Міністерство освіти і науки України  
Національний технічний університет України

«Київський політехнічний інститут  ім. І. Сікорського»

Кафедра інженерії програмного забезпечення в енергетиці

Лабораторна робота № 3

з курсу: «Розробка програмного забезпечення мобільних пристроїв»

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

Гущін Володимир Сергійович

Посилання на GitHub репозиторій:

https://github.com/HushchinVolodymyr/MobileDevLr\_3

**Перевірив:**

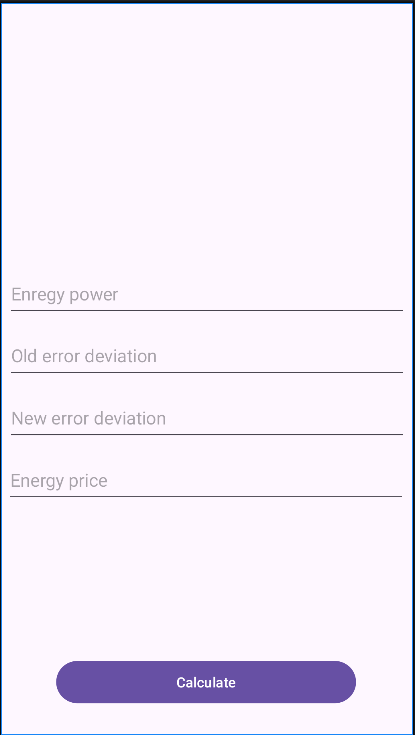
Недашківський О.Л.

Київ 2024/2025

Завдання:

Створіть мобільний калькулятор розрахунку прибутку від сонячних електростанцій з встановленою системою прогнозування сонячної потужності

Сторінка калькулятору:



Є поля для вводу:

* Потужність генерування нерегії
* Стандартне відхилення
* Зменшене відхилення
* Ціна енергії (кВт/год)

Для вирішення завдання створено клас калькулятора:

*package* com.example.lr\_3  
  
*import* kotlin.math.PI  
*import* kotlin.math.exp  
*import* kotlin.math.sqrt  
  
*class* SolarCalculator {  
 *fun* calculateProfit(  
 averagePower: Double,  
 probability: Double,  
 energyPrice: Double,  
 ): Double {  
 *return* (averagePower \* *24* \* probability) \* energyPrice  
 }  
  
 *private fun* normalDistribution(x: Double, mean: Double = *5.0*, stdDev: Double = *1.0*): Double {  
 *return* (*1* / (stdDev \* *sqrt*(*2* \* PI))) \* *exp*(-((x - mean) \* (x - mean)) / (*2* \* stdDev \* stdDev))  
 }  
  
 *fun* calculateProbability(lowerBound: Double, upperBound: Double, step: Double = *0.001*): Double {  
 *var* integral = *0.0  
 var* x = lowerBound  
  
 *while* (x < upperBound) {  
 *val* y1 = *normalDistribution*(x)  
 *val* y2 = *normalDistribution*(x + step)  
 integral += (y1 + y2) \* step / *2* x += step  
 }  
  
 *return* integral \* *100* }  
}

Також отримуються данні з головної сторінки з полів вода відправляються до калькулятора та обчислюються після чого виводиться результат на екран.