Міністерство освіти і науки України

Департамент освіти і науки Дніпропетровської облдержадміністрації

Дніпропетровське територіальне відділення МАН України

Відділення: комп’ютерні науки

Секція: Мультимедійні системи, навчальні та

ігрові програми

РОЗРОБКА МОБІЛЬНОЇ НАВЧАЛЬНОЇ ПРОГРАМИ ДЛЯ ОБЧИСЛЕННЯ ЕЛЕКТРОННИХ СХЕМ НА ПЛАТФОРМІ JAVA

Роботу виконав:   
Фоменко Олег Володимирович,

учень 10 класу КЗО «Дніпропетровський

обласний ліцей-інтернат фізико-математичного профілю»

Наукові керівники:

Безрукава Оксана Григорівна,

вчитель інформатики КЗО *«*Дніпропетровський обласний ліцей-інтернат фізико-математичного профілю»

Ільченко Ірина Григорівна

вчитель інформатики КЗО *«*Дніпропетровський обласний ліцей-інтернат фізико-математичного профілю»

Дніпро-2018

ЗМІСТ

[ПЕРЕЛІК ТЕРМІНІВ ТА УМОВНИХ СКОРОЧЕНЬ 3](#_Toc504398230)

[ВСТУП 4](#_Toc504398231)

[РОЗДІЛ 1. АНАЛІЗ ІСНУЮЧИХ ПРОГРАМ І ПОСТАНОВКА ЗАДАЧІ 6](#_Toc504398232)

[1.1 Дослідження існуючих програм 6](#_Toc504398233)

[1.2. Постановка задачі 7](#_Toc504398234)

[РОЗДІЛ 2. РОЗРОБКА МОБІЛЬНОГО ДОДАТКУ 8](#_Toc504398235)

[2.1. Обґрунтування вибору мови програмування 8](#_Toc504398236)

[2.2. Розробка структури 9](#_Toc504398237)

[2.3. Опис мобільного додатка 10](#_Toc504398238)

[РОЗДІЛ 3. АПРОБАЦІЯ МОБІЛЬНОГО ДОДАТОКУ 13](#_Toc504398239)

[3.1. Результат апробації роботи і перспективи розвитку 13](#_Toc504398240)

[3.2. Вимоги до мобільних пристроїв, інструкція до встановлення 13](#_Toc504398241)

[3.3. Інструкція для користувачів 14](#_Toc504398242)

[ВИСНОВКИ 18](#_Toc504398243)

[СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ 19](#_Toc504398244)

# ПЕРЕЛІК ТЕРМІНІВ ТА УМОВНИХ СКОРОЧЕНЬ

**Android Studio**  -  інтегроване середовище розробки (IDE) для платформи Android.

**Android SDK** - набір засобів розробки (SDK) від Google, створений для розробки додатків під Android. Android SDK працює з середовищами розробки Android Studio та Eclipse.

**libGDX** - це Java фреймворк, який надає крос-платформне API для розробки ігор і додатків, що працюють в режимі реального часу.

**Java**  - об'єктно-орієнтована мова програмування, випущена як основний компонент платформи Java. Синтаксис мови багато в чому походить від C та C++, тому що велика частина синтаксису була запозичена у цих мов програмування. Java програми виконуються у середовищі віртуальної машини Java. Java програми компілюються у байткод, який при виконанні інтерпретується віртуальною машиною для конкретної платформи.

**SDK** (*Software Development Kit*) - набір із засобів розробки, утиліт і документації, який дозволяє програмістам створювати прикладні програми за визначеною технологією або для певної платформи (програмної або програмно-апаратної).

**Фреймворк** - інфраструктура програмних рішень, що полегшує розробку складних систем. Спрощено дану інфраструктуру можна вважати своєрідною комплексною бібліотекою, але при цьому вона має ряд обмежень, що задають правила створення структури проекту та написання коду.

# ВСТУП

В наш час все більш популярним є навчання за допомогою електронних засобів (мобільних пристроїв, комп’ютерів та ін.). Деякі з них були схвалені Міністерством освіти і науки [для використання у загальноосвітніх навчальних закладах [4].](https://plus1s.com/wp-content/uploads/2015/12/Grif-matem.pdf)

Дуже цікавим є можливість використання мобільних додатків для навчання [5]. Це дозволяє вивчати обрані дисципліни в будь-якому місці, у зручний для користувача час. Тому використання мобільних пристроїв для навчання становиться дуже важливою частиною освітнього процесу. Вивчення шкільних предметів за допомогою смартфонів, планшетів викликає інтерес до дисципліни у учнів. До того, є можливість використання їх у позаурочний час для виконання домашніх завдань. Кількість відомих програм обмежена, до того ж їх використання часто є платним. Тому є сенс в розробці нових мобільних додатків для допомоги учням у вивченні окремих шкільних предметів.

**Актуальність.** Розробка мобільних додатків для допомоги в навчанні є актуальною задачею.

**Тема роботи.** «Розробка мобільної навчальної програми для обчислень електронних схем на платформі Java»

**Метароботи*.*** Створити програму для мобільного пристрою, яка містить можливість створювати та розраховувати електронні схеми, а також виконувати навчальні завдання.

**Об’єкт дослідження**. Мова програмування Java. Середовище розробки **Android Studio** v.2.2**.**

**Предмет дослідження.** Навчальна програма для обчислень електронних схем на платформі Java.

Програма призначена допомогти учням у вивченні теми «Електричне коло» у курсі фізики. За допомогою додатка можна моделювати та розраховувати власні схеми, або виконувати навчальні завдання. Додаток може бути цікавим для користувачів, які вивчають тему «Електричне коло» в школі, готуються до ДПА і ЗНО або мають необхідність швидко розраховувати аналогові електричні кола. До того, враховуючи, що мобільні пристрої зараз дуже поширені серед учнів, програму можна використовувати для проведення лабораторних робіт з фізики.

**Наукова новизна**. На відміну від існуючих програм, додаток має більш сприятливий для користувача інтерфейс, зручний спосіб користування, не вибагливий до технічних характеристик пристрою та орієнтований перш за все на кінцевого користувача – учня.

# РОЗДІЛ 1. АНАЛІЗ ІСНУЮЧИХ ПРОГРАМ І ПОСТАНОВКА ЗАДАЧІ

# 1.1. Дослідження існуючих програм

Проведено аналіз існуючих програм для моделювання та розрахунку електричних схем. Вдалося знайти наступні:

* **«EveryCircuit».** Мобільний додаток «EveryCircuit» є найбільш популярною програмою для розрахунку електричних схем на Google Play. В ньому реалізована можливість моделювати та розраховувати складні аналогові та цифрові електронні схеми та можливість пройти базове навчання. Основним недоліком програми є доступність для обмеженої кількості платформ (Chromium-браузери, Android та IOS), незручність використання та обмежена кількість мов (тільки англійська) [6];
* **«Circuit Solver».** Мобільний додаток **«**Circuit Solver» дає користувачу можливість моделювати та розраховувати аналогові електронні схеми. Основними недоліками програми є обмежена кількість платформ (Тільки Android), обмежена кількість мов (тільки англійська) та відсутність завдань для навчання [7];
* **«Proteus Design Suite».** Proteus Design Suite – пакет програм для проектування аналогових та цифрових електронних схем. Являє собою дуже потужний засіб для моделювання та розрахунку. Основними недоліками є платне використання, обмежена кількість платформ (тільки Windows) та переважна орієнтованість на досвідченого користувача (студенти, вчителі, тощо) [7];

Помітно, що основними мінусами розглянутих програм є недостатня локалізація та відсутність у більшості програм функцій для навчання недосвідченого у фізиці користувача. Тому є сенс в розробці мобільного додатка, в якому відсутні вищезазначені недоліки.

# 1.2. Постановка задачі

Поставлено задачу створити навчальний додаток для мобільного пристрою, в якому були б відсутні вищезазначені недоліки:

1. Програма повинна бути розроблена на кросплатформенній мові програмування, завдяки чому її можна буде використовувати під різними платформами (Windows, IOS тощо);
2. Програма має бути зручною та легкою для використання ;
3. Програма має містити завдання для допомоги та навчання користувача фізиці (курс «Електричне коло»);

Для вирішення цієї задачі необхідно зробити наступне:

1. Визначити мову програмування і середовище розробки;
2. Розробити структуру додатка;
3. Розробити алгоритм для обчислення аналогових електронних схем;
4. Розробити програму для мобільного додатка та інтерфейс;
5. Розробити завдання для навчання, тренування та перевірки знань користувача з теми «Електричне коло» шкільного курсу фізики.

# РОЗДІЛ 2. РОЗРОБКА МОБІЛЬНОГО ДОДАТКУ

# 2.1. Обґрунтування вибору мови програмування

В якості мови програмування при розробці програм для мобільних додатків в наш час найчастіше використовують Java. Це обумовлено наступними факторами:

1. Java – кросплатформенна мова програмування, додатки написані на цій мові легко переносяться на пристрої, які керуються іншими операційними системами;
2. Java – сучасна та швидка мова програмування;
3. у мові програмування Java є багато вбудованих алгоритмів які полегшують написання коду та спрощують його.
4. Java – має автоматичне керування пам’яттю, що полегшує написання коду та економить оперативну пам’ять, що робить програму менш вимогливою до ресурсів пристрою.

Java – об'єктно-орієнтована мова програмування, випущена як основний компонент платформи Java. Синтаксис мови запозичений від C та C++. Java програми запускаються у середовищі віртуальної машини Java.

Велику популярність здобули мобільні пристрої під керуванням операційної системи Android. Програми для таких пристроїв дуже зручно писати за допомогою середовище Android Studio та використовуючи фреймворк LibGDX для полегшення роботи.

**Android Studio —** це інтегроване середовище (IDE) для роботи з платформою Android, анонсоване 16 травня 2013 року на конференції Google I/O менеджером по продукції корпорації Google — Еллі Паверс (Ellie Powers). 8 грудня 2014 року компанія Google випустила перший стабільний реліз Android Studio 1.0. Android Studio, заснована на програмному забезпеченні IntelliJ IDEA від компанії JetBrains, офіційне засіб розробки Android додатків. Дане середовище розробки доступне для Windows, macOS і Linux. Мова програмування, яка використовується у цьому середовищі – це Java.

**libGDX** - це Java фреймворк, який надає крос-платформне API для розробки ігор і додатків, що працюють в режимі реального часу.  Це високопродуктивний, кросплатформний ігровий фреймворк, що в першу чергу використовується для написання ігрових рушіїв та ігор.  LibGDX надає гнучкість і дозволяє уникнути суворої методології. За допомогою даної бібліотеки можна використовувати один і той самий код як для настільних комп’ютерів, так і для мобільних систем. Бібліотека є кросплатформенною і підтримує Windows, Linux, Mac OS X, Android, та браузери, з підтримкою WebGL.

На основі вищезазначеного було вирішено в якості мови програмування обрати Java, а в якості середовища розробки - Android Studio v.2.2.

# 2.2. Розробка структури

При розробці програми у середовищі Android Studio та використовуючи фреймворк libGdx дотримуються наступної структуризації:

1. будь яка програма являє собою набір класів – наслідників від State-класу;
2. кожен State-клас містить у собі набір полів та методів які реалізують відображення контенту та обчислення подій;
3. всі активні State-класи знаходяться у структурі даних стек у класі управління StateManager.java. У ньому відбувається запуск роботи поточного State класу.

Програма повинна включати наступне:

1. Можливість проектування та зберігання аналогових електричних схем;
2. Можливість розрахувати основні характеристики електричного кола для кожної зі схем;
3. Завдання для навчання та перевірки знань користувача з фізики (курс «електрика»);
4. Можливість швидкого отримання довідки для допомоги в моделюванні схеми.

# 2.3. Опис мобільного додатка

Розроблена програма містить середовище для моделювання та розрахунку електричних схем та навчальні завдання.

У додатку реалізовано три види взаємодії з проектами:

1. створення нового проекту;
2. завантаження створеного раніше проекту з папки Platform на SD картці пристрою;
3. завантаження навчального завдання з умовою для виконання зі списку доступних завдань;

При створенні нового проекту потрібно вказати його ім’я латиницею. Після цього користувач буде направлений до головного поля, де він може проектувати свою схему. Поле має розмір 2560 на 1282 пікселя, та складається з квадратів 128 на 128 на гранях яких розміщуються елементи. Користувачу доступні наступні елементи та редагування їх параметрів:

Таблиця 2.1

Елементи схеми та їхні параметри

|  |  |
| --- | --- |
| Назва | Параметри,  які вводить користувач |
| З’єднання | - |
| Резистор | Електричний опір |
| Джерело напруги | Електрична напруга |
| Лампа | Потужність, необхідна напруга |

Для кожного елемента реалізована можливість отримання швидкої довідки на інтернет-ресурсі «Wikipedia». Кожен елемент має обов’язкові для вводу параметри (табл. 2.1). Якщо параметри не введені, стає неможливим зробити розрахунок схеми. Користувач має можливість налаштовувати елемент за власним бажанням. Два елементи вважаються безпосередньо з’єднаними якщо існує з’єднання саме між контактами цих елементів. З’єднання належить полю, тому після переміщення елементу, його потрібно буде перемалювати.

Після того, як всі елементи будуть налаштовані, з’явиться кнопка запуску обчислення схеми. Результатом буде новий екран, з розрахованими параметрами на кожній ділянці кола та анімацією руху електронів.

Все коло розглядається як зважений орієнтований граф. Для розрахунку був написаний спеціальний алгоритм, який складається з трьох частин:

1. визначення напрямку руху струму на кожному елементі або з’єднанні;
2. визначення загального опору та опору компонентів паралельності;
3. заповнення параметрів кожного елементу.

Формули, які використовуються для обчислень:

Закон Ома для ділянки кола

Загальна напруга при послідовному з’єднанні

Загальна напруга при паралельному з’єднанні

Загальна сила струму при послідовному з’єднанні

Загальна сила струму при паралельному з’єднанні

Загальний опір при паралельному з’єднанні

Загальний опір при послідовному з’єднанні

Напруга кожної ділянки кола, якщо відомі опори всіх ділянок та загальна напруга

Розділ «Навчання» містить список завдань статус яких відмічений кольором (зелений – виконано, синій – в іншому випадку). Після вибору завдання буде показано його умову. Виконанням завдання є моделювання схеми яка б відповідала заданим умовам, та в якій будуть використовуватися задані елементи (якщо така умова є).

# РОЗДІЛ 3. АПРОБАЦІЯ МОБІЛЬНОГО ДОДАТОКУ

## 3.1. Результат апробації роботи і перспективи розвитку

Результатом виконання даної роботи є навчальна програма на платформі Java.

Даний продукт успішно пройшов тестування школярами і вчителями КЗО «Дніпропетровський обласний ліцей-інтернат фізико-математичного профілю». За результатами тестування додаток був допрацьований, були враховані всі знайдені недоліки у роботі алгоритму, дизайну та функціоналу.

Сподіваємось, що програма допоможе учням у вивченні шкільного курсу фізики «електрика», закріпить їхні знання з нього та полегшить вчителям процес навчання дітей.

Ідеї щодо подальшого розвитку додатка:

* розробити базу даних у мережі Інтернат, де буде зафіксовано досягнення користувачів;
* розробити можливість оновлювати завдання, завантажуючи їх з бази даних у мережі Інтернет;
* розробити можливість моделювання та обчислення цифрових електронних схем;
* поширити додаток на керування системою «Розумний дім».

## 3.2. Вимоги до мобільних пристроїв, інструкція до встановлення

Пристрій повинен працювати на платформі Android. Оптимальний роздільна здатність дисплею – 1280\*720. Використання відбувається в портретному режимі. Для встановлення додатку потрібно завантажити інсталяційний файл на пристрій (або на карту пам’яті). Його треба відкрити за допомогою файлового менеджеру та діяти за інструкціям по встановленню на екрані пристрою. Після встановлення його можна відкрити із головного меню встановлених програм.

## 3.3. Інструкція для користувачів

Після запуску додатку, на головному екрані, користувачу доступно три дії – створення нового проекту, завантаження вже створеного, завантаження завдання (Рис. 3.1.). За кожну з цих функцій відповідають кнопки з відповідними записами.

3.3.1. Створення нового проекту. Після кліку користувачем на кнопку «New» з’явиться меню вводу імені нового файлу. Ім’я необхідно вводити використовуючи букви латинського алфавіту та цифри. Після закінчення введення, потрібно підтвердити дії, натиснувши кнопку «Ок». (Рис. 3.2.)

Якщо ім’я введено невірно, або такий файл вже існує з’явиться повідомлення про помилку.

Якщо створення пройшло успішно користувач буде направлений на екран поля проекту.

3.3.2. Завантаження вже створеного проекту. Після кліку користувачем на кнопку «Open» з’явиться меню з рухомим списком створених користувачем файлів. Кожен пункт списку містить назву проекту, кнопку вибору проекту, кнопку видалення проекту. Після видалення, повернути проект буде неможливо. Після вибору проекту користувач буде направлений на екран поля проекту. (Рис. 3.3.)

3.3.3. Завантаження завдань. Після натискання на кнопку «Learn» з’явится рухомий список навчальних проектів. Кожен пункт списку мстить кнопку запуску та назву завдання. Після вибору завдання, користувачу відображатиметься список умов для його виповнення. (Наприклад отримати певне значення сили струму, використовуючи елементи з заданими параметрами).

3.3.4. Приклад виконання завдання. Кожне завдання має умову, та може мати список елементів які можна використовувати. Завдання вважається виконаним, тільки коли виконані всі його умови. Елементи, які прописані в умові, не можливо не використати або видалити. Якщо є чітко встановлені параметри елементу, то їх неможливо змінювати. Приклад умови завдання зображений на Рис. 3.4. В цьому завданні немає обмежень на використання елементів, або на їх параметри. Найпростіша послідовність виконання завдання:

1. відкриваємо меню «Додати» та вибираємо резистор;
2. на панелі елементу вибираємо кнопку редагування параметрів;
3. вказуємо значення опору = 9, натискаємо «Ок», натискаємо кнопку підтвердити;
4. в меню «Додати» вибираємо джерело напруги і аналогічно до резистору, вказуємо значення напруги джерела = 27;
5. у меню «Додати» вибираємо з’єднання;
6. з’єднуємо два контакти – джерела і резистора (натискаємо на один контакт, і не відриваючи палець, проводимо з’єднання до іншого);
7. аналогічно з’єднуємо контакти, що залишилися;
8. запускаємо схему;
9. якщо все зроблено вірно, то запуститься новий екран з розрахунком схеми, анімацією руху електронів та повідомленням про успішне виконання завдання. (Рис. 3.5.)

Якщо завдання було виконано успішно, то у списку завдань воно буде виділене зеленим кольором.

##### Screenshot_2018-01-21-16-48-10-801 Screenshot_2018-01-21-16-48-14-383 Screenshot_2018-01-21-16-48-18-316

Рис. 3.1.

Стартовий екран

Рис. 3.2.

Введення імені

Рис. 3.3.

Вибір проекту

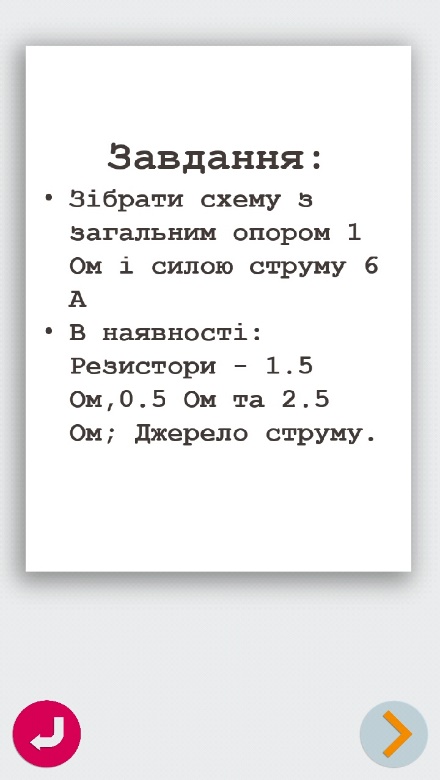
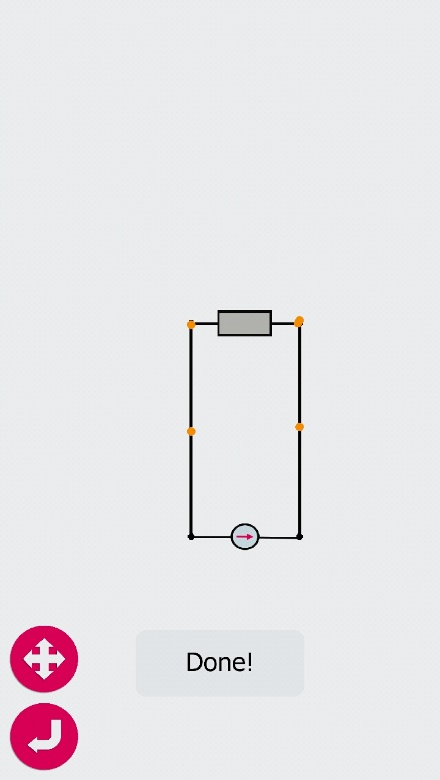
 

Рис. 3.4.

Умова завдання

Рис. 3.5.

Виконане завдання

3.3.5. Поле проекту. Головне поле проекту має розмір 2560 на 1282 пікселі і складається з квадратів 128 на 128, на сторонах яких розміщуються елементи. На головному полі знаходяться кнопки «Назад», «Переміщувати», «Додати».

При активації кнопки переміщення, користувач має змогу рухати поле при цьому не взаємодіючі з іншими кнопками або елементами (для того щоб уникнути випадкових кліків).

Кнопка «Назад» повертає користувача на попередній екран, при цьому зберігаючи зміни у проекті.

Кнопка «Додати» відкриває екран вибору елементів.

3.3.6. Екран «Додати». Екран додати містить кнопки «Назад», «Інформація» та рухому стрічку з елементами (З’єднання, Резистор, Лампа, Джерело напруги). (Рис. 3.6. і 3.7.)

На екрані відображається назва поточного елементу та його доступна кількість (у нового проекту це 256 ламп, 256 резисторів, 1 джерело напруги; у режимі навчання кількість елементів може бути інша).

При натисканні кнопки «Інформація» буде відкрита сторінка з «Wikipedia» з інформацією про обраний елемент. (Рис. 3.8.)

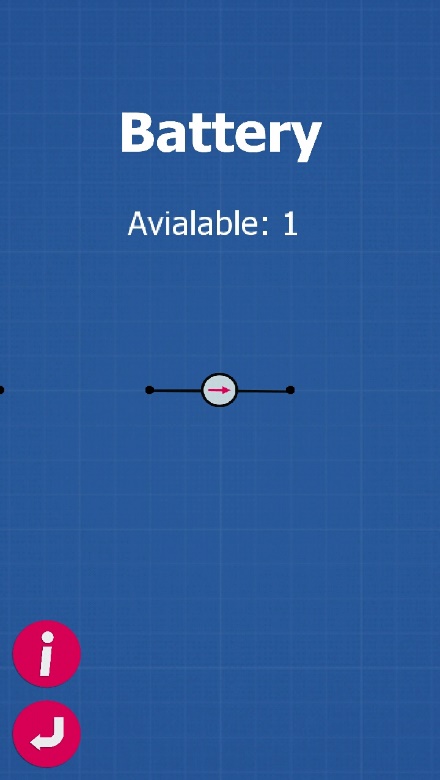
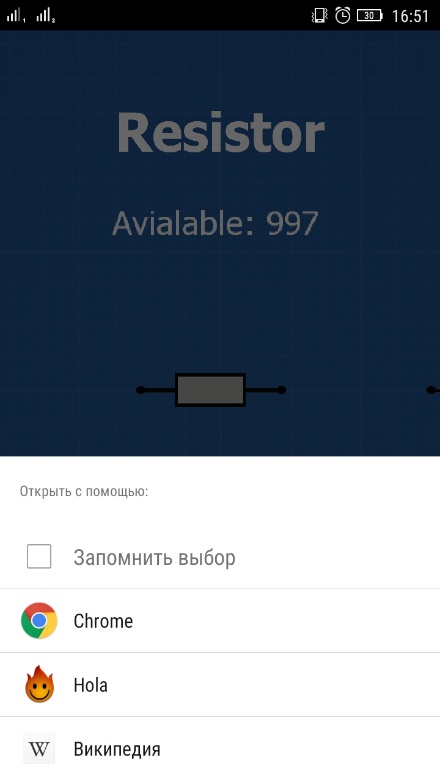
  

Рис. 3.7.

Екран «Додати»

Рис. 3.8.

Стартовий екран

Рис. 3.6.

Екран «Додати»

3.3.7. Елементи та розрахунок. На полі проекту, в кожного елементу є панель з кнопок («Видалити», «Повернути», «Підтвердити», «Настроїти»), яка активується при натисканні на елемент. (Рис. 3.10.)

Кожен елемент має параметри які необхідно ввести для запуску розрахунку схеми (табл. 2.1.). Якщо параметри не введені, на полі проекту, поряд з елементом, буде відображатись знак застереження (Рис. 3.9.).

Після того, як параметри будуть встановлені, з’явиться кнопка розрахунку. Її натискання відкриє екран з розрахованим колом і анімацією руху електронів. При натисканні на елемент буде відображена таблиця з його параметрами, яку користувач може рухати по екрану. При повторному натисканні – таблиця буде схована (Рис. 3.11).

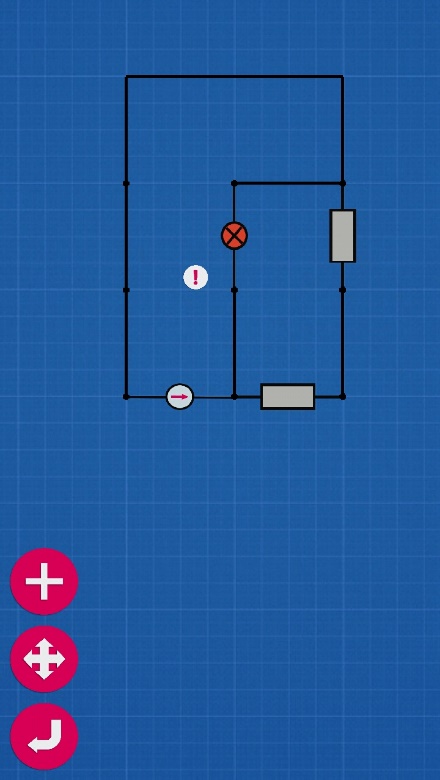
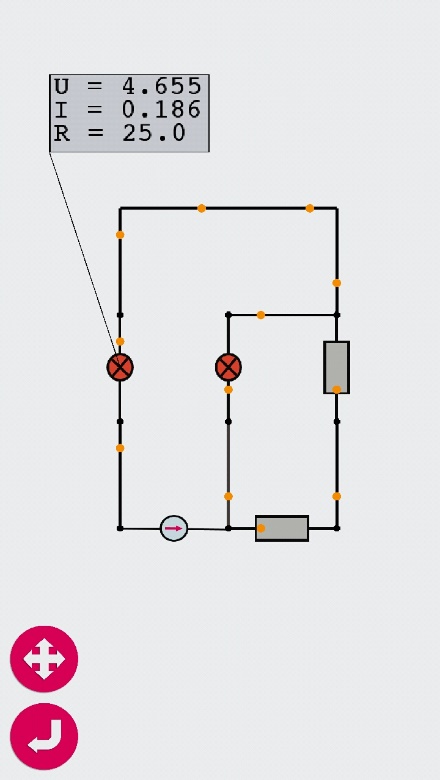
  

Рис. 3.9. Знак застереження

Рис. 3.10. Панель елементу

Рис. 3.11. Екран з розрахунком

# ВИСНОВКИ

1. Було проведено аналіз існуючих додатків, їх можливостей, призначення, функціоналу тощо.
2. Виявлені головні недоліки існуючих програм, а саме: відсутність локалізації, обмеженість платформи та відсутність навчальних завдань.
3. Обрано мову програмування Java, середовище розробки Android Studio, та фреймворк libGDX. Це вибір було обумовлено можливостями мови Java, популярністю мобільних пристроїв на платформі Android, зручність використання фреймворків для відтворення нетипового функціоналу програми,
4. Розроблена структура програми з наступним функціоналом: можливість моделювання та розрахунку аналогових електричних схем, можливість виконувати завдання для навчання та поліпшення знань з фізики (курс «електрика»), можливість швидко переглянути довідку на популярному інтернет-ресурсі «Wikipedia».
5. Розроблено простий у використанні мобільний навчальний додаток, який можна за необхідністю перенести на більшість популярних платформ, створено можливість використання інтернет-довідки.
6. Даний продукт успішно пройшов апробацію серед школярів 8, 9 та 10-х класів та їхніх вчителів. За результатами тестування додаток був допрацьований, враховуючі всі знайдені недоліки у роботі алгоритму, дизайну та функціоналу.
7. Результати апробації дозволили виявити низку ідей для подальшого розвитку програми.

# СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Вікіпедія. [електронний ресурс]. – Режим доступу: <https://uk.wikipedia.org/wiki/Java>
2. Офіційна документація Android. [електронний ресурс]. – Режим доступу: <https://developer.android.com>
3. Офіційна документація libGDX [електронний ресурс]. – Режим доступу: <http://www.libgdx.ru/p/guide.html>
4. Педагогічний програмний засіб Математика для комп’ютера 1-5 клас [електронний ресурс]. Режим доступу: https://plus1s.com/uk/matematika-dlya-kompyutera/
5. Топ мобільних додатків для школярів. Режим доступу: http://lady.tochka.net/ua/59398-top-mobilnykh-prilozheniy-dlya-shkolnikov/
6. EveryCircuit [електронний ресурс]. Режим доступу: https://play.google.com/store/apps/details?id=com.everycircuit.free&hl=uk
7. Circuit Solver [електронний ресурс]. Режим доступу: <https://play.google.com/store/apps/details?id=summersedge23.com.circuitbreaker.lite&hl=ru>.
8. Proteus Design Suite [електронний ресурс]. Режим доступу: https://en.wikipedia.org/wiki/Proteus\_Design\_Suite.