Міністерство освіти і науки України  
Національний технічний університет України

«Київський політехнічний інститут ім. І. Сікорського»

Кафедра інженерії програмного забезпечення в енергетиці

Лабораторна робота № 6

з курсу: «*Кросплатформна розробка мобільних застосунків*»

**Виконав:**  
студент 4-го курсу,  
групи ТВ-11  
Аспарян Дмитро Сергійович

Посилання на GitHub репозиторій: <https://github.com/SupCS/PW6TB-11_AsparianDmytroSerhiyovychFlutter>

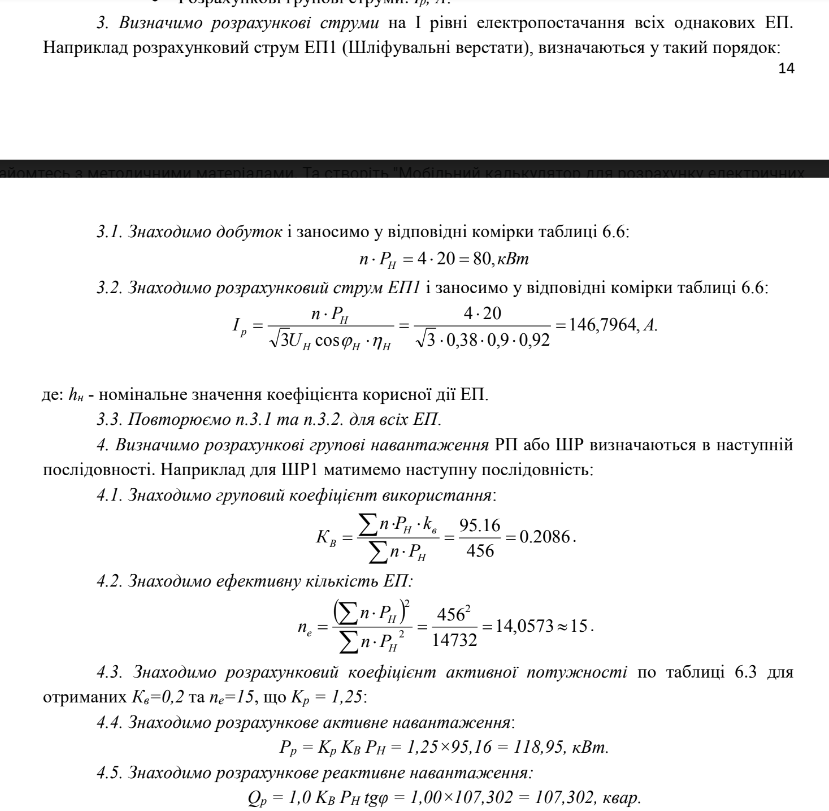
**Перевірив:**

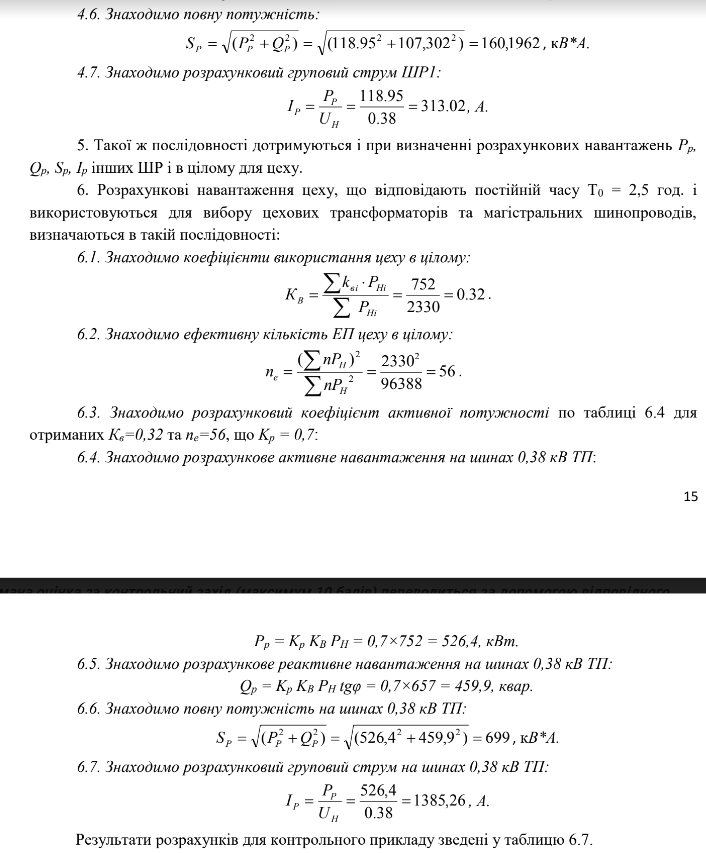
Недашківський О.Л.

Київ 2024/2025

Лабораторна робота № 6

**Короткий теоретичний матеріл:**

****

****

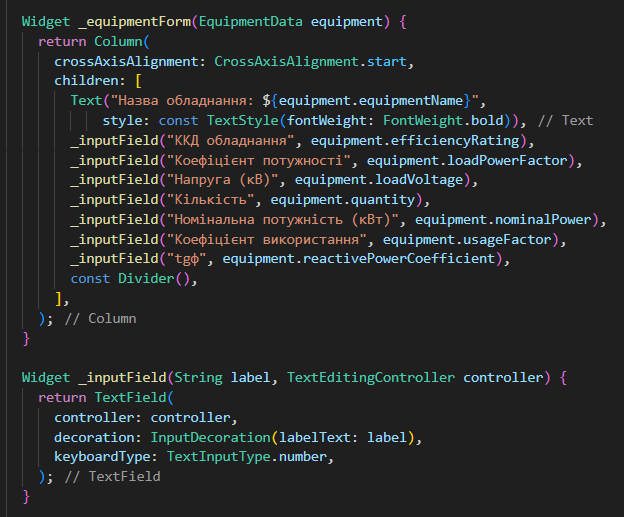
**Завдання:**

Створіть кросплатформений мобільний калькулятор для розрахунку електричних навантажень об’єктів з використанням методу впорядкованих діаграм. Цехова мережа складається з трьох типових цехів які під’єднується до трьох різних розподільчих шин (ШР1-ШР3) та кількох крупних електроприймачів (ЕП). Для спрощення приймемо що склад, номенклатура і характеристики ЕП всіх трьох цехів однакові.

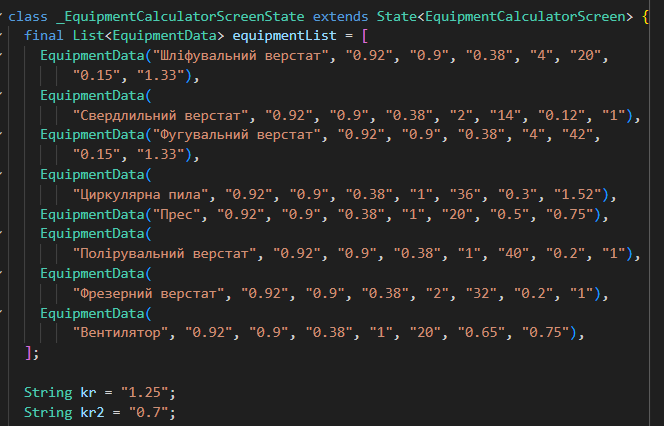
На основі складу ЕП та їх характеристик необхідно розрахувати силове навантаження цехової мережі.

**Хід виконання:**

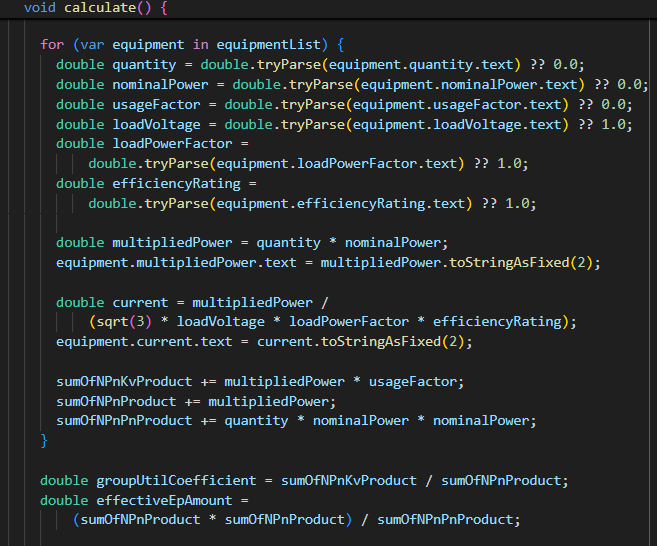
Реалізував форму для введення параметрів окремого електроприводу та місце для виведення результатів.

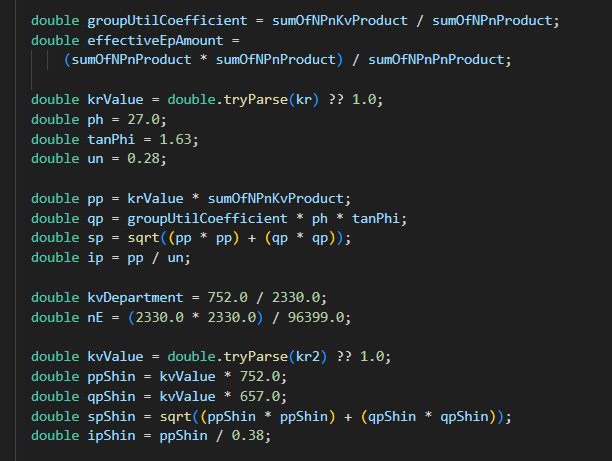


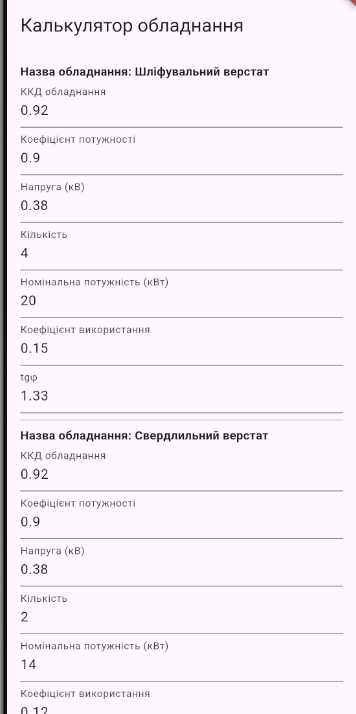
Cтворено жорстко заданий список equipmentList, що містить дані про кілька типів обладнання.

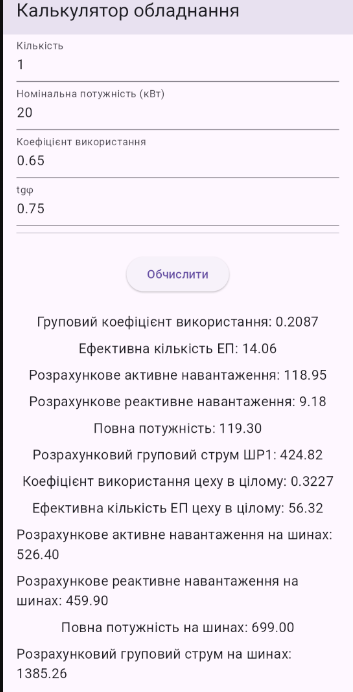


Реалізовано алгоритм для обчислення кількісних характеристик. Розрахунки проводяться для кожного пристрою окремо, а також для групового підсумування.









**Висновок**

В рамках лабораторної роботи було створено кросплатформений застосунок для розрахунку характеристик електроприводів, що дозволяє вводити параметри обладнання та отримувати результати розрахунків, зокрема ефективну кількість електроприводів, активне та реактивне навантаження, а також струмові характеристики.

Одна відмінність між реалізаціями на Котлін і Флаттер полягає в підході до побудови інтерфейсу та управління станом. У Jetpack Compose стан зберігався безпосередньо в змінних за допомогою remember та mutableStateOf, тоді як у Flutter використано StatefulWidget і setState для оновлення інтерфейсу після кожного розрахунку. В Jetpack Compose всі компоненти UI створюються через @Composable функції, що дозволяє декларативно описати логіку та взаємодію між елементами, у Flutter натомість використовується класова модель Widget, що вимагає явного визначення стану кожного компонента.

Також перенос програми на Flutter вирішує проблему кросплатформеності, адже тепер додаток доступний і на Android, і на iOS без дублювання коду.