

ระบบบ้านพลังงานแสงอาทิตย์

ไฟล์ main.dart

ส่วนนี้จะแสดงข้อมูลของพลังงาน

```
lib > main.dart > MyApp > build
47 class _MyHomePageState extends State<MyHomePage> {
124   Widget build(BuildContext context) {
172     const SizedBox(height: 10),
173     Row(
174       mainAxisAlignment: MainAxisAlignment.spaceBetween,
175       children: [
176         Text('กำลังการผลิตไฟฟ้า/วัน :'),
177         Text('${powerGeneration.toInt()} kW/day'),
178       ],
179     ), // Row
180     const SizedBox(height: 8),
181     Row(
182       mainAxisAlignment: MainAxisAlignment.spaceBetween,
183       children: [
184         Text('การใช้ไฟฟ้า :'),
185         Text('${powerConsumption.toInt()} kW'),
186       ],
187     ), // Row
188     const SizedBox(height: 8),
189     Row(
190       mainAxisAlignment: MainAxisAlignment.spaceBetween,
191       children: [
192         Text('พลังงานที่เหลือ :'),
193         Text('${remainingEnergy.toInt()} kW'),
194       ],
195     ), // Row
196     const SizedBox(height: 8),
197     Row(
198       mainAxisAlignment: MainAxisAlignment.spaceBetween,
199       children: [
200         Text('จำนวนอุปกรณ์ที่กำลังใช้งานอยู่ :'),
201         Text('${activeDevices.length} เครื่อง'),
202       ],
203     ), // Row
204     const SizedBox(height: 8),
205     Row(
206       mainAxisAlignment: MainAxisAlignment.spaceBetween,
207       children: [
208         Text('จำนวนอุปกรณ์ที่ไม่ได้ใช้งาน :'),
209         Text('${inactiveDevices.length} เครื่อง'),
210       ],
211     ), // Row
212   }
```

databaseที่ใช้ สำหรับเก็บข้อมูล

```
lib > database > deviceDB.dart > DeviceDB > loadAllData
7  class DeviceDB {
20  Future<int> insertDatabase(DeviceItem item) async {
30  };
31  db.close();
32  return keyID;
33  }
34
35  Future<List<DeviceItem>> loadAllData() async {
36  var db = await openDatabase();
37
38  var store = intMapStoreFactory.store('devices');
39
40  var snapshot = await store.find(db,
41  |finder: Finder(sortOrders: [SortOrder('date', false)]));
42
43  List<DeviceItem> devices = [];
44
45  for (var record in snapshot) {
46  DeviceItem item = DeviceItem(
47  |keyID: record.key,
48  |title: record['title'].toString(),
49  |status: record['status'].toString(),
50  |imageUrl: record['imageUrl']?.toString(),
51  |date: DateTime.parse(record['date'].toString()),
52  );
53
54  devices.add(item);
55  }
56  db.close();
57  return devices;
58  }
59
60  deleteData(DeviceItem item) async {
61  var db = await openDatabase();
62  var store = intMapStoreFactory.store('devices');
63  store.delete(db,
64  |finder: Finder(filter: Filter.equals(Field.key, item.keyID)));
65  db.close();
66  }
67
68  updateData(DeviceItem item) async {
```

โค้ดตัวอย่างสำหรับการเพิ่ม ข้อมูลของอุปกรณ์

```
lib > formScreen.dart > ...
1 import 'package:solarhomes/model/deviceItem.dart';
2 import 'package:solarhomes/provider/deviceProvider.dart';
3 import 'package:flutter/material.dart';
4 import 'package:image_picker/image_picker.dart';
5 import 'package:provider/provider.dart';
6 import 'dart:io';
7
8 class FormScreen extends StatefulWidget {
9   const FormScreen({super.key});
10
11   @override
12   State<FormScreen> createState() => _FormScreenState();
13 }
14
15 class _FormScreenState extends State<FormScreen> {
16   final formKey = GlobalKey<FormState>();
17   final titleController = TextEditingController();
18   String selectedStatus = 'กำลังใช้งาน';
19   String? imageUrl; // เพิ่มตัวแปรสำหรับเก็บ URL ของรูปภาพ
20   final ImagePicker _picker = ImagePicker();
21
22   Future<void> _pickImage() async {
23     final XFile? image = await _picker.pickImage(source: ImageSource.gallery);
24     if (image != null) {
25       setState(() {
26         imageUrl = image.path; // เก็บ path ของรูปภาพ
27       });
28     }
29   }
30
31   @override
32   Widget build(BuildContext context) {
33     return Scaffold(
34       appBar: AppBar(
35         backgroundColor: Theme.of(context).colorScheme.inversePrimary,
36         title: const Text('เพิ่มอุปกรณ์'),
37         actions: [
38           IconButton(
39             icon: const Icon(Icons.save),
40             onPressed: () {
41               if (formKey.currentState!.validate()) {
```

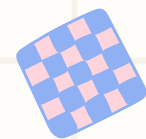
โค้ดในส่วนของการลบข้อมูล

```
lib > editScreen.dart > _EditScreenState > _confirmDelete
17 class _EditScreenState extends State<EditScreen> {
40
41 void _confirmDelete(BuildContext context) {
42   showDialog(
43     context: context,
44     builder: (BuildContext context) {
45       return AlertDialog(
46         title: const Text('ยืนยันการลบ'),
47         content: const Text('คุณต้องการลบรายการนี้ใช่หรือไม่?'),
48         actions: [
49           TextButton(
50             child: const Text('ยกเลิก'),
51             onPressed: () {
52               Navigator.of(context).pop();
53             },
54           ), // TextButton
55           TextButton(
56             child: const Text('ลบ'),
57             onPressed: () {
58               var provider = Provider.of<DeviceProvider>(context, listen: false);
59               provider.deleteDevice(widget.item);
60               Navigator.of(context).pop();
61               Navigator.of(context).pop();
62
63               ScaffoldMessenger.of(context).showSnackBar(
64                 SnackBar(
65                   content: const Text('ลบอุปกรณ์สำเร็จแล้ว'),
66                   duration: const Duration(seconds: 2),
67                   action: SnackBarAction(
68                     label: 'รับทราบ',
69                     onPressed: () {},
70                   ), // SnackBarAction
71                 ), // SnackBar
72               );
73             },
74           ), // TextButton
75         ],
76       ); // AlertDialog
77     },
78   );
79 }
```


โค้ดในส่วนของการแก้ไขข้อมูล

```
17 class _EditScreenState extends State<EditScreen> {
81   @override
82   Widget build(BuildContext context) {
83     return Scaffold(
84       appBar: AppBar(
85         backgroundColor: Theme.of(context).colorScheme.inversePrimary,
86         title: const Text('แก้ไขอุปกรณ์'),
87         actions: [
88           IconButton(
89             icon: const Icon(Icons.delete),
90             onPressed: () {
91               _confirmDelete(context);
92             },
93           ), // IconButton
94         ],
95       ), // AppBar
96       body: Form(
97         key: formKey,
98         child: Padding(
99           padding: const EdgeInsets.all(16.0),
100          child: SingleChildScrollView(
101            child: Column(
102              children: [
103                TextFormField(
104                  decoration: InputDecoration(
105                    label: const Text('ชื่ออุปกรณ์'),
106                    border: OutlineInputBorder(
107                      borderRadius: BorderRadius.circular(8.0),
108                    ), // OutlineInputBorder
109                  ), // InputDecoration
110                  autofocus: true,
111                  controller: titleController,
112                  validator: (String? value) {
113                    if (value!.isEmpty) {
114                      return "กรุณานำชื่ออุปกรณ์";
115                    }
116                    return null;
117                  },
118                ), // TextFormField
119                SizedBox(height: 16),
120                TextFormField(
```

โค้ดในส่วนของการปรับเปลี่ยน สถานะการใช้งานของอุปกรณ์



```
47 class _MyHomePageState extends State<MyHomePage> {  
48   />  
76   void _toggleDeviceStatus(DeviceItem device, DeviceProvider provider) {  
77     showDialog(  
78       context: context,  
79       builder: (BuildContext context) {  
80         return AlertDialog(  
81           title: const Text('ยืนยันการเปลี่ยนสถานะ'),  
82           content: Text(device.status == 'กำลังใช้งาน'  
83             ? 'คุณต้องการปิดใช้งานอุปกรณ์นี้ใช่หรือไม่?'  
84             : 'คุณต้องการเปิดใช้งานอุปกรณ์นี้ใช่หรือไม่?'), // Text  
85           actions: [  
86             TextButton(  
87               child: const Text('ยกเลิก'),  
88               onPressed: () {  
89                 Navigator.of(context).pop();  
90               },  
91             ), // TextButton  
92             TextButton(  
93               child: const Text('ตกลง'),  
94               onPressed: () {  
95                 setState(() {  
96                   device.status = device.status == 'กำลังใช้งาน' ? 'ไม่ได้ใช้งาน' : 'กำลังใช้งาน';  
97                   provider.updateDevice(device);  
98                 });  
99  
100                ScaffoldMessenger.of(context).showSnackBar(  
101                  SnackBar(  
102                    content: Text(device.status == 'กำลังใช้งาน'  
103                      ? 'เปิดใช้งานอุปกรณ์แล้ว'  
104                      : 'ปิดใช้งานอุปกรณ์แล้ว'), // Text  
105                    duration: Duration(seconds: 2),  
106                    action: SnackBarAction(  
107                      label: 'รับทราบ',  
108                      onPressed: () {},  
109                    ), // SnackBarAction  
110                  ), // SnackBar  
111                );  
112  
113                Navigator.of(context).pop();  
114              },
```

ผู้จัดทำ

นายอนุชา เปรมทอง 65315413

จบการนำเสนอ
ขอขอบคุณครับ