Міністерство освіти і науки України  
Національний технічний університет України

«Київський політехнічний інститут ім. І. Сікорського»

Кафедра інженерії програмного забезпечення в енергетиці

Лабораторна робота № 2

з курсу: «*Кросплатформна розробка мобільних застосунків*»

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

Посилання на GitHub репозиторій:

<https://github.com/Chivas1717/cross-platform-labs/blob/master/app/lib/calculators/lab2.dart>

**Перевірив:**

Недашківський О.Л.

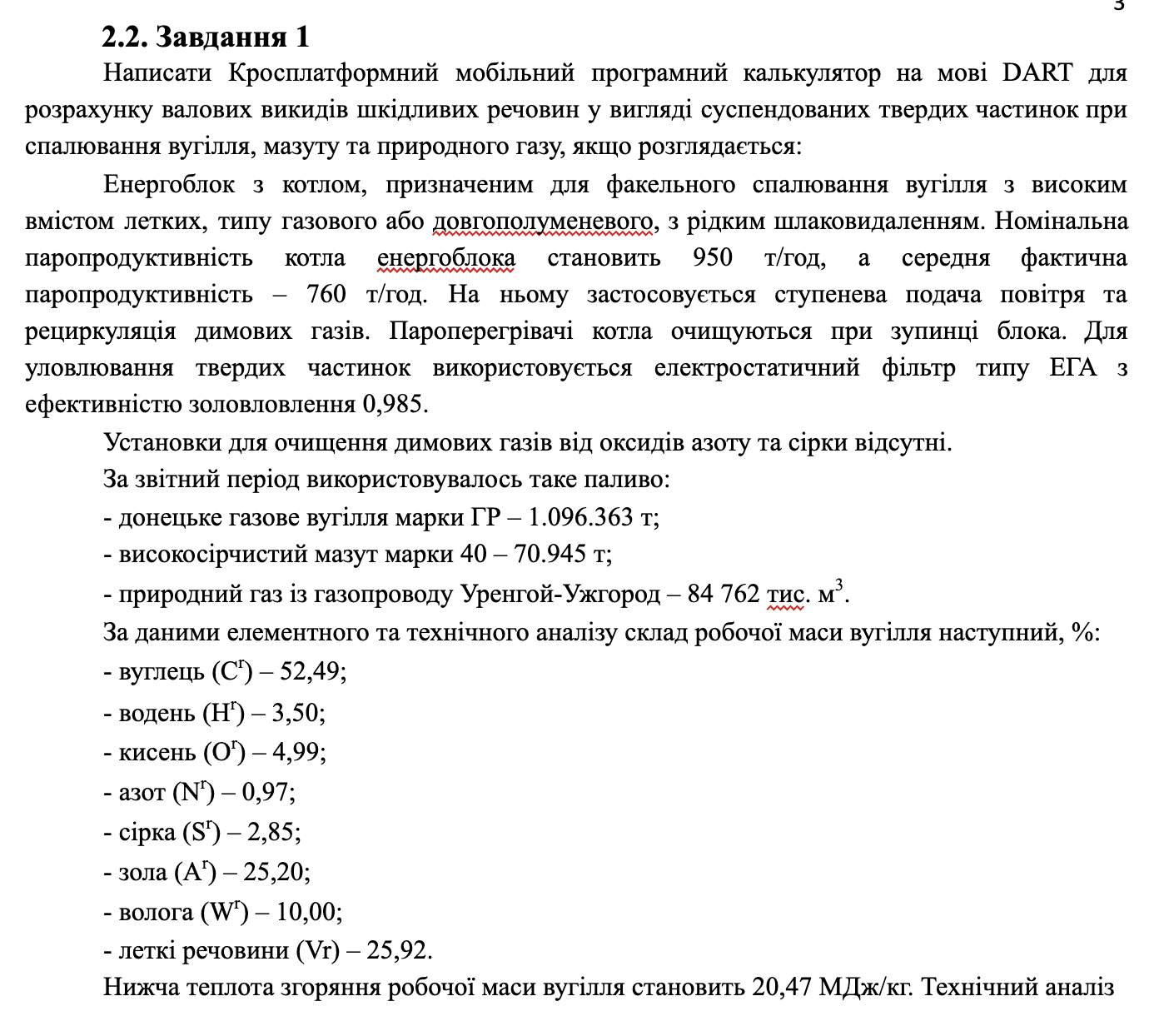
Київ 2025

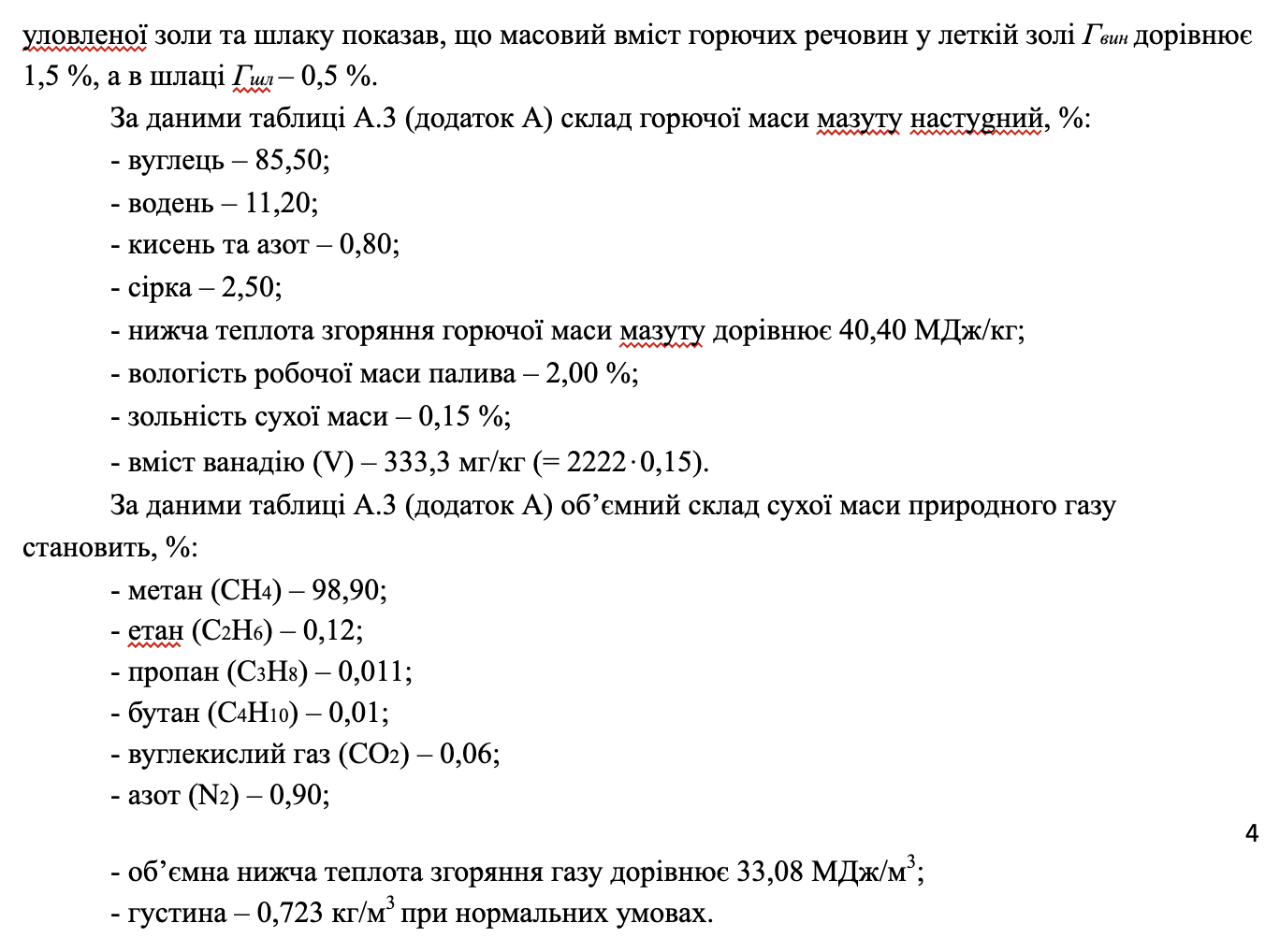
Лабораторна робота № 2

Варіант №4

**Хід виконання:**

Метою роботи є створення кросплатформеного мобільного програмного калькулятора на мові DART для розрахунку валових викидів шкідливих речовин у вигляді суспендованих твердих частинок при спалювання вугілля, мазуту та природного газу.





Програмний код:

import 'package:flutter/material.dart';

class Lab2Task1Calculator extends StatefulWidget {

@override

\_Lab2Task1CalculatorState createState() => \_Lab2Task1CalculatorState();

}

class \_Lab2Task1CalculatorState extends State<Lab2Task1Calculator> {

final coalUsedCtrl = TextEditingController(text: '672419.96');

final coalQCtrl = TextEditingController(text: '20.47');

final coalEFCtrl = TextEditingController(text: '150');

final mazutUsedCtrl = TextEditingController(text: '111633.33');

final mazutQCtrl = TextEditingController(text: '40.40');

final mazutEFCtrl = TextEditingController(text: '0.57');

final gasUsedCtrl = TextEditingController(text: '128674.68');

final gasQCtrl = TextEditingController(text: '33.08');

final gasEFCtrl = TextEditingController(text: '0.01');

double? coalEF, coalGross;

double? mazutEF, mazutGross;

double? gasEF, gasGross;

void calculate() {

setState(() {

double cUsed = double.*tryParse*(coalUsedCtrl.text) ?? 0;

double cQ = double.*tryParse*(coalQCtrl.text) ?? 0;

double cEF = double.*tryParse*(coalEFCtrl.text) ?? 0;

double cEnergy = cUsed \* cQ;

coalEF = cEF;

coalGross = (cEF \* cEnergy) / 1e6;

double mUsed = double.*tryParse*(mazutUsedCtrl.text) ?? 0;

double mQ = double.*tryParse*(mazutQCtrl.text) ?? 0;

double mEF = double.*tryParse*(mazutEFCtrl.text) ?? 0;

double mEnergy = mUsed \* mQ;

mazutEF = mEF;

mazutGross = (mEF \* mEnergy) / 1e6;

double gUsed = double.*tryParse*(gasUsedCtrl.text) ?? 0;

double gQ = double.*tryParse*(gasQCtrl.text) ?? 0;

double gEF = double.*tryParse*(gasEFCtrl.text) ?? 0;

double gEnergy = gUsed \* gQ;

gasEF = gEF;

gasGross = (gEF \* gEnergy) / 1e6;

});

}

@override

Widget build(BuildContext context) {

return SingleChildScrollView(

padding: const EdgeInsets.all(16),

child: Column(

crossAxisAlignment: CrossAxisAlignment.start,

children: [

const Text('Вугілля', style: TextStyle(fontWeight: FontWeight.*bold*)),

const SizedBox(height: 8),

buildField(coalUsedCtrl, 'Маса вугілля, т'),

const SizedBox(height: 8),

buildField(coalQCtrl, 'Нижча теплота згоряння, ГДж/т'),

const SizedBox(height: 8),

buildField(coalEFCtrl, 'Показник емісії, г/ГДж'),

const SizedBox(height: 16),

const Text('Мазут', style: TextStyle(fontWeight: FontWeight.*bold*)),

const SizedBox(height: 8),

buildField(mazutUsedCtrl, 'Маса мазуту, т'),

const SizedBox(height: 8),

buildField(mazutQCtrl, 'Нижча теплота згоряння, ГДж/т'),

const SizedBox(height: 8),

buildField(mazutEFCtrl, 'Показник емісії, г/ГДж'),

const SizedBox(height: 16),

const Text('Природний газ',

style: TextStyle(fontWeight: FontWeight.*bold*)),

const SizedBox(height: 8),

buildField(gasUsedCtrl, 'Витрата газу, тис.м³'),

const SizedBox(height: 8),

buildField(gasQCtrl, 'Нижча теплота згоряння, ГДж/тис.м³'),

const SizedBox(height: 8),

buildField(gasEFCtrl, 'Показник емісії, г/ГДж'),

const SizedBox(height: 16),

ElevatedButton(onPressed: calculate, child: const Text('Calculate')),

const SizedBox(height: 24),

if (coalEF != null) ...[

const Text('2.2.2. Результат',

style: TextStyle(fontWeight: FontWeight.*bold*)),

const SizedBox(height: 8),

const Text('1. Для заданого енергоблоку:'),

const SizedBox(height: 8),

Text('1.1. Показник емісії твердих частинок (вугілля): '

'${coalEF!.toStringAsFixed(2)} г/ГДж'),

Text('1.2. Валовий викид (вугілля): '

'${coalGross!.toStringAsFixed(2)} т.'),

const SizedBox(height: 8),

Text('1.3. Показник емісії твердих частинок (мазут): '

'${mazutEF!.toStringAsFixed(2)} г/ГДж'),

Text('1.4. Валовий викид (мазут): '

'${mazutGross!.toStringAsFixed(2)} т.'),

const SizedBox(height: 8),

Text('1.5. Показник емісії твердих частинок (газ): '

'${gasEF!.toStringAsFixed(2)} г/ГДж'),

Text('1.6. Валовий викид (газ): '

'${gasGross!.toStringAsFixed(2)} т.'),

const SizedBox(height: 24),

],

],

),

);

}

Widget buildField(TextEditingController c, String label) {

return TextField(

controller: c,

decoration: InputDecoration(

labelText: label,

border: const OutlineInputBorder(),

),

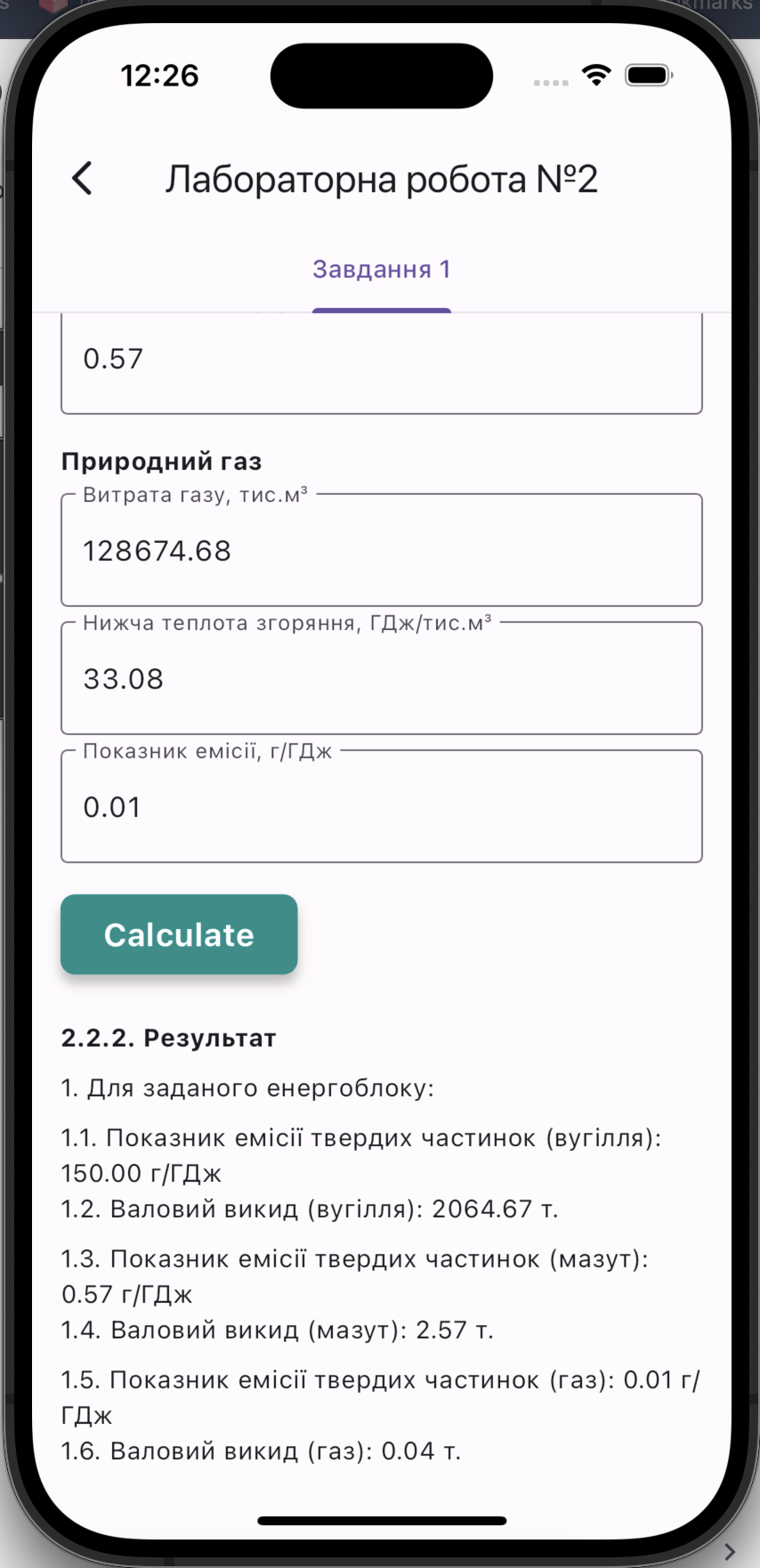
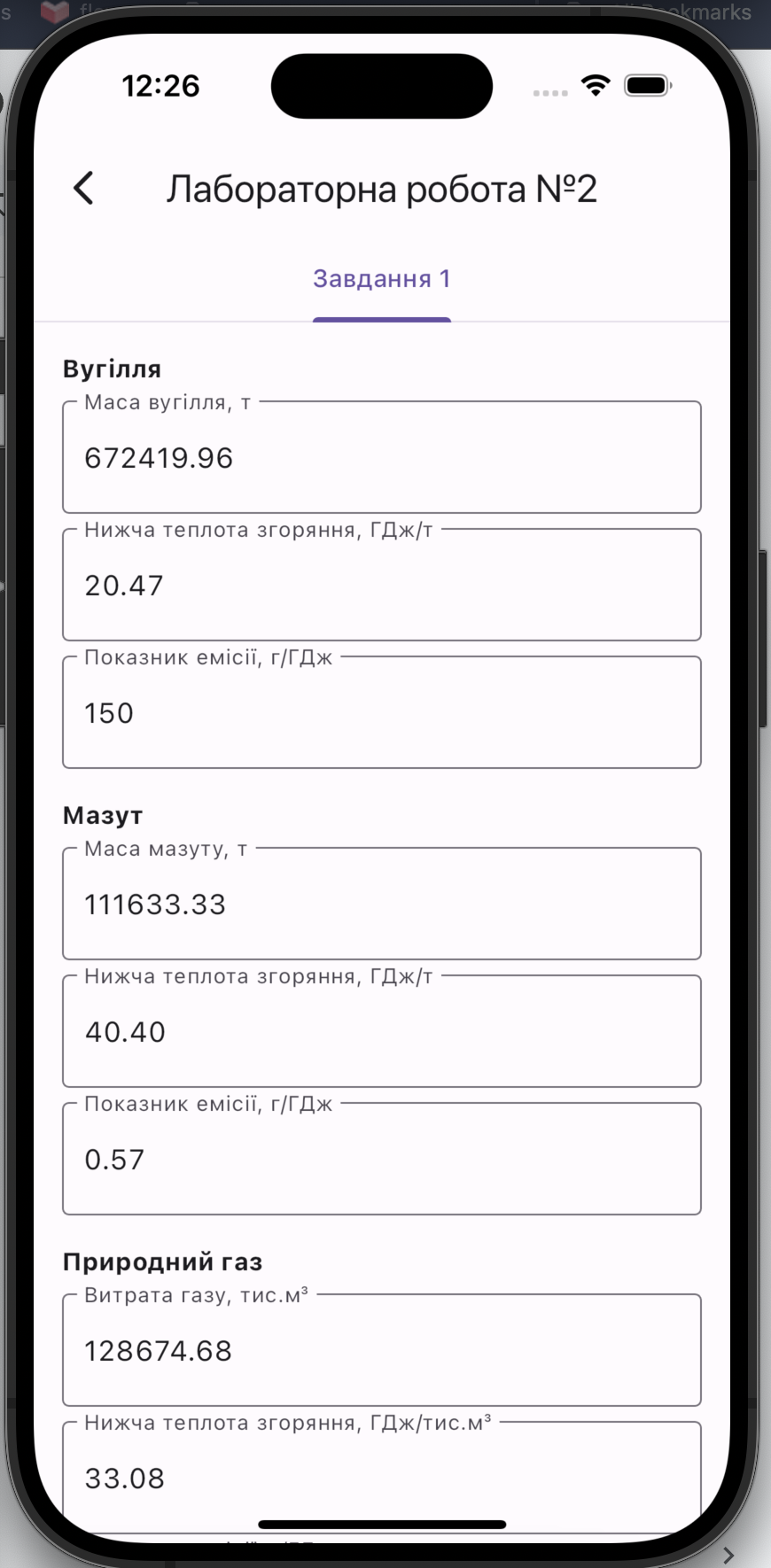
keyboardType: TextInputType.*number*,

);

}

}

Результат перевірки для варіанта №4:



**Висновок**

У цій роботі створено кросплатформний застосунок на Dart/Flutter для розрахунку валових викидів твердих частинок при спалюванні різних видів палива (вугілля, мазут, природний газ).  
Порівняно з підходом на Kotlin, Flutter дає змогу охопити кілька платформ єдиним кодом і швидко розробляти інтерфейси завдяки віджетам. Водночас Kotlin має нативну підтримку Android, що може забезпечувати вищу продуктивність у великих спеціалізованих проєктах.  
Загалом, застосування Flutter у цій роботі дало змогу реалізувати гнучкий та універсальний інструмент, який спрощує подальше розширення функціоналу і може бути використаний як на мобільних, так і на десктопних платформах.