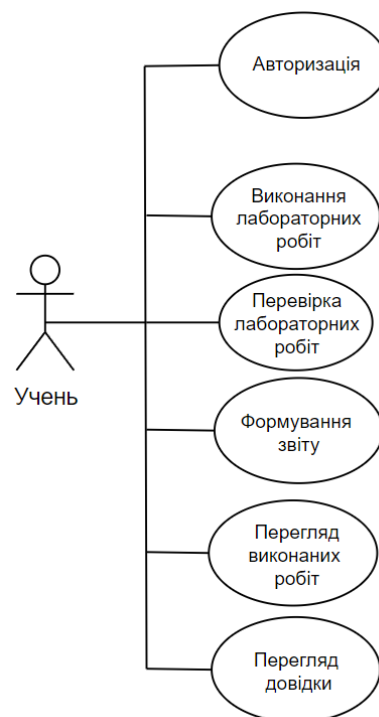


Рисунок 2.1 – Діаграма варіантів використання

Розглянувши діаграму стає зрозуміло, що основним користувачем системи буде учень. Для нього наявна можливість авторизації для підпису

лабораторних робіт. В подальшому буде розроблений профіль вчителя, який допоможе зручніше перевіряти роботи та вести облік успішності учнів.

Для учнів виділені такі можливості як:

- перегляд виконаних робіт;
- перегляд підказок приблизного результату обчислень;
- перегляд довідки;
- виконання лабораторних робіт;
- формування звіту;
- авторизація.

Отже, розроблена діаграма варіантів використання охоплює всі можливі функціональні можливості які будуть надані користувачу.

2.2 Розробка структурної та функціональної схеми програмного комплексу

Для подальшого розвитку будь-якого ПЗ потрібно розробити структурну та функціональну схему програмного комплексу, які визначають алгоритми, що виконуються у системі та як вони взаємодіють одне з одним. Це допоможе зрозуміти, як вона буде працювати та виконувати поставлені завдання.

Схема зображена на рисунку 2.2

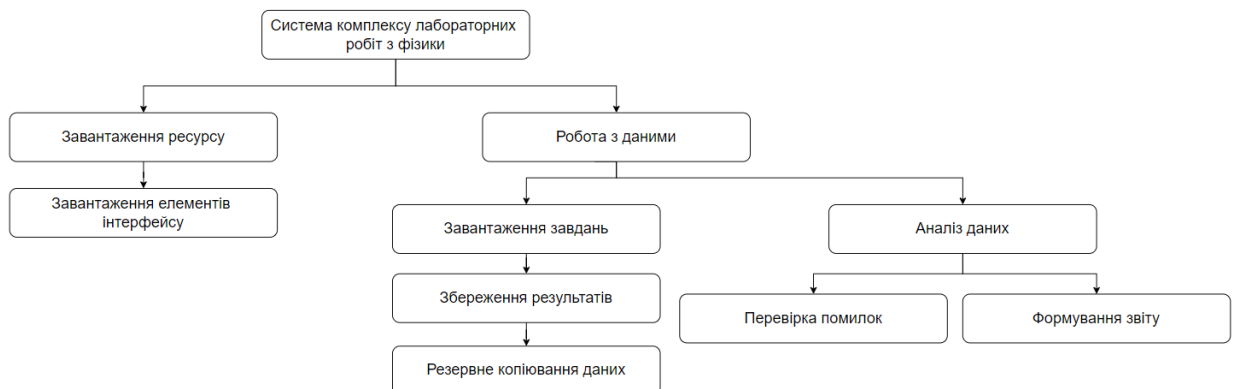


Рисунок 2.2 – Структурна схема програмного комплексу

На рисунку 2.3 зображена контекстна функціональна схема, що була розроблена на основі функціональної та структурної схеми.

Вона допоможе забезпечити розуміння загальної архітектури системи.

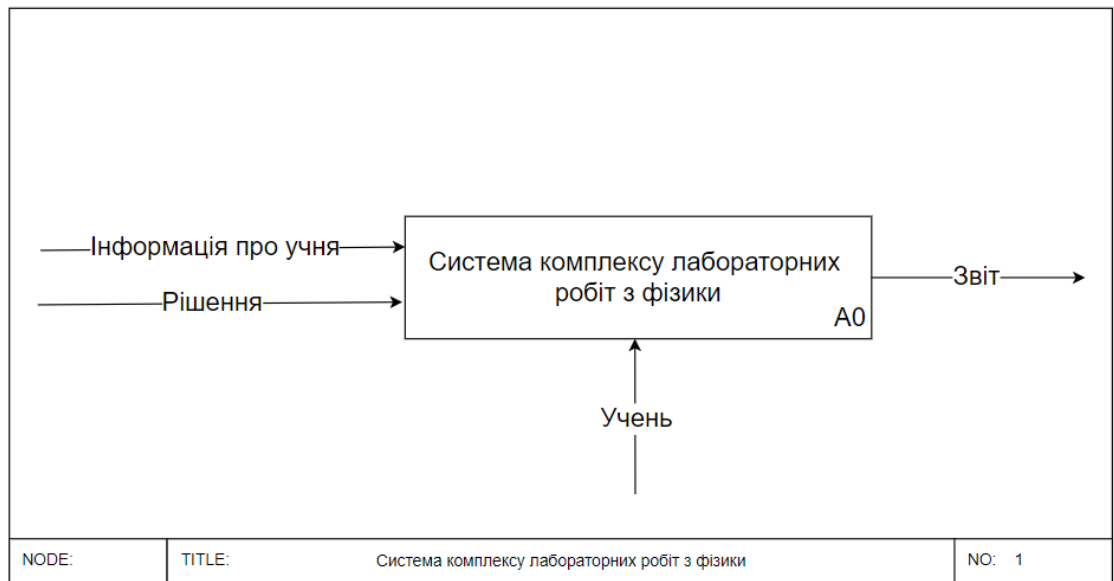


Рисунок 2.3 – Контекстна функціональна схема

На вхід вводяться дані користувача: інформація про учня та його обчислення у лабораторній роботі. Вихідні дані представлені у вигляді звіту, який згенерований програмою. Створений звіт зберігається локально, на пристрої користувача, яким можна поділитися у зручний для себе спосіб.

На рисунку 2.4 зображена декомпозиція контекстної функціональної схеми програмного комплексу, яка описує роботу програми більш детально.

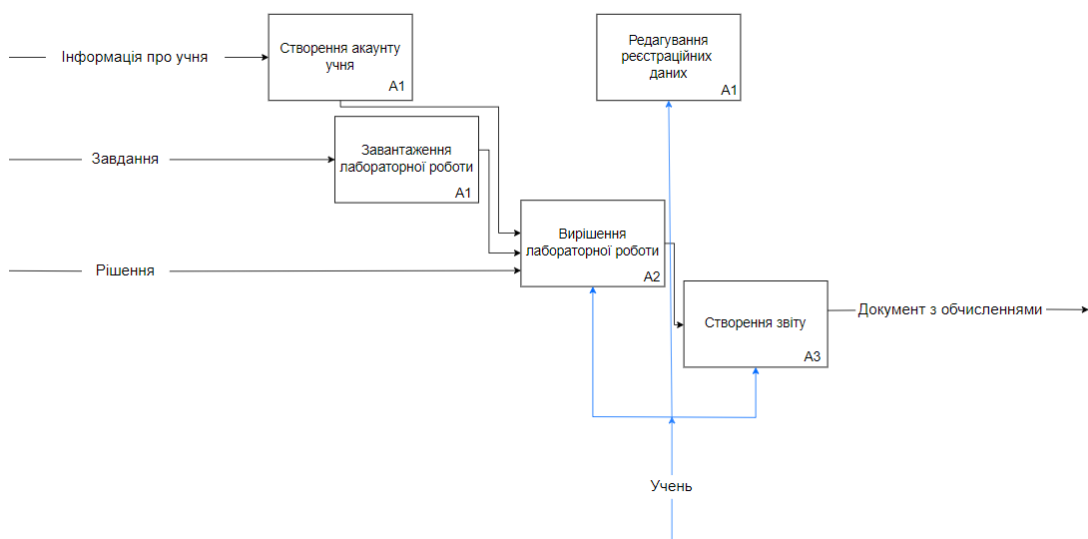


Рисунок 2.4 – Декомпозиція контекстної функціональної схеми

Спочатку вводяться дані від учня для підпису лабораторних робіт, він має змогу завантажити та виконати обчислення, необхідні для відкритої лабораторної роботи. Наступним кроком в функціональній схемі є створення учнем звіту, яким можна ділитися у будь-які соціальні мережі. Також буде передбачена можливість змінити реєстраційні дані.

2.3 Розробка блок-схем основних алгоритмів програмного комплексу

Нами було створено блок-схеми основних алгоритмів програмного застосунку. На рисунку 2.5 зображена блок-схема, яка описує введення даних про нового учня у систему. Після вводу та натискання кнопки «Продовжити» відбувається валідація введеної інформації через маску для вводу, чи відповідає ім'я та прізвище наданим програмою стандартом. Наступним кроком дані зберігаються у програмі, а учень переходить на головну сторінку з лабораторними роботами.

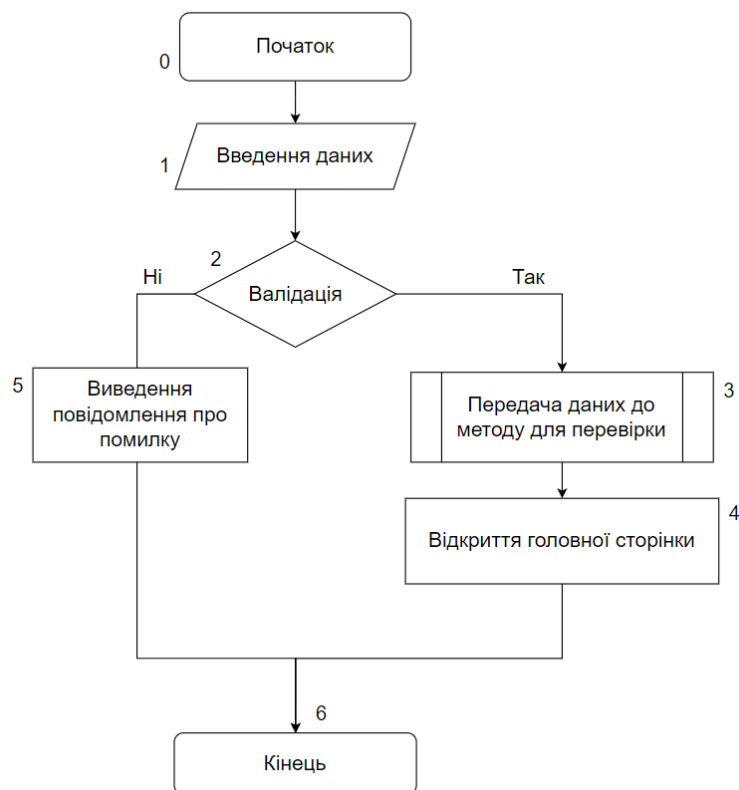


Рисунок 2.5 – Блок-схема реєстрації нового учня

Розглянемо блок-схему перевірки, яка зображена на рисунку 2.6, вона описує перевірку введених даних у лабораторній роботі. Після вводу даних та натискання кнопки «Перевірити» відбувається перевірка даних. Якщо число у полі не входить у допустимий діапазон значень, воно змінюється на червоний колір. Перевірка носить рекомендаційний характер, тому створити звіт можна не дотримуючись норм застосунку. Остаточне рішення про оцінку вносить вчитель, після того як отримав документ з результатами.

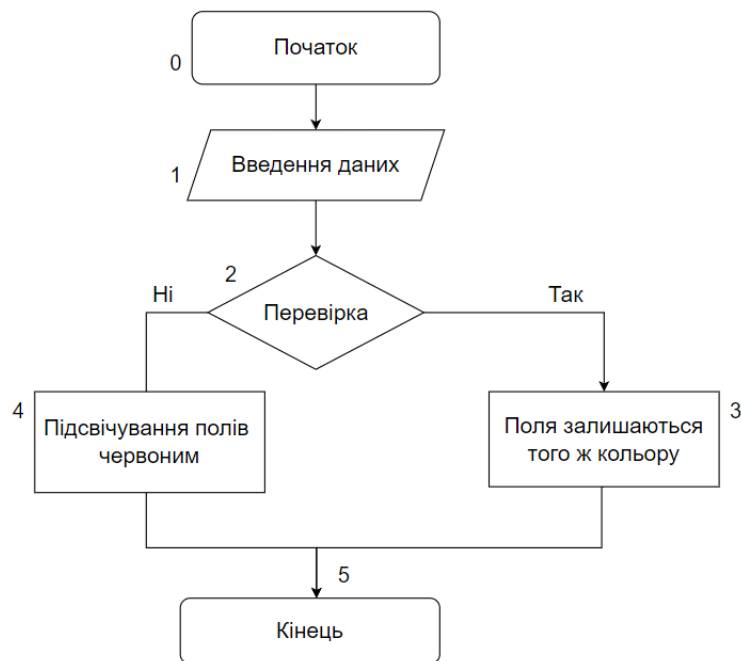


Рисунок 2.6 – Блок-схема перевірки лабораторної роботи

Розглянемо блок-схему виконання лабораторної роботи користувача на рисунку 2.7. Спочатку учень переглядає доступні для нього роботи на головній сторінці, після чого йому потрібно відкрити завдання та провести обчислення, потім заповнити відповідні поля. Створення звіту відбувається за будь-яких умов надаючи учню більше свободи, у тому випадку, якщо він не бажає виконувати усі запропоновані обчислення.



Рисунок 2.7 – Блок-схема виконання лабораторної роботи

Розглянемо блок-схему резервного копіювання лабораторних робіт, яка зображена на рисунку 2.8.

Резервне копіювання створюється автоматично, одразу після того, як був згенерований звіт. Це допоможе підвищити безпеку та зберегти роботу учня.



Рисунок 2.8 – Блок-схема резервного копіювання даних

Отже, нами були розглянуті блок-схеми основних алгоритмів майбутнього програмного забезпечення комплексу віртуальних лабораторій з фізики.

2.4 Розробка моделі візуального інтерфейсу програмного комплексу

Після проведення проектування інформаційної системи та розробки блок-схем, нами було створено моделі майбутнього візуального інтерфейсу користувача за допомогою функціональних можливостей програми Figma. Інтерфейс програми буде складатися з 15 форм, після запуску з'являється стартове вікно введення даних про учня, яке зображено на рисунку 2.9.

У даному вікні вводиться прізвище, ініціали користувача для подальшого підпису звіту.

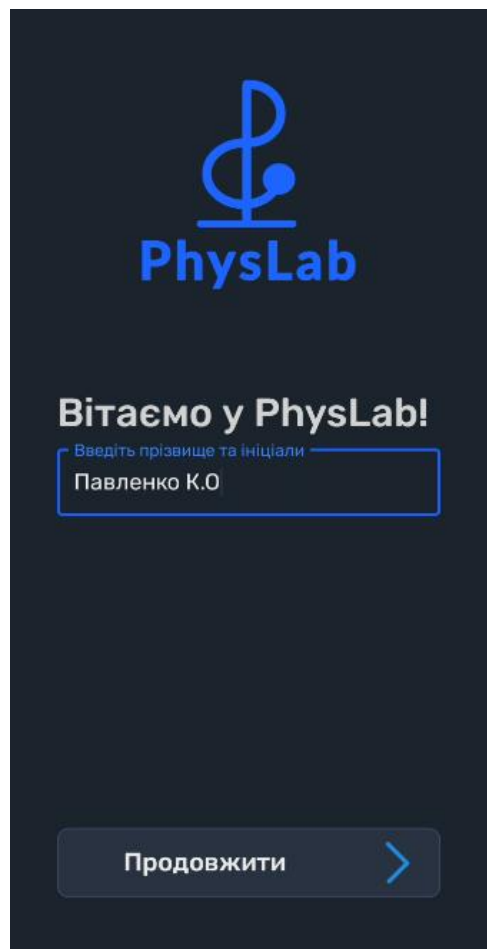


Рисунок 2.9 – модель сторінки «Авторизація»

Головна форма у режимі користувача зображена на рисунку 2.10. Дане вікно містить список усіх доступних лабораторних робіт, що містять тему, мету, інтерактивну анімацію та інструкцію до виконання, яка може бути корисною для розрахунків у програмі. Для того, щоб відкрити лабораторну роботу треба натиснути на кнопку з її назвою.

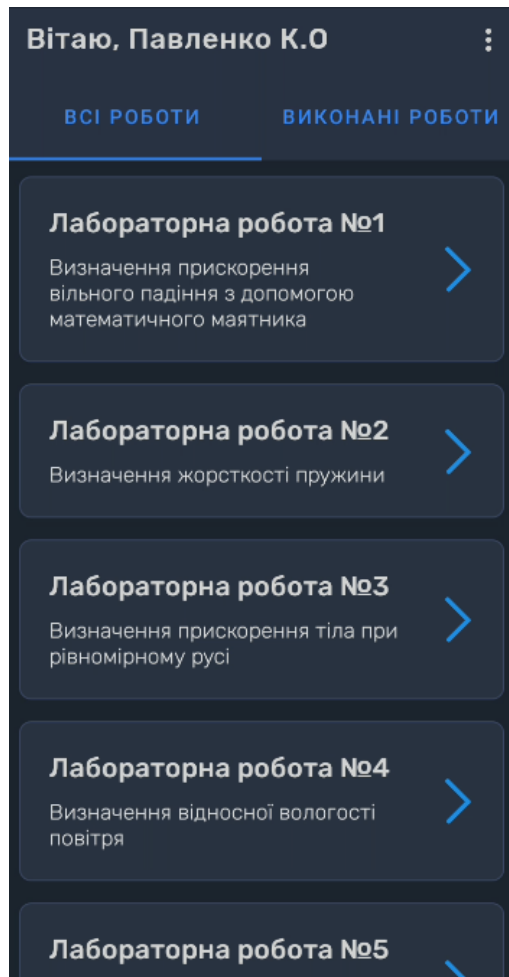


Рисунок 2.10 – модель сторінки «Головна»

Після натискання кнопки «Виконані роботи» відкриється вікно зі списком виконаних робіт, що зображене на рисунку 2.11. Також це вікно надає змогу поділитися створеним звітом після натискання кнопки «Поділитися»

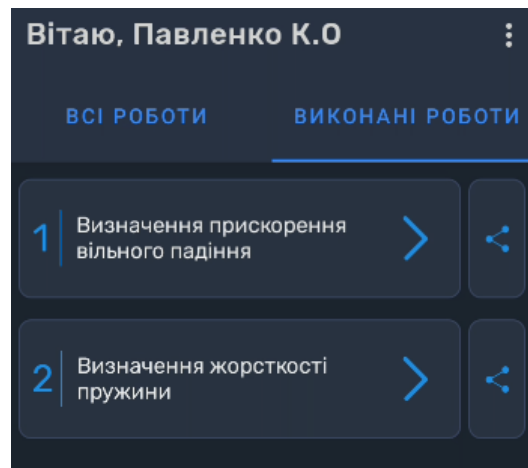


Рисунок 2.11 – модель сторінки «Виконані роботи»

На сторінці з лабораторною роботою, користувачу надається вся потрібна інформація для її виконання. Одна робота складається з 3 умовних частин. Перша надає користувачу інформацію про тему, мету лабораторної роботи. Нижче йде детальна інструкція, вказівки щодо виконання у розділі «Хід роботи», ці елементи зображені на рисунку 2.12.

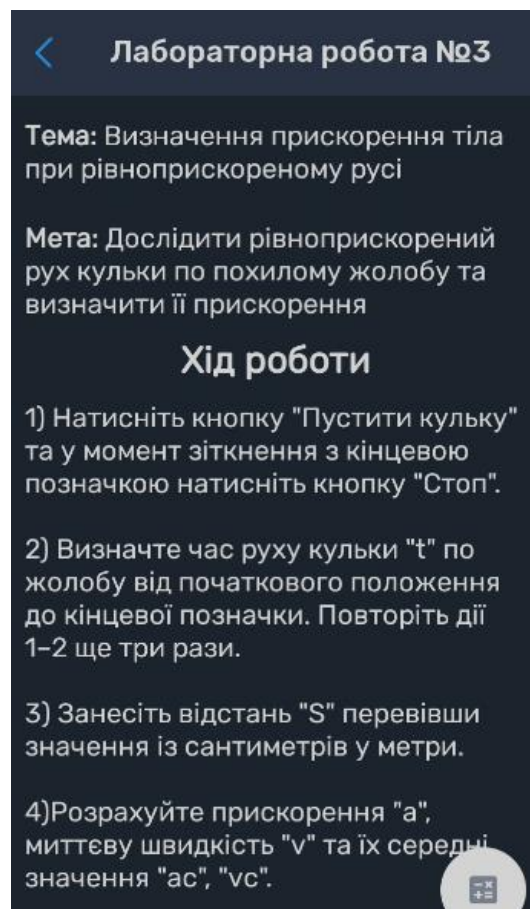


Рисунок 2.12 – перша частина моделі сторінки «Лабораторна робота»

Друга складається з унікальної інтерактивної анімації та кнопки керування, які потрібні для виконання лабораторної роботи, що зображені на рисунку 2.13.



Рисунок 2.13 – друга частина моделі сторінки «Лабораторна робота»

Третя містить у собі поля, які необхідні для занесення обчислювальних даних. Напроти них міститься формула за якою потрібно розрахувати значення, номер поля відповідає номеру досліду. Також присутні 2 кнопки: для створення звіту «Створити звіт», та виявлення помилок «Перевірити»

У разі неправильних розрахунків кнопка «Перевірити» зробить поле з помилкою в обчисленні червоним. Поля для введення обчислень зображені на рисунку 2.14.

Прискорення | $a = \frac{2s}{t}$

1. _____ 2. _____ 3. _____ 4. _____

Миттєва швидкість | $v = \frac{s}{t}$

1. _____ 2. _____ 3. _____ 4. _____

Сер. знач. прискорення | $a_c = \frac{a_1 + a_2 + a_3 + a_4}{4}$

1. _____

Сер. знач. миттєвої швидкості | $v_c = \frac{v_1 + v_2 + v_3 + v_4}{4}$

1. _____

Створити звіт Перевірити

Рисунок 2.14 – третя частина моделі сторінки «Лабораторна робота»

На сторінці з лабораторною роботою присутня кнопка для відкриття форми з калькулятором для обчислень, що зображена на рисунку 2.15.

1. _____

Пр

0 ✕

7 8 9 \div

4 5 6 \times

1 2 3 $-$

. 0 \pm $+$

1 $=$

ОЧИСТИТИ НАЗАД ОК

Рисунок 2.15 – модель вікна «Калькулятор»

Після натискання кнопки «Довідка» з'являється форма, на якій міститься інформація про можливості програми, розробника, детальна інформація щодо інтерфейсу додатку, інструкція користування анімаціями та кнопками. Модель представлена на рисунку 2.16.



Рисунок 2.16 – модель вікна «Довідка»

Отже, нами були розглянуті основні моделі майбутніх вікон для розробки програмного забезпечення комплексу віртуальних лабораторій з фізики.

3 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

3.1 Аналіз обраного середовища програмування

Згідно виділених завдань, обране середовище програмування повинно підтримувати розробку застосунків для мобільних пристроїв та мати наступні можливості:

- інструменти для вводу та виводу інформації;
- інструменти для роботи з анімаціями;
- інструменти для створення звіту.

Перераховані інструменти мають наступні середовища:

1. Eclipse for Android Developers - потужне програмне забезпечення для розробки додатків на Android, яке було створено IBM. Це надійне та гнучке середовище розробки з такими функціями як підтримка Gradle, контроль версій програмного забезпечення та велика бібліотека плагінів, що розширюють можливості середовища.

2. XCode - це IDE від Apple, яку розробники можуть використовувати для створення коду для iPhone, iPad та інших пристроїв цієї компанії. IDE підтримує розробку коду з використанням мови програмування Swift. Також є можливість розширити підтримку до декількох мов програмування.

Це офіційне середовище, яке можна використовувати для створення застосунків, що будуть запускатися тільки на пристроях компанії Apple.

3. Android Studio - має підтримку Google та JetBrains. З моменту свого запуску у 2013 році воно стало основним середовищем для розробників Android. Інтелектуальний редактор коду Android Studio включає складні функції, такі як автоматичне завершення коду та аудит у реальному часі.

Емулятор Android - одна з багатьох функцій, що підтримуються IDE, яка також працює з популярними платформами контролю версій, такими як GitHub і Git. Крім того, він сумісний з такими мовами, як Java та Kotlin.

Порівняємо ці технології та зробимо висновки.

Eclipse має широкі можливості для розробки мобільних застосунків, але не має підтримки сучасних бібліотек, які необхідні для реалізації задуманих ідей. Крім того, середовище не користується широкою популярністю серед розробників та великих компаній, які готові його підтримувати, що ускладнює користування цим програмним засобом.

XCode включає тісну інтеграцію з екосистемою пристроїв Apple, зручний інтерфейс користувача, потужні інструменти для налагодження та тестування, а також підтримку мов програмування Swift та Objective-C. Однак, одним з головних недоліків є те, що XCode призначений виключно для платформ Apple. Наш застосунок націлений на широку аудиторію, що включає користувачів Android-пристроїв, тому ми обрали розробку під Android.

Android Studio включає різний інструментарій, що забезпечує комфортні умови роботи на різних етапах розробки програми. Підтримка великих компаній, широка спільнота розробників, сучасні інструменти, емуляція операційної системи Android, яка дозволяє переглядати результат виконаної роботи прямо на робочій машині, робить цей програмний продукт найкращим вибором для програмування мобільних застосунків. В протизагагу можна зазначити, що IDE споживає значну кількість ресурсів комп'ютера.

Проаналізувавши мінуси і плюси середовищ програмування, враховуючи поставлені задачі проекту, для розробки застосунку нами було обрано середовище Android Studio.

Android Studio найбільш доцільне середовище програмування для реалізації нашого мобільного застосунку, так як XCode та Eclipse for Android Developers мають недоліки, які не дозволяють реалізувати бажані можливості.

Отже, для розробки мобільного застосунку, нами було обрано таке середовище програмування Android Studio, яке має повний набір інструментів для реалізації завдання та задовольняє наші потреби. Мовою програмування є Java.

3.2 Програмна реалізація основних функцій

Для застосунку потрібна анімація, яка б дозволяла наочно демонструвати властивості фізичних процесів, тому для її реалізації нами буде використано бібліотеку Lottie, яка містить у собі компонент LottieAnimationView, за допомогою якого можна програмувати відтворення анімації за допомогою методів:

- setProgress() – допомагає керувати тривалістю анімації;
- start() – дає старт анімації;
- stop() – дозволяє припинити анімацію;
- resolveKeyPath() - приймає ключовий шлях і повертає список з нуля або більше існуючих ключових шляхів.

Наведемо програмну реалізацію функції:

```
public void buttonStart(View view) {
    LottieAnimationView anim = findViewById(R.id.animationView);
    //встановлення шляху до папки з ресурсами
    anim.setImageAssetsFolder("raw/");
    //оголошення лічильника
    k++;
    //перевірка значення лічильника
    if (k % 3 == 0) {
        //встановлення та програвання анімації lr23 кожні 3
натискання
        anim.setAnimation(R.raw.lr23);
        anim.playAnimation();
    } else {
        //перемикання між анімаціями lr21 та lr22
        if (flag) {
            anim.setAnimation(R.raw.lr21);
        } else {
            anim.setAnimation(R.raw.lr22);
        }
        anim.playAnimation();
        flag = !flag;}}}
```

Для створення графічних інтерфейсів за допомогою Android Studio застосовуються Activity, в .xml файл яких можна помістити такі елементи керування як button, imageView, actionBar, tabBar, listView для взаємодії користувача з програмою та інтерактивними графічними елементами.

Також для створення анімації буде використана програма Adobe After Effects з подальшим портуванням до середовища Android Studio.

Програма містить великий спектр можливостей та анімаційних експресій, які використовуються для надання більш природнього ефекту рухам об'єктів. Серед необхідних анімаційних експресій можемо відзначити:

- drop bounce expression – використовується для імітації інерції об'єктів;
- wiggle expression – використовується для імітації пружинних об'єктів;

Для їх програмування використовуються такі методи:

- velocityAtTime() – допомагає керувати швидкістю експресії;
- math.sin() – обчислення синусу для розрахунку поведінки експресії;
- math.exp() – повертає натуральний логарифм для обчислень

положення;

- nearestKey() – повертає найближчий об'єкт ключового кадру.

Для створення звіту у форматі .docx, буде використано бібліотеку POI-XWPF за допомогою таких методів:

- setText() – для запису тексту у файл;
- addBreak() – надання відступів між рядками;
- setFontSize() – встановлення розміру тексту;
- createTable() – створення таблиці для запису результатів дослідів;
- write() – зберігання всіх налаштувань у .docx файл;

Наведемо програмну реалізацію цієї функції:

```
public void buttonCreate(View view) {
    filePath = new File(getExternalFilesDir(null),
        "Jp1_"+loadstring+".docx"); //надання імені документу та його
    створення
    try {
```

```

        if (!filePath.exists()){ //якщо файлу не існує, то створюється
        новий
        filePath.createNewFile();}
    } catch (IOException e) {
    e.printStackTrace(); //у разі виявлення помилки повідомити її код
    }
    try {
    XWPFDocument xwpfDocument = new XWPFDocument(); //створення
    документу та надання параметрів
    XWPFParagraph xwpfParagraph = xwpfDocument.createParagraph();
    XWPFRun xwpfRun = xwpfParagraph.createRun();
    XWPFParagraph paragraph = xwpfDocument.createParagraph();
    paragraph.setAlignment(ParagraphAlignment.CENTER); //розташування
    тексту по центру
    XWPFRun run = paragraph.createRun();
    xwpfRun.setText("Лабораторна робота №1"); //занесення номеру лр
    до документу
    xwpfRun.addBreak(); //надання відступів між рядками

```

Підключення бібліотеки до середовища програмування Android Studio відбудуватиметься через Gradle.

Отже, нами були розглянута програмна реалізація основних функцій нашого програмного забезпечення, особливості керування анімацією в Android Studio, основні методи програмування анімації та вихідної документації, можливості для створення графічного інтерфейсу додатків.

3.3 Методика роботи користувача

Під час першого запуску, користувача вітає стартове вікно програми в якому пропонується ввести прізвище та ініціали для підпису лабораторних робіт. Після чого, учню доступні 7 лабораторних робіт на вибір.

Кожна лабораторна робота дає змогу ознайомитися з інструкцією до виконання, обчислення та керування анімацією. Кінцевим результатом є заповнені відповідні поля з розрахунками та згенерований звіт у форматі .docx. Користувач має змогу поділитися документом через спеціальне меню. Також

пропонується можливість скористатися додатковими функціями програми, такі як калькулятор або перевірка даних на коректність.

Для зручності, нами було вирішено створити довідкову систему в застосунку, це допоможе користувачу отримати корисну інформацію про роботу.

Зміст довідки наступний:

- загальні відомості;
- про програму;
- про розробника;
- запуск програми;
- інтерфейс;
- можливості програми;
- друкування звітів;
- особливості роботи з анімаціями.

Для додавання довідки у застосунок потрібно виконати ряд дій у інтерфейсі середовища програмування: File – New – Activity – Empty Activity.

У результаті отримаємо файл у форматі .xml в якому заносимо потрібну інформацію.

Щоб довідкова система стала доступна користувачу необхідно:

- додати кнопку на форму;
- для виклику довідки за допомогою кнопки необхідно створити метод

та надати його потрібній кнопці, приклад коду:

```
Intent intent = new Intent(this, dovidka.class);  
startActivity(intent);
```

Отже, нами було спроектовано та реалізовано довідкову систему програми, розібрано методика роботи користувача.

ВИСНОВОК

Мобільний застосунок «Комплекс віртуальних лабораторій з фізики» розроблений для здобувачів освіти 10-11 класів та призначений для автоматизації процесу виконання лабораторних робіт з фізики та зберігання якості навчання у складних умовах.

В режимі користувача програма виконує такі функції:

- знайомить користувача з рекомендаціями виконання розрахунків;
- створює звіт для друку;
- дає рекомендації стосовно обчислень;
- створює резервне копіювання даних.

Вимоги до програмного забезпечення:

- операційна система Android 8 або новіші версії.

Для роботи з програмою необхідно встановити APK файл.

Для отримання пояснень, щодо роботи застосунку користувач може скористатися довідковою системою, що викликається в програмі.

Отже, нами була проаналізована предметна область, розроблені схеми, інтерфейс додатку, на основі яких було створено клієнтську частину застосунку, яка дозволить здобувачам освіти 10-11 класів виконувати лабораторні роботи з фізики.

ПЕРЕЛІК ПОСИЛАНЬ

1. Курніаван Е. Android Application Development. New York, 2016. 320 с.
2. Дарсі М., Кондер С. Sams Teach Yourself Android Application Development in 24 Hours. Indianapolis, 2013. 432 с.
3. Блекберн А. Android Development Patterns: Best Practices for Professional Developers. Boston, 2016. 400 с.
4. ДіМарціо Д. Ф. Beginning Android Programming with Android Studio. Indianapolis, 2016. 432 с.
5. Меєр М. Professional Android 4 Application Development. Indianapolis, 2012. 576 с.
6. Філіпс Б. Android Programming: The Big Nerd Ranch Guide, 2nd Edition. Atlanta, 2015. 600 с.
7. Флінг Б. Mobile Design and Development. Sebastopol, 2009. 368 с.
8. Бар'яхтар В.Г., Довгий С.О., Божинова Ф.Я. Фізика 10 клас, Харків, 2018. 271 с.
9. Дженсен В., Ерландссон Н., Бурманн У., Опаріна Б. Use of Visual Learning Media to Increase Student Learning Motivation, Австралія, 2022. 17 с.

Додаток А

Код програми

Стартова форма:

```

public class MainActivity extends AppCompatActivity {
    ListView listView;
    Toolbar toolbar;
    final static String SHARED_NAME_STRING="sharedp";
    final static String USER_NAME="prizv";
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        SectionsPagerAdapter sectionsPagerAdapter = new
        SectionsPagerAdapter(this, getSupportFragmentManager());
        ViewPager viewPager = findViewById(R.id.view_pager);
        viewPager.setAdapter(sectionsPagerAdapter);
        ActionBar();
        TabLayout tabs = findViewById(R.id.tabs);
        tabs.setupWithViewPager(viewPager);
        SharedPreferences prefs = getSharedPreferences("prefs",
        MODE_PRIVATE);
        boolean firstStart = prefs.getBoolean("firstStart", true);
        if(firstStart ){
            name();
        }
        ActivityCompat.requestPermissions(this, new
        String[]{Manifest.permission.READ_EXTERNAL_STORAGE,
        Manifest.permission.WRITE_EXTERNAL_STORAGE},
        PackageManager.PERMISSION_GRANTED);
    }
}

```

```

}
@Override
public boolean onCreateOptionsMenu(Menu menu) {
// Inflate the menu this adds items to the action bar if it is present.
getMenuInflater().inflate(R.menu.menu_main, menu);
return true;
}
@Override
public boolean onOptionsItemSelected(MenuItem item) {
int id = item.getItemId();
if (id == R.id.action_settings) {
Intent intent = new Intent(this, reg.class);
startActivity(intent);
return true;
}
if (id == R.id.action_dovid) {
Intent intent = new Intent(this, dovidka.class);
startActivity(intent);
return true;
}
return super.onOptionsItemSelected(item);
}
@SuppressWarnings("ResourceAsColor")
public void ActionBar() {
toolbar = (Toolbar) findViewById(R.id.topAppBar);
TypedValue typedValue = new TypedValue();
getTheme().resolveAttribute(R.attr.colorSecondary, typedValue, true);
int color2 = typedValue.data;
SharedPreferences prefs =
getSharedPreferences(SHARED_NAME_STRING, MODE_PRIVATE);

```

```

String loadstring = prefs.getString(USER_NAME, "");
toolbar.setTitle(Html.fromHtml("<font      face      =      'rubik-medium'
color="" + color2 + "">Вітаю, " + loadstring + " </font>"));
toolbar.getOverflowIcon().setColorFilter(color2,
PorterDuff.Mode.SRC_ATOP);
setSupportActionBar(toolbar);
}
private void name(){
SharedPreferences      prefs      =      getSharedPreferences("prefs",
MODE_PRIVATE);
SharedPreferences.Editor editor = prefs.edit();
editor.putBoolean("firstStart", false);
editor.apply();
Intent intent = new Intent(this, reg.class);
startActivity(intent);
}}

```

Форма зі списком лабораторних робіт:

```

public class tab1contacts extends Fragment {
ListView listView;

String[] catNames = new String[] {
    "Лабораторна робота №1",
    "Лабораторна робота №2",
    "Лабораторна робота №3",
    "Лабораторна робота №4",
    "Лабораторна робота №5",
    "Лабораторна робота №6",
    "Лабораторна робота №7",
    "Лабораторна робота №8",
    "Лабораторна робота №9",
    "Лабораторна робота №10"
}
}

```

```

};
String[]description = new String[] {
    "Визначення прискорення вільного падіння з допомогою
математичного маятника",
    "Визначення жорсткості пружини",
    "Визначення прискорення тіла при рівномірному русі",
    "Визначення відносної вологості повітря",
    "Визначення поверхневого натягу рідини",
    "Визначення питомого опору провідника",
    "Визначення послідовного і паралельного з'єднання
провідників",
    "Визначення ЕРС і внутрішнього опору джерела струму",
    "З'ясування умови рівноваги важення",
    "Визначення фокусної відстані й оптичної сили лінзи"
};
@Override
public View onCreateView( LayoutInflater inflater, ViewGroup container,
Bundle savedInstanceState) {
    View rootView = inflater.inflate(R.layout.tab1contacts,container,false);
    listView = rootView.findViewById(R.id.listView);
    ProgramAdapter          programAdapter          =new
ProgramAdapter(getContext(),catNames,description);
    listView.setAdapter(programAdapter);
    listView.setOnItemClickListener(new
AdapterView.OnItemClickListener() {
        @Override
        public void onItemClick(AdapterView<?> parent, View view, int
position, long id) {
            if (position == 0) {
                Intent intent = new Intent(getContext(), lab1.class);

```

```
        startActivity(intent);
    }
    if (position == 1)
        Intent intent = new Intent(getApplicationContext(), lab2.class);
        startActivity(intent);
    }
    if (position == 2) {
        Intent intent = new Intent(getApplicationContext(), lab3.class);
        startActivity(intent);
    }
    if (position == 3) {
        Intent intent = new Intent(getApplicationContext(), lab4.class);
        startActivity(intent);
    }
    if (position == 4) {
        Intent intent = new Intent(getApplicationContext(), lab5.class);
        startActivity(intent);
    }
    if (position == 5) {
        Intent intent = new Intent(getApplicationContext(), lab6.class);
        startActivity(intent);
    }
    if (position == 6) {
        Intent intent = new Intent(getApplicationContext(), lab7.class);
        startActivity(intent);}
    if (position == 7) {
        Intent intent = new Intent(getApplicationContext(), lab8.class);
        startActivity(intent);
    }
    if (position ==8) {
```

```

        Intent intent = new Intent(getContext(), lab9.class);
        startActivity(intent);
    }
    if (position == 9) {
        Intent intent = new Intent(getContext(), lab10.class);
        startActivity(intent);
    } }); return rootView; } }

```

Форма для списку виконаних робіт:

@Override

```

    public View onCreateView( LayoutInflater inflater, ViewGroup container,
    Bundle savedInstanceState) {
        View rootView = inflater.inflate(R.layout.tab2contacts,container,false);
        rowlab1 = (TableRow) rootView.findViewById(R.id.rowlab1);
        rowlab2 = (TableRow) rootView.findViewById(R.id.rowlab2);
        rowlab3 = (TableRow) rootView.findViewById(R.id.rowlab3);
        rowlab4 = (TableRow) rootView.findViewById(R.id.rowlab4);
        button32 = (Button)rootView.findViewById(R.id.button32);
        button33 = (Button)rootView.findViewById(R.id.button33);
        button34 = (Button)rootView.findViewById(R.id.button34);
        button35 = (Button)rootView.findViewById(R.id.button35);
        share1 = (ImageButton)rootView.findViewById(R.id.share1);
        share2 = (ImageButton)rootView.findViewById(R.id.share2);
        share3 = (ImageButton)rootView.findViewById(R.id.share3);
        share4 = (ImageButton)rootView.findViewById(R.id.share4);
        button32.setOnClickListener(this);
        button33.setOnClickListener(this);
        button34.setOnClickListener(this);
        button35.setOnClickListener(this);
        share1.setOnClickListener(this);
        share2.setOnClickListener(this);
    }

```

```

        share3.setOnClickListener(this);
        share4.setOnClickListener(this);
        save("i1",rowlab1);
        save("i2",rowlab2);
        save("i3",rowlab3);
        return rootView;}

public SharedPreferences save(String name, TableRow rownam){
    SharedPreferences prefs =
getContext().getSharedPreferences(SHARED_NAME_STRING,
MODE_PRIVATE);
    String loadstring = prefs.getString(name, "");
    if(loadstring.equals("0")){
rownam.setVisibility(Integer.parseInt(loadstring));}
return prefs;}

@Override
public void onClick(View v) {
    SharedPreferences prefs =
getContext().getSharedPreferences(SHARED_NAME_STRING,
MODE_PRIVATE);
    String loadstring = prefs.getString(USER_NAME, "");
    switch (v.getId()) {
        case R.id.button32: labs(getContext(),lab1.class); break;
        case R.id.button33: labs(getContext(),lab2.class); break;
        case R.id.button34: labs(getContext(),lab3.class); break;
        case R.id.share1:  shar(getActivity(),"JIp1_"+loadstring+".docx");
break;
        case R.id.share2:  shar(getActivity(),"JIp2_"+loadstring+".docx");
break;
        case R.id.share3:  shar(getActivity(),"JIp3_"+loadstring+".docx");
break;

```

```

    }
}
public Intent labs(Context context, Class lab){
    Intent intent = new Intent(context, lab);
    startActivity(intent);
    return intent;
}
public Intent shar(Context context, String lname){
    File path = new File(Environment.getExternalStorageDirectory(),
"Android/data/com.example.phyz/files");
    File filelocation = new File(path, lname);
    Intent shareIntent = new Intent(Intent.ACTION_SEND);
    shareIntent.setType("application/pdf");
    Uri fileUri = FileProvider.getUriForFile(context, "com.example.phyz",
filelocation);
    shareIntent.putExtra(Intent.EXTRA_STREAM, fileUri);
    Intent chooser = Intent.createChooser(shareIntent, "Share File");
    List<ResolveInfo> resInfoList =
context.getPackageManager().queryIntentActivities(chooser,
PackageManager.MATCH_DEFAULT_ONLY);
    for (ResolveInfo resolveInfo : resInfoList) {
        String packageName = resolveInfo.activityInfo.packageName;
        context.grantUriPermission(packageName, fileUri,
Intent.FLAG_GRANT_WRITE_URI_PERMISSION|
Intent.FLAG_GRANT_READ_URI_PERMISSION);
    }
    startActivity(chooser);
    return shareIntent;
}
}

```


Форма для введення прізвища, ініціалів користувача:

```
@Override
```

```
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_reg);
    getSupportActionBar().hide();
    button4 = (Button)findViewById(R.id.button4);
    button4.setOnClickListener(this);
}
```

```
@Override
```

```
public void onClick(View view) {
    fieldnam =findViewById(R.id.fieldnam);
    nam=fieldnam.getText().toString();
    String stringToSave = nam;
    SharedPreferences.Editor editor =
getSharedPreferences(SHARED_NAME_STRING,
MODE_PRIVATE).edit();
    editor.putString(USER_NAME, stringToSave);
    editor.apply();
    Intent intent = new Intent(this, MainActivity.class);
    startActivity(intent);
}
```

Адаптер для списку лабораторних робіт:

```
public class ProgramAdapter extends ArrayAdapter<String> {
    Context context;
    String[]programName;
    String[]programDescription;
    public ProgramAdapter(Context context, String[]programName,
String[]programDescription) {
        super(context, R.layout.row, R.id.textView1, programName);
```

```

        this.context = context;
        this.programName=programName;
        this.programDescription=programDescription;
    }
    @Override
    public View getView(int position, View convertView, ViewGroup parent)
    {
        View singleItem=convertView;
        ProgramViewHolder holder=null;
        View view;
        if(singleItem==null){
            LayoutInflater inflater = (LayoutInflater)
context.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
            singleItem=inflater.inflate(R.layout.row,parent, false);
            holder = new ProgramViewHolder(singleItem);
            singleItem.setTag(holder);
        }
        else{
            holder= (ProgramViewHolder) singleItem.getTag();
        }
        holder.programTitle.setText(programName[position]);
        holder.programDescription.setText(programDescription[position]);
        return singleItem;
    }
}

```

Форма для довідки:

```

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_dovidka);
}

```

```

        ActionBar();
        slidr = Slidr.attach(this);
    }
    public void ActionBar() {
        ActionBar actionBar = getSupportActionBar();
        actionBar.setHomeAsUpIndicator(R.drawable.ic_array2);
        TypedValue typedValue = new TypedValue();
        getTheme().resolveAttribute(R.attr.colorPrimary, typedValue, true);
        int color = typedValue.data;
        getTheme().resolveAttribute(R.attr.colorSecondary, typedValue, true);
        int color2 = typedValue.data;
        actionBar.setTitle(Html.fromHtml("<font face = 'rubik-medium'
color="+color2+">Довідка</font>"));
        ColorDrawable colorDrawable = new ColorDrawable(color);
        actionBar.setBackgroundDrawable(colorDrawable);
        if (actionBar != null) {
            actionBar.setDisplayHomeAsUpEnabled(true);
            actionBar.setDisplayHomeAsUpEnabled(true);
        }
    }
}

```

Форма лабораторної роботи №1

@Override

```

protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_lab1);
    editText1 = findViewById(R.id.editTextNumberDecimal49);
    editText2 = findViewById(R.id.editTextNumberDecimal50);
    editText3 = findViewById(R.id.editTextNumberDecimal51);
    editText4 = findViewById(R.id.editTextNumberDecimal52);
    editText11 = findViewById(R.id.editTextNumberDecimal53);
}

```

```
editText22 = findViewById(R.id.editTextNumberDecimal54);
editText33 = findViewById(R.id.editTextNumberDecimal55);
editText44 = findViewById(R.id.editTextNumberDecimal56);
editText111 = findViewById(R.id.editTextNumberDecimal57);
editText222 = findViewById(R.id.editTextNumberDecimal58);
editText333 = findViewById(R.id.editTextNumberDecimal59);
editText444 = findViewById(R.id.editTextNumberDecimal60);
editTextKolyv = findViewById(R.id.editTextNumberDecimal61);
editText1111 = findViewById(R.id.editTextNumberDecimal62);
editText2222 = findViewById(R.id.editTextNumberDecimal63);
editText3333 = findViewById(R.id.editTextNumberDecimal64);
editText4444 = findViewById(R.id.editTextNumberDecimal65);
editTextSerzn = findViewById(R.id.editTextNumberDecimal66);
editTextPoxybk = findViewById(R.id.editTextNumberDecimal67);
editText1.setOnClickListener(this);
editText2.setOnClickListener(this);
editText3.setOnClickListener(this);
editText4.setOnClickListener(this);
editText11.setOnClickListener(this);
editText22.setOnClickListener(this);
editText33.setOnClickListener(this);
editText44.setOnClickListener(this);
editText111.setOnClickListener(this);
editText222.setOnClickListener(this);
editText333.setOnClickListener(this);
editText444.setOnClickListener(this);
editText1111.setOnClickListener(this);
editText2222.setOnClickListener(this);
editText3333.setOnClickListener(this);
editText4444.setOnClickListener(this);
```

```

editTextPoxybk.setOnClickListener(this);
editTextSerzn.setOnClickListener(this);
editTextKolyv.setOnClickListener(this);
titleText = findViewById(R.id.titleText);
descriptionText = findViewById(R.id.descriptionText);
slidr = Slidr.attach(this);
ActionBar();
WebView();
SharedPreferences prefs =
getSharedPreferences(SHARED_NAME_STRING, MODE_PRIVATE);
loadstring = prefs.getString(USER_NAME, "");
ActivityCompat.requestPermissions(this, new
String[]{Manifest.permission.READ_EXTERNAL_STORAGE,
Manifest.permission.WRITE_EXTERNAL_STORAGE},
PackageManager.PERMISSION_GRANTED);
if (position == 0) { titleText.setText(R.string.lab1_tema);
descriptionText.setText(R.string.lab1_meta); }
scrollView = findViewById(R.id.scrollView);
fabAddOitransferIn = findViewById(R.id.fabAddOitransferIn);
scrollView.getViewTreeObserver().addOnScrollChangeListener(new
ViewTreeObserver.OnScrollChangeListener() {
@Override
public void onScrollChanged() {
if (scrollView.getScrollY() > previousScrollY &&
fabAddOitransferIn.getVisibility() == View.VISIBLE) {
fabAddOitransferIn.hide();
} else if (scrollView.getScrollY() < previousScrollY &&
fabAddOitransferIn.getVisibility() != View.VISIBLE) {
fabAddOitransferIn.show();
}
}
}

```

```

        previousScrollY = scrollView.getScrollY();
    }
});

}

public void WebView(){
    webView = findViewById(R.id.webView);
    webView.setBackgroundColor(0);
    webView.setVerticalScrollBarEnabled(false);
    webView.setHorizontalScrollBarEnabled(false);
    webView.setBackgroundResource(R.drawable.forweb);
    webView.getSettings().setJavaScriptEnabled(true);
    webView.loadUrl("file:///android_asset/index.html");
}

@SuppressWarnings("ResourceAsColor")
public void check(View view){
    try {
        TypedValue typedValue = new TypedValue();
        getTheme().resolveAttribute(R.attr.colorSecondary, typedValue, true);
        int color2 = typedValue.data;

        double t1= Double.parseDouble(editText1.getText().toString());
        double t2= Double.parseDouble(editText2.getText().toString());
        double t3= Double.parseDouble(editText3.getText().toString());
        double t4= Double.parseDouble(editText4.getText().toString());
        double l1= Double.parseDouble(editText11.getText().toString());
        double l2= Double.parseDouble(editText22.getText().toString());
        double l3= Double.parseDouble(editText33.getText().toString());
        double l4= Double.parseDouble(editText44.getText().toString());
        double T1= Double.parseDouble(editText111.getText().toString());
        double T2= Double.parseDouble(editText222.getText().toString());
    }
}

```

```

double T3= Double.parseDouble(editText333.getText().toString());
double T4= Double.parseDouble(editText444.getText().toString());
double g1= Double.parseDouble(editText1111.getText().toString());
double g2= Double.parseDouble(editText2222.getText().toString());
double g3= Double.parseDouble(editText3333.getText().toString());
double g4= Double.parseDouble(editText4444.getText().toString());
double p= Double.parseDouble(editTextPoxybk.getText().toString());
double gser= Double.parseDouble(editTextSerzn.getText().toString());
int kolyv = parseInt(editTextKolyv.getText().toString());
    if ((t1 >= 46.10 && t1 <= 49.99)) { editText1.setTextColor(color2); }
    else { editText1.setTextColor(Color.parseColor("#FF0000")); }
    if ((t2 >= 41.6 && t2 <= 43.99)) { editText2.setTextColor(color2); }
    else { editText2.setTextColor(Color.parseColor("#FF0000")); }
    if ((t3 >= 35.1 && t3 <= 37.99)) { editText3.setTextColor(color2); }
    else { editText3.setTextColor(Color.parseColor("#FF0000")); }
    if ((t4 >= 28.1 && t4 <= 30.99)) { editText4.setTextColor(color2); }
    else { editText4.setTextColor(Color.parseColor("#FF0000")); }
    if (l1==1.4) editText1111.setTextColor(color2);
    else editText1111.setTextColor(Color.parseColor("#FF0000"));
    if (l2==1.1) editText2222.setTextColor(color2);
    else editText2222.setTextColor(Color.parseColor("#FF0000"));
    if (l3==0.8) editText3333.setTextColor(color2);
    else editText3333.setTextColor(Color.parseColor("#FF0000"));
    if (l4==0.5) editText4444.setTextColor(color2);
    else editText4444.setTextColor(Color.parseColor("#FF0000"));
    if ((T1 >= 33.5 && T1 <= 33.9)) { editText1111.setTextColor(color2); }
    else { editText1111.setTextColor(Color.parseColor("#FF0000")); }
    if ((T2 >= 37.2 && T2 <= 37.9)) { editText2222.setTextColor(color2); }
    else { editText2222.setTextColor(Color.parseColor("#FF0000")); }
    if ((T3 >= 1.77 && T3 <= 1.79)) { editText3333.setTextColor(color2); }

```

```

else { editText333.setTextTextColor(Color.parseColor("#FF0000")); }
if ((T4 >= 1.43 && T4 <= 1.49)) { editText444.setTextTextColor(color2); }
else { editText444.setTextTextColor(Color.parseColor("#FF0000")); }
if (kolyv == 20) editTextKolyv.setTextTextColor(color2);
else editTextKolyv.setTextTextColor(Color.parseColor("#FF0000"));
if ((g1 >= 0.03 && g1 <= 0.09)) { editText1111.setTextTextColor(color2); }
else { editText1111.setTextTextColor(Color.parseColor("#FF0000")); }
if ((g2 >= 0.02 && g2 <= 0.04)) { editText2222.setTextTextColor(color2); }
else { editText2222.setTextTextColor(Color.parseColor("#FF0000")); }
if ((g3 >= 10.06 && g3 <= 10.09)) { editText3333.setTextTextColor(color2);
}
else { editText3333.setTextTextColor(Color.parseColor("#FF0000")); }
if ((g4 >= 10.06 && g4 <= 10.09)) { editText4444.setTextTextColor(color2);
}
else { editText4444.setTextTextColor(Color.parseColor("#FF0000")); }
if ((gser >= 5.02 && gser <= 5.08)) {
editTextSerzn.setTextTextColor(color2); }
else { editTextSerzn.setTextTextColor(Color.parseColor("#FF0000")); }
if ((p >= 48.50 && p <= 48.54)) { editTextPoxybk.setTextTextColor(color2);
}
else { editTextPoxybk.setTextTextColor(Color.parseColor("#FF0000")); }
} catch (Exception e) {
    Toast toast = Toast.makeText(getApplicationContext(),
        "Ви ввели не всі поля!", Toast.LENGTH_SHORT);
    toast.show();
}
}
}
@SuppressLint("NonConstantResourceId")
@Override
public void onClick(View v) {

```



```
TypedValue typedValue = new TypedValue();
getTheme().resolveAttribute(R.attr.colorSecondary, typedValue, true);
int color2 = typedValue.data;
switch (v.getId()) {
    case R.id.editTextNumberDecimal49:
editText1.setTextColor(color2); break;
    case R.id.editTextNumberDecimal50:
editText2.setTextColor(color2); break;
    case R.id.editTextNumberDecimal51:
editText3.setTextColor(color2); break;
    case R.id.editTextNumberDecimal52:
editText4.setTextColor(color2); break;

    case R.id.editTextNumberDecimal53:
editText11.setTextColor(color2); break;
    case R.id.editTextNumberDecimal54:
editText22.setTextColor(color2); break;
    case R.id.editTextNumberDecimal55:
editText33.setTextColor(color2); break;
    case R.id.editTextNumberDecimal56:
editText44.setTextColor(color2); break;

    case R.id.editTextNumberDecimal57:
editText111.setTextColor(color2); break;
    case R.id.editTextNumberDecimal58:
editText222.setTextColor(color2); break;
    case R.id.editTextNumberDecimal59:
editText333.setTextColor(color2); break;
    case R.id.editTextNumberDecimal60:
editText444.setTextColor(color2); break;
```

```
        case R.id.editTextNumberDecimal61:
editTextKolyv.setTextTextColor(color2); break;

        case R.id.editTextNumberDecimal62:
editText1111.setTextTextColor(color2); break;
        case R.id.editTextNumberDecimal63:
editText2222.setTextTextColor(color2); break;
        case R.id.editTextNumberDecimal64:
editText3333.setTextTextColor(color2); break;
        case R.id.editTextNumberDecimal65:
editText4444.setTextTextColor(color2); break;
        case R.id.editTextNumberDecimal66:
editTextSerzn.setTextTextColor(color2); break;
        case R.id.editTextNumberDecimal67:
editTextPoxybk.setTextTextColor(color2); break;
    }
}
@Override
public boolean onOptionsItemSelected(@NonNull MenuItem item) {
    switch (item.getItemId()) {
        case android.R.id.home:
            this.finish();
            return true;
    }
    return super.onOptionsItemSelected(item);
}
public void ActionBar() {
    ActionBar actionBar = getSupportActionBar();
    actionBar.setHomeAsUpIndicator(R.drawable.ic_array2);
```

```

TypedValue typedValue = new TypedValue();
getTheme().resolveAttribute(R.attr.colorPrimary, typedValue, true);
int color = typedValue.data;
getTheme().resolveAttribute(R.attr.colorSecondary, typedValue, true);
int color2 = typedValue.data;
actionBar.setTitle(Html.fromHtml("<font face = 'rubik-medium'
color="" + color2 + "">Лабораторна работа №1</font>"));
ColorDrawable colorDrawable = new ColorDrawable(color);
actionBar.setBackgroundDrawable(colorDrawable);
if (actionBar != null) {
    actionBar.setDisplayHomeAsUpEnabled(true);
    actionBar.setDisplayShowHomeEnabled(true);
}
}

public void buttonCreate(View view) {
    filePath = new File(getExternalFilesDir(null),
"Лп1_" + loadstring + ".docx");
    try {
        if (!filePath.exists()){
            filePath.createNewFile();
        }
    } catch (IOException e) {
        e.printStackTrace();
    }
    try {
        XWPFDocument xwpfDocument = new XWPFDocument();
        XWPFParagraph xwpfParagraph =
xwpfDocument.createParagraph();
        XWPFRun xwpfRun = xwpfParagraph.createRun();
        XWPFParagraph paragraph = xwpfDocument.createParagraph();

```

```

paragraph.setAlignment(ParagraphAlignment.CENTER);
XWPFRun run = paragraph.createRun();
xwpfRun.setText("Лабораторна робота №1");
xwpfRun.addBreak();
xwpfRun.setText("Тема: Визначення прискорення вільного
падіння за допомогою математичного маятника");
xwpfRun.addBreak();
xwpfRun.setText("Мета: Вивчення закону руху математичного
маятника, експериментальне визначення прискорення вільного падіння
за допомогою цього маятника");
run.setText("Хід роботи");
run.setFontSize(14);
xwpfRun.setFontSize(14);
XWPFTable table = xwpfDocument.createTable(5,8);
String[][] arrStr = new String[5][8];
arrStr[0][0]="№ Досліду";
arrStr[0][1]="Час коливань";
arrStr[0][2]="Довжина нитки";
arrStr[0][3]="Період коливань";
arrStr[0][4]="Кількість коливань";
arrStr[0][5]="Прискорення";
arrStr[0][6]="Сер.знач. прискорення";
arrStr[0][7]="Відносна похибка";
arrStr[1][0]="1";
arrStr[2][0]="2";
arrStr[3][0]="3";
arrStr[4][0]="4";
arrStr[1][1]= String.valueOf(editText1.getText());
arrStr[2][1]= String.valueOf(editText2.getText());
arrStr[3][1]= String.valueOf(editText3.getText());

```

```

arrStr[4][1]= String.valueOf(editText4.getText());
arrStr[1][2]= String.valueOf(editText11.getText());
arrStr[2][2]= String.valueOf(editText22.getText());
arrStr[3][2]= String.valueOf(editText33.getText());
arrStr[4][2]= String.valueOf(editText44.getText());
arrStr[1][3]= String.valueOf(editText111.getText());
arrStr[2][3]= String.valueOf(editText222.getText());
arrStr[3][3]= String.valueOf(editText333.getText());
arrStr[4][3]= String.valueOf(editText444.getText());
arrStr[1][4]= String.valueOf(editTextKolyv.getText());
arrStr[2][4]= String.valueOf(editTextKolyv.getText());
arrStr[3][4]= String.valueOf(editTextKolyv.getText());
arrStr[4][4]= String.valueOf(editTextKolyv.getText());
arrStr[1][5]= String.valueOf(editText1111.getText());
arrStr[2][5]= String.valueOf(editText2222.getText());
arrStr[3][5]= String.valueOf(editText3333.getText());
arrStr[4][5]= String.valueOf(editText4444.getText());
arrStr[1][6]= String.valueOf(editTextSerzn.getText());
arrStr[2][6]= String.valueOf(editTextSerzn.getText());
arrStr[3][6]= String.valueOf(editTextSerzn.getText());
arrStr[4][6]= String.valueOf(editTextSerzn.getText());
arrStr[1][7]= String.valueOf(editTextPoxybk.getText());
arrStr[2][7]= String.valueOf(editTextPoxybk.getText());
arrStr[3][7]= String.valueOf(editTextPoxybk.getText());
arrStr[4][7]= String.valueOf(editTextPoxybk.getText());
for(int i=0; i< arrStr.length; i++){
    for(int j=0;j < arrStr[i].length; j++){
table.getRow(i).getCell(j).setText(arrStr[i][j]); } }
    FileOutputStream fileOutputStream = new
FileOutputStream(filePath);

```

```

xwpfDocument.write(fileOutputStream);
if (fileOutputStream != null) {
    fileOutputStream.flush();
    fileOutputStream.close();
}
xwpfDocument.close();
} catch (Exception e) {
    e.printStackTrace();
}
Toast toast = Toast.makeText(getApplicationContext(),
    "Звіт створено", Toast.LENGTH_SHORT);
toast.show();

File path = new File(Environment.getExternalStorageDirectory(),
"Android/data/com.example.phyz/files");

File filelocation = new File(path, "Лп1_" + loadstring + ".docx");
Intent shareIntent = new Intent(Intent.ACTION_SEND);
shareIntent.setType("application/pdf");
Uri fileUri = FileProvider.getUriForFile(this, "com.example.phyz",
filelocation);

shareIntent.putExtra(Intent.EXTRA_STREAM, fileUri);
Intent chooser = Intent.createChooser(shareIntent, "Share File");
List<ResolveInfo> resInfoList =
this.getPackageManager().queryIntentActivities(chooser,
PackageManager.MATCH_DEFAULT_ONLY);
for (ResolveInfo resolveInfo : resInfoList) {
    String packageName = resolveInfo.activityInfo.packageName;
    this.grantUriPermission(packageName, fileUri,
Intent.FLAG_GRANT_WRITE_URI_PERMISSION
Intent.FLAG_GRANT_READ_URI_PERMISSION);}
startActivity(chooser);

```

```

        String stringToSave = "0";
        SharedPreferences.Editor editor =
getSharedPreferences(SHARED_NAME_STRING,
MODE_PRIVATE).edit();
        editor.putString(USER_NAME_STRING, stringToSave);
        editor.apply();
        tab2contacts.rowlab1.setVisibility(View.VISIBLE);
    }
    public void butcalc(View view){
        calcDialog.getSettings().setInitialValue(value);
        calcDialog.show(getSupportFragmentManager(), "calc_dialog");
        calcDialog.getSettings().setExpressionShown(true);
    }
    @Nullable
    private BigDecimal value = null;
    @Override
    public void onValueEntered(int requestCode, @Nullable BigDecimal
value) {
        this.value = value;
    }
}

```

Форма для анімацій:

```

public class CustomWebview extends WebView {
    public CustomWebview(Context context) {
        super(context);
    }
    public CustomWebview(Context context, AttributeSet attrs) {
        super(context, attrs);
    }
    public CustomWebview(Context context, AttributeSet attrs, int defStyle) {

```

```

        super(context, attrs, defStyle);
    }
    @Override
    public boolean onTouchEvent(MotionEvent event){
        requestDisallowInterceptTouchEvent(true);
        return super.onTouchEvent(event);
    }
}

```

Адаптер для секій головного меню:

```

public class SectionsPagerAdapter extends FragmentPagerAdapter {
    @StringRes
    private static final int[] TAB_TITLES = new int[]{R.string.tab_text_1,
R.string.tab_text_2};
    private final Context mContext;
    public SectionsPagerAdapter(Context context, FragmentManager fm) {
        super(fm);
        mContext = context;
    }
    @Override
    public Fragment getItem(int position) {
switch (position){
    case 0:
        tab1contacts tab1 = new tab1contacts();
        return tab1 =new tab1contacts();
    case 1:
        tab2contacts tab2 = new tab2contacts();
        return tab2 =new tab2contacts();
    default: return null;
}
}
}

```



```

@Nullable
@Override
public CharSequence getPageTitle(int position) {
    return mContext.getResources().getString(TAB_TITLES[position]);
}
@Override
public int getCount() {
    // Show 2 total pages.
    return 2;
}
}

```

Форма лабораторної роботи №2

```

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_lab2);
    editTextP=findViewById(R.id.editTextNumber31);
    editTextK1=findViewById(R.id.editTextNumber32);
    editTextK2=findViewById(R.id.editTextNumber2);
    editTextK3=findViewById(R.id.editTextNumber3);
    editText1 = findViewById(R.id.editTextNumberDecimal4);
    editText2 = findViewById(R.id.editTextNumberDecimal3);
    editText3 = findViewById(R.id.editTextNumberDecimal2);
    editText11 = findViewById(R.id.editTextNumberDecimal7);
    editText22 = findViewById(R.id.editTextNumberDecimal6);
    editText33 = findViewById(R.id.editTextNumberDecimal5);
    editText111 = findViewById(R.id.editTextNumberDecimal10);
    editText222 = findViewById(R.id.editTextNumberDecimal9);
    editText333 = findViewById(R.id.editTextNumberDecimal8);
    editText1111 =findViewById(R.id.editTextNumberDecimal11);
}

```

```

editText1.setOnClickListener(this);
editText2.setOnClickListener(this);
editText3.setOnClickListener(this);
editText11.setOnClickListener(this);
editText22.setOnClickListener(this);
editText33.setOnClickListener(this);
editText111.setOnClickListener(this);
editText222.setOnClickListener(this);
editText333.setOnClickListener(this);
editTextP.setOnClickListener(this);
editTextK1.setOnClickListener(this);
editTextK2.setOnClickListener(this);
editTextK3.setOnClickListener(this);
editText1111.setOnClickListener(this);
ActionBar();
slidr = Slidr.attach(this);
SharedPreferences prefs =
getSharedPreferences(SHARED_NAME_STRING, MODE_PRIVATE);
loadstring = prefs.getString(USER_NAME, "");
scrollView = findViewById(R.id.scrollView);
fabAddOitransferIn = findViewById(R.id.fabAddOitransferIn);
scrollView.getViewTreeObserver().addOnScrollChangeListener(new
ViewTreeObserver.OnScrollChangeListener() {
    int previousScrollY = 0;
    @Override
    public void onScrollChanged() {
        if (scrollView.getScrollY() > previousScrollY &&
fabAddOitransferIn.getVisibility() == View.VISIBLE) {
            fabAddOitransferIn.hide();

```

```

        } else if (scrollView.getScrollY() < previousScrollY &&
fabAddOItransferIn.getVisibility() != View.VISIBLE) {
            fabAddOItransferIn.show();
        }
        previousScrollY = scrollView.getScrollY();
    }
});
}

```

```
@SuppressWarnings("ResourceAsColor")
```

```

public void check(View view){
    try {
        TypedValue typedValue = new TypedValue();
        getTheme().resolveAttribute(R.attr.colorSecondary, typedValue,
true);
        int color2 = typedValue.data;
if ((lp==31)) { editTextP.setTextColor(color2); }
        else { editTextP.setTextColor(Color.parseColor("#FF0000")); }
if ((k1==43)) { editTextK1.setTextColor(color2); }
        else { editTextK1.setTextColor(Color.parseColor("#FF0000")); }
if ((k2 == 56)) { editTextK2.setTextColor(color2); }
        else { editTextK2.setTextColor(Color.parseColor("#FF0000")); }
if ((k3 == 64)) { editTextK3.setTextColor(color2); }
        else { editTextK3.setTextColor(Color.parseColor("#FF0000")); }
if (f1==1) editText1.setTextColor(color2);
        else editText1.setTextColor(Color.parseColor("#FF0000"));
if (f2==2) editText2.setTextColor(color2);
        else editText2.setTextColor(Color.parseColor("#FF0000"));

```

```

    if (f3==3) editText3.setTextColor(color2);
    else editText3.setTextColor(Color.parseColor("#FF0000"));
    if ((x1==0.012)) { editText11.setTextColor(color2); }
    else { editText11.setTextColor(Color.parseColor("#FF0000")); }
    if ((x2==0.025)) { editText22.setTextColor(color2); }
    else { editText22.setTextColor(Color.parseColor("#FF0000")); }
    if ((x3==0.033)) { editText33.setTextColor(color2); }
    else { editText33.setTextColor(Color.parseColor("#FF0000")); }
    if ((j1==83)) { editText111.setTextColor(color2); }
    else{ editText111.setTextColor(Color.parseColor("#FF0000")); }
    if ((j2==80)) { editText222.setTextColor(color2); }
    else { editText222.setTextColor(Color.parseColor("#FF0000")); }
    if ((j3==90)) { editText333.setTextColor(color2); }
    else { editText333.setTextColor(Color.parseColor("#FF0000")); }
    if ((kc== 84.3)) { editText1111.setTextColor(color2); }
    else { editText1111.setTextColor(Color.parseColor("#FF0000")); }
} catch (Exception e) {
    Toast toast = Toast.makeText(getApplicationContext(),
        "Ви ввели не всі поля!", Toast.LENGTH_SHORT);
    toast.show();
}
}

public void buttonStart(View view){
    LottieAnimationView anim = (LottieAnimationView)
findViewById(R.id.animationView);
    anim.setImageAssetsFolder("raw/");
    k++;
}

```

```

    if (flag) {
        anim.setAnimation(R.raw.lr21);
        anim.playAnimation();
        flag = false;
    } else {
        anim.setAnimation(R.raw.lr22);
        anim.playAnimation();
        flag = true; }
    if(k%3==0){anim.setAnimation(R.raw.lr23);
    anim.playAnimation();}
}

@Override
public boolean onOptionsItemSelected(@NonNull MenuItem item) {
    switch (item.getItemId()) {
        case android.R.id.home:
            this.finish();
            return true;
    }
    return super.onOptionsItemSelected(item);
}

@SuppressWarnings("ResourceAsColor")
public void ActionBar() {
    ActionBar actionBar = getSupportActionBar();
    actionBar.setHomeAsUpIndicator(R.drawable.ic_array2);
    TypedValue typedValue = new TypedValue();
    getTheme().resolveAttribute(R.attr.colorPrimary, typedValue, true);
    int color = typedValue.data;
}

```

```
getTheme().resolveAttribute(R.attr.colorSecondary, typedValue, true);
int color2 = typedValue.data;
actionBar.setTitle(Html.fromHtml("<font face = 'rubik-medium'
color='"+color2+"'>Лабораторна робота №2</font>"));
ColorDrawable colorDrawable = new ColorDrawable(color);
actionBar.setBackgroundDrawable(colorDrawable);
if (actionBar != null) {
    actionBar.setDisplayHomeAsUpEnabled(true);
    actionBar.setDisplayShowHomeEnabled(true);
}
}
```

Додаток Б

Диск з програмою