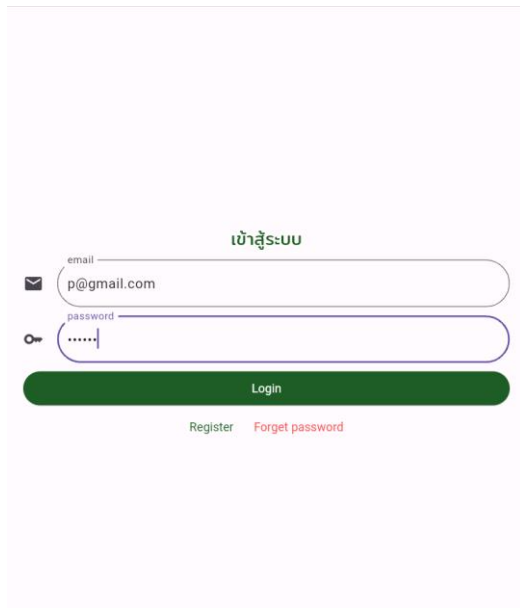


## Miniproject 1 Booking Ticket



เข้าสู่ระบบ

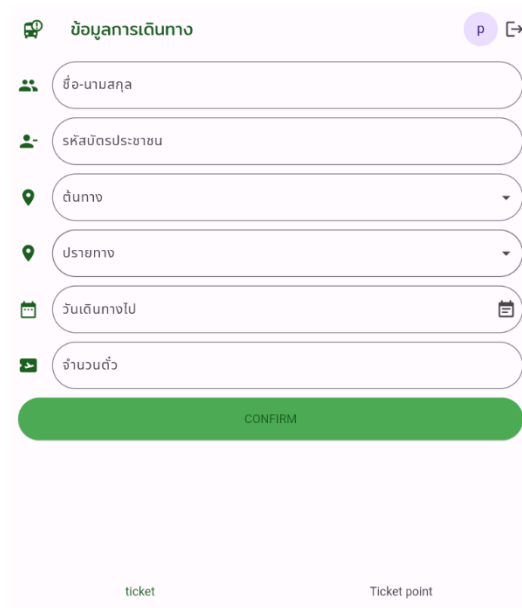
email  
p@gmail.com

password  
.....

Login

Register Forget password

หน้า Login, Register, Forget password



ข้อมูลการเดินทาง

ชื่อ-นามสกุล

รหัสบัตรประชาชน

ต้นทาง

ปลายทาง

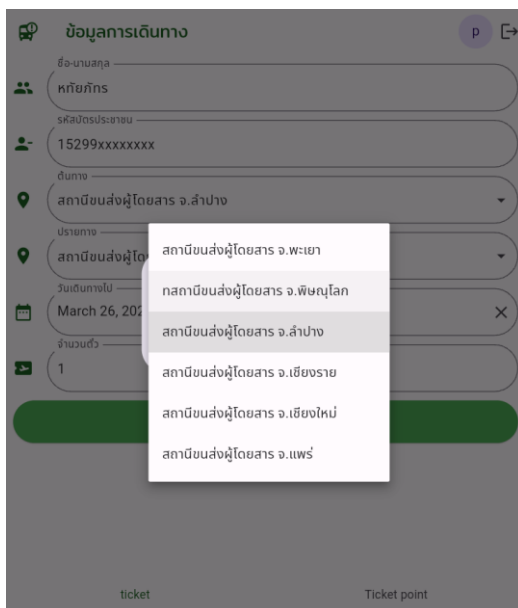
วันเดินทางไป

จำนวนตัว

CONFIRM

ticket Ticket point

หน้ากรอกข้อมูลซื้อตั๋วเดินทาง



ข้อมูลการเดินทาง

ชื่อ-นามสกุล

รหัสบัตรประชาชน

15299xxxxxx

ต้นทาง

สถานีขนส่งผู้โดยสาร จ.ลำปาง

ปลายทาง

สถานีขนส่งผู้โดยสาร จ.พะเยา

วันเดินทางไป

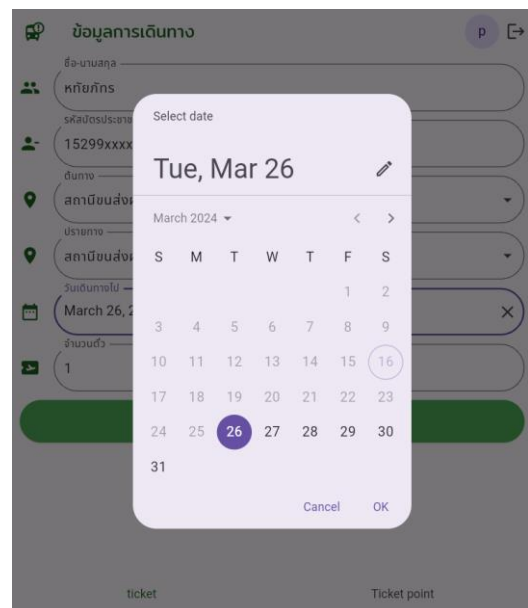
March 26, 2024

จำนวนตัว

1

ticket Ticket point

ตัวอย่างการเลือกข้อมูลแบบ dropdown



ข้อมูลการเดินทาง

ชื่อ-นามสกุล

รหัสบัตรประชาชน

15299xxxxxx

ต้นทาง

สถานีขนส่งผู้โดยสาร จ.ลำปาง

ปลายทาง

สถานีขนส่งผู้โดยสาร จ.พะเยา

วันเดินทางไป

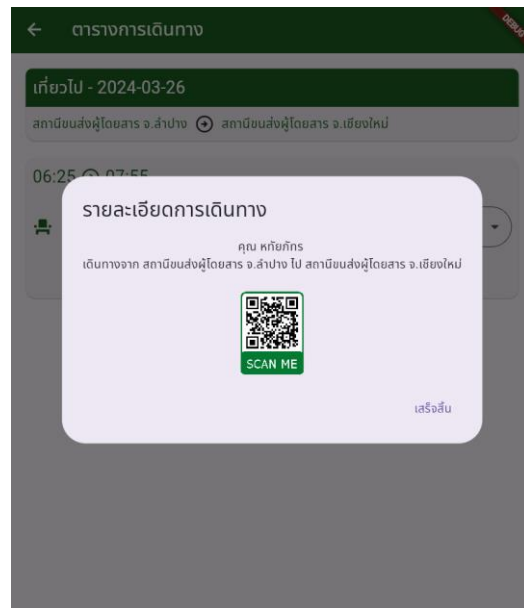
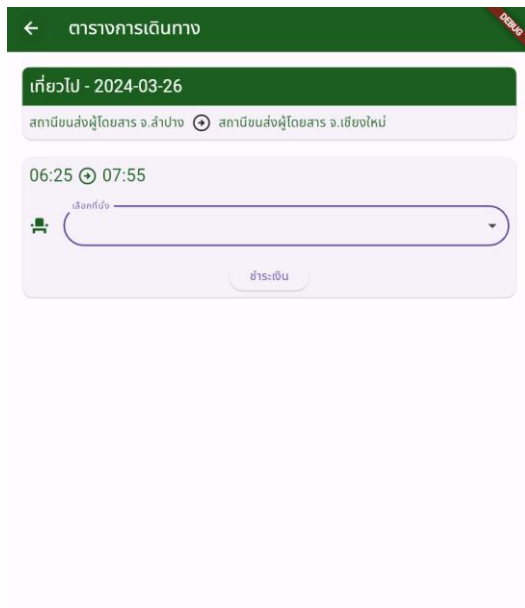
March 26, 2024

จำนวนตัว

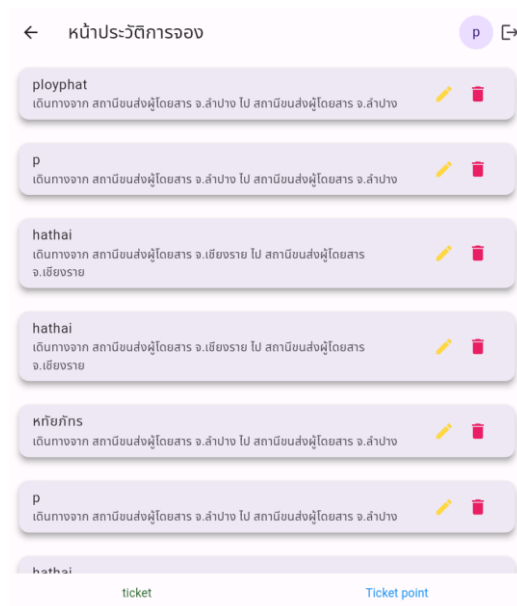
1

ticket Ticket point

ตัวอย่างการเลือกวัน เดือน ปี



หน้าเลือกที่นั่งและชำระเงิน



หน้าประวัติการจองตั๋ว

## หน้า main.dart

```
import 'package:chapter4_cloud_firestore/page/ticket_screen.dart';
import 'package:chapter4_cloud_firestore/page/history_ticket.dart';
import 'package:chapter4_cloud_firestore/page/sign_in_screen.dart';
import 'package:firebase_core/firebase_core.dart';
import 'package:flutter/material.dart';

void main() async {
  WidgetsFlutterBinding.ensureInitialized();
  await Firebase.initializeApp(
    options: FirebaseOptions(
      apiKey: "AlzaSyAKT5f30ZLZMldUErn_uaxGaZVV7QNZ1sl",
      appId: "1:994850625306:web:6fb130818d30618530fea3",
      messagingSenderId: "994850625306",
      projectId: "flutter-firebase-cloud-2fbac"));
  runApp(MaterialApp(
    debugShowCheckedModeBanner: false,
    home: SignInScreen(),
  ));
}

class MyApp extends StatelessWidget {
  const MyApp({super.key});

  @override
  Widget build(BuildContext context) {
    return DefaultTabController(
      length: 2,
      child: Scaffold(
        bottomNavigationBar: TabBar(
          indicatorColor: Colors.blue,
          labelColor: Colors.blue,
          tabs: [
            Tab(
              child: Text("ticket", style: TextStyle(color: Colors.green[900])),
            ),
            Tab(
```

```

        child: Text("Ticket point"),
      ),
    ],
  ),
  body: TabBarView(
    children: [
      TicketScreen(),
      History(),
    ],
  ),
),
);
}
}

```

หน้า sign\_in\_screen.dart

```

import 'package:chapter4_cloud_firestore/main.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:flutter/material.dart';

class SignInScreen extends StatefulWidget {
  const SignInScreen({super.key});

  @override
  State<SignInScreen> createState() => _SignInScreenState();
}

class _SignInScreenState extends State<SignInScreen> {
  TextEditingController _email = TextEditingController();
  TextEditingController _password = TextEditingController();

  final auth = FirebaseAuth.instance;

  bool _isChecked = true;

  @override
  Widget build(BuildContext context) {

```

```
return Scaffold(  
  appBar: AppBar(  
    title: Text(""),  
  ),  
  body: Container(  
    margin: EdgeInsets.all(20),  
    child: Column(  
      mainAxisAlignment: MainAxisAlignment.center,  
      children: [  
        Text(  
          "เข้าสู่ระบบ",  
          style: TextStyle(fontSize:20,fontFamily:'Mitr',color: Colors.green[900]),  
        ),  
        SizedBox(  
          height: 10,  
        ),  
        TextField(  
          controller: _email,  
          decoration: InputDecoration(  
            icon: Icon(Icons.email),  
            labelText: "email",  
            border: OutlineInputBorder(  
              borderRadius: BorderRadius.circular(40))),  
          ),  
        SizedBox(  
          height: 10,  
        ),  
        TextField(  
          obscureText: true,  
          controller: _password,  
          decoration: InputDecoration(  
            icon: Icon(Icons.key),  
            labelText: "password",  
            border: OutlineInputBorder(  
              borderRadius: BorderRadius.circular(40))),  
          ),  
        SizedBox(  
          height: 10,  
        ),  
      ],  
    ),  
  ),  
);
```

```

),
  SizedBox(
    height: 40,
    width: double.infinity,
    child: ElevatedButton(
      onPressed: () async {
        var email = _email.text.toString();
        var pass = _password.text.toString();
        setState(() {
          _isChecked = false;
        });
        try {
          await auth
            .signInWithEmailAndPassword(
              email: email, password: pass
            )
            .then((value) {
              Navigator.pushAndRemoveUntil(context,
                MaterialPageRoute(builder: (BuildContext context) {
                  return MyApp();
                })), (route) => false);
            });
        } on FirebaseAuthException catch (e) {
          print(e.message);
          setState(() {
            _isChecked = true;
          });
        }
      },
      style: ElevatedButton.styleFrom(
        backgroundColor: Colors.green[900], // เปลี่ยนสีพื้นหลังตามที่ต้องการ
      ),
      child: (_isChecked) ? Text("Login", style: TextStyle(color: Colors.white),)
        : SizedBox(
            width: 16,
            height: 16,
            child: CircularProgressIndicator(
              color: Colors.white,
              strokeWidth: 1.5,
            ),
          ),
    ),
  ),

```

```
    ),  
    ),  
),  
  
      SizedBox(  
        height: 10,  
      ),  
  
      Row(  
        mainAxisAlignment: MainAxisAlignment.center,  
        children: [  
          FlatButton(  
            onPressed: () {  
              registerEmail();  
            },  
            child: Text("Register", style: TextStyle(color: Colors.green[900])),  
          ),  
          FlatButton(onPressed: () {}, child: Text("Forget password", style: TextStyle(color:  
Colors.redAccent))),  
        ],  
      )  
    ],  
  ),  
)  
);  
}  
  
registerEmail() {  
  showDialog(  
    context: context,  
    builder: (BuildContext context) {  
      return StatefulBuilder(  
        builder: (context, setState) {  
          return AlertDialog(  
            title: Text("Register Email"),  
            content: Column(  
              mainAxisAlignment: MainAxisAlignment.min,  
              children: [  
                TextField(  
                  controller: email,
```





```

        email: email, password: pass)
        .then((value) {
        Navigator.pop(context);
        ScaffoldMessenger.of(context).showSnackBar(
            SnackBar(
                content: Text("สร้างบัญชีสำเร็จ"),
                backgroundColor: Colors.green,
            ),
        );
    });
} on FirebaseAuthException catch (e) {
    //แสดง Error แบบละเอียดแบบยาว
    print(e.message);
} finally {
    setState(() {
        _isChecked = true;
    });
}
},
child: (_isChecked)
    ? Text("Register")
    : SizedBox(
        width: 16,
        height: 16,
        child: CircularProgressIndicator(
            color: Colors.white,
            strokeWidth: 1.5,
        ),
    ),
),
],
),
);
},
);
},
);

```

```
}  
}
```

หน้า ticket\_screen.dart

```
import 'package:chapter4_cloud_firestore/page/sign_in_screen.dart';  
import 'package:cloud_firestore/cloud_firestore.dart';  
import 'package:date_field/date_field.dart';  
import 'package:firebase_auth/firebase_auth.dart';  
import 'package:flutter/material.dart';  
import 'detail_ticket.dart';
```

```
List<String> list = <String>[  
  'สถานีขนส่งผู้โดยสาร จ.พะเยา',  
  'สถานีขนส่งผู้โดยสาร จ.พิษณุโลก',  
  'สถานีขนส่งผู้โดยสาร จ.ลำปาง',  
  'สถานีขนส่งผู้โดยสาร จ.เชียงราย',  
  'สถานีขนส่งผู้โดยสาร จ.เชียงใหม่',  
  'สถานีขนส่งผู้โดยสาร จ.แพร่',  
];
```

```
List<String> list1 = <String>[  
  'สถานีขนส่งผู้โดยสาร จ.พะเยา',  
  'สถานีขนส่งผู้โดยสาร จ.พิษณุโลก',  
  'สถานีขนส่งผู้โดยสาร จ.ลำปาง',  
  'สถานีขนส่งผู้โดยสาร จ.เชียงราย',  
  'สถานีขนส่งผู้โดยสาร จ.เชียงใหม่',  
  'สถานีขนส่งผู้โดยสาร จ.แพร่',  
];
```

```
class TicketScreen extends StatefulWidget {  
  const TicketScreen({super.key});  
  
  @override  
  State<TicketScreen> createState() => _TicketScreen();  
}
```

```

class _TicketScreen extends State<TicketScreen> {
  TextEditingController _name = TextEditingController(); //ชื่อ
  TextEditingController _id = TextEditingController(); //รหัสบัตร ปชช
  TextEditingController _origin = TextEditingController(); //เมืองต้นทาง
  TextEditingController _destination = TextEditingController(); //เมืองปลายทาง
  TextEditingController _numticket = TextEditingController(); //จำนวนตั๋ว
  var selectedDate; //วันที่เดินทางไป

  String dropdownValue = list.first;
  String dropdownValue1 = list1.first;
  //ส่วนเป็นการสร้าง Object studentCollection เป็นตัวแทนของ student
  CollectionReference busCollection =
    FirebaseFirestore.instance.collection("bus");

  bool _isChecked = true;

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text("ข้อมูลการเดินทาง",
          style: TextStyle(
            fontSize: 20, fontFamily: 'Mitr', color: Colors.green[900])),
        leading: Icon(
          Icons.bus_alert,
          color: Colors.green[900],
        ),
        actions: [
          CircleAvatar(
            // substring หั่นตัวอักษร เพื่อโชว์อักขรตัวแรก โดยตำแหน่ง เริ่มที่ 0 จบที่ 1
            child: Text(FirebaseAuth.instance.currentUser!.email
              .toString()
              .substring(0, 1)),
          ),
          IconButton(
            onPressed: () async {
              FirebaseAuth.instance.signOut().then((value) {

```

```

        Navigator.pushAndRemoveUntil(context,
            MaterialPageRoute(builder: (BuildContext context) {
                return SignInScreen();
            }, (route) => false);
    });
},
    icon: Icon(Icons.logout)),
],
),
body: Container(
    margin: EdgeInsets.all(10),
    child: Column(
        children: [
            TextField(
                controller: _name,
                decoration: InputDecoration(
                    icon: Icon(Icons.people, color: Colors.green[900]),
                    labelText: "ชื่อ-นามสกุล",
                    border:
                        OutlinedInputBorder(borderRadius: BorderRadius.circular(40)),
                ),
            ),
            SizedBox(
                height: 10,
            ),
            TextField(
                controller: _id,
                decoration: InputDecoration(
                    icon: Icon(Icons.person_remove_alt_1_rounded, color: Colors.green[900]),
                    labelText: "รหัสบัตรประชาชน",
                    border:
                        OutlinedInputBorder(borderRadius: BorderRadius.circular(40)),
                ),
            ),
            SizedBox(
                height: 10,
            ),
            TextField(

```

```

controller: _origin,
decoration: InputDecoration(
  icon: Icon(Icons.location_on_sharp, color: Colors.green[900]),
  suffixIcon: Icon(Icons.arrow_drop_down),
  labelText: "ต้นทาง",
  border:
    OutlineInputBorder(borderRadius: BorderRadius.circular(40)),
),
onTap: () {
  _showDropdown(context);
},
readOnly: true,
),
 SizedBox(
  height: 10,
),
 TextField(
  controller: _destination,
  decoration: InputDecoration(
    icon: Icon(Icons.location_on_sharp, color: Colors.green[900]),
    suffixIcon: Icon(Icons.arrow_drop_down),
    labelText: "ปลายทาง",
    border:
      OutlineInputBorder(borderRadius: BorderRadius.circular(40)),
  ),
  onTap: () {
    _showDropdown1(context);
  },
  readOnly: true,
),
  SizedBox(
    height: 10,
  ),
  DateTimeFormField(
    decoration: InputDecoration(
      icon: Icon(Icons.date_range, color: Colors.green[900]),
      labelText: "วันเดินทางไป",
      border: OutlineInputBorder(

```

```

        borderRadius: BorderRadius.circular(40),
      ),
    ),
    firstDate: DateTime.now().add(const Duration(days: 10)),
    lastDate: DateTime.now().add(const Duration(days: 40)),
    mode: DateTimeFieldPickerMode.date,
    onChanged: (DateTime? value) {
      selectedDate = value;
    },
  ),
  SizedBox(
    height: 10,
  ),
  TextField(
    controller: _numticket,
    decoration: InputDecoration(
      icon: Icon(Icons.airplane_ticket_rounded, color: Colors.green[900]),
      labelText: "จำนวนตั๋ว",
      border:
        OutlineInputBorder(borderRadius: BorderRadius.circular(40)),
    ),
  ),
  SizedBox(
    height: 10,
  ),
  SizedBox(
    height: 50,
    width: double.infinity,
    child: ElevatedButton(
      onPressed: (_isChecked)
        ? () async {
            var name = _name.text.toString();
            var id = _origin.text.toString();
            var origin = _origin.text.toString();
            var destination = _destination.text.toString();
            var numticket = _numticket.text.toString();

            if (name.isEmpty ||

```

```

        id.isEmpty ||
        origin.isEmpty ||
        destination.isEmpty ||
        selectedDate == "" ||
        numticket.isEmpty) {
ScaffoldMessenger.of(context)
    .showSnackBar(SnackBar(
        content: Text("กรุณากดกรอกข้อมูลให้ครบ"),
        backgroundColor: Colors.orange,
    ));
} else {
    setState(() {
        _isChecked = false;
    });

    await busCollection.add({
        "name": name,
        "id": id,
        "origin": origin,
        "destination": destination,
        "selecteddate": selectedDate,
        "numticket": numticket
    }).then((value) {
        ScaffoldMessenger.of(context)
            .showSnackBar(SnackBar(
                content: Text("บันทึกข้อมูลสำเร็จ"),
                backgroundColor: Colors.green,
            )); // callbackFunction

        // _name.clear();
        // _id.clear();
        // _origin.clear();
        // _destination.clear(); // clear ใช้สำหรับ Controller
        // selectedDate = " "; // กำหนดให้เป็นค่าว่าง
        // _numticket.clear();

        setState(() {
            _isChecked = true;

```

```

        ));
        Navigator.push(context, MaterialPageRoute(
          builder: (BuildContext context) {
            return DetailTiclet(name, id, origin, destination,
              selectedDate.toString(), numticket);

          }));
      });
    }
  }
  : null,
  child: (_isChecked)
    ? Text(
      "CONFIRM",
      style: TextStyle(color: Colors.green[900]),
    )
    : SizedBox(
      width: 16,
      height: 16,
      child: CircularProgressIndicator(
        color: Colors.white,
        strokeWidth: 1.5,
      ),
    ),
  style: ButtonStyle(
    backgroundColor: MaterialStateColor.resolveWith(
      (states) => Colors.green),
  )))
],
),
),
);
}

void _showDropdown(BuildContext context) {
  showDialog(
    context: context,
    builder: (BuildContext context) {

```



```

return AlertDialog(
  title: Text('ค้นหาทาง'),
  content: DropdownButton<String>(
    value: dropdownValue,
    onChanged: (String? newValue) {
      setState(() {
        dropdownValue = newValue!;
        _origin.text = newValue; // Update text field value
      });
      Navigator.of(context).pop();
    },
    items: list.map<DropdownMenuItem<String>>((String value) {
      return DropdownMenuItem<String>(
        value: value,
        child: Text(value),
      );
    }).toList(),
  ),
);
},
);
}
);
}
}

```

```

void _showDropdown1(BuildContext context) {
  showDialog(
    context: context,
    builder: (BuildContext context) {
      return AlertDialog(
        title: Text('ปลายทาง'),
        content: DropdownButton<String>(
          value: dropdownValue1,
          onChanged: (String? newValue) {
            setState(() {
              dropdownValue1 = newValue!;
              _destination.text = newValue; // Update text field value
            });
            Navigator.of(context).pop();
          },

```

```

        items: list1.map<DropdownMenuItem<String>>((String value) {
          return DropdownMenuItem<String>(
            value: value,
            child: Text(value),
          );
        }).toList(),
      ),
    );
  },
);
}
}

```

หน้า detail\_ticket.dart

```

import 'package:chapter4_cloud_firestore/main.dart';
import 'package:chapter4_cloud_firestore/page/ticket_screen.dart';
import 'package:flutter/material.dart';

List<String> list = <String>[
  'ที่นั่งแถวที่ 1 เบาะที่ 1',
  'ที่นั่งแถวที่ 1 เบาะที่ 2',
  'ที่นั่งแถวที่ 1 เบาะที่ 3',
  'ที่นั่งแถวที่ 2 เบาะที่ 1',
  'ที่นั่งแถวที่ 2 เบาะที่ 2',
  'ที่นั่งแถวที่ 2 เบาะที่ 3',
];

class DetailTicket extends StatefulWidget {
  String name;
  String id;
  String origin;
  String destination;
  String numticket;
  String selectedDate;

  DetailTicket(this.name, this.id, this.origin, this.destination,

```

```
this.selectedDate, this.numticket);
```

```
@override
```

```
_DetailTicletState createState() => _DetailTicletState();
```

```
}
```

```
class _DetailTicletState extends State<DetailTiclet> {
```

```
  TextEditingController _seat = TextEditingController();
```

```
  String dropdownValue = list.first;
```

```
@override
```

```
Widget build(BuildContext context) {
```

```
  return MaterialApp(
```

```
    home: Scaffold(
```

```
      appBar: AppBar(
```

```
        title: Text(
```

```
          "ตารางการเดินทาง",
```

```
          style: TextStyle(color: Colors.white),
```

```
        ),
```

```
        leading: IconButton(
```

```
          icon: Icon(
```

```
            Icons.arrow_back,
```

```
            color: Colors.white,
```

```
          ),
```

```
          onPressed: () {
```

```
            Navigator.pop(context);
```

```
          },
```

```
        ),
```

```
        backgroundColor: Colors.green[900],
```

```
      ),
```

```
      body: ListView(
```

```
        padding: const EdgeInsets.all(8),
```

```
        children: <Widget>[
```

```
          Card(
```

```
            margin: EdgeInsets.all(10),
```

```
            child: Column(
```

```
              children: [
```

```

Container(
  padding: EdgeInsets.all(8),
  decoration: BoxDecoration(
    borderRadius: BorderRadius.only(
      topRight: Radius.circular(8),
      topLeft: Radius.circular(8),
    ),
    color: Colors.green[900],
  ),
  child: Row(
    children: [
      Text(
        "เที่ยวไป - ${widget.selectedDate}.substring(0, 22),
        style: TextStyle(fontSize: 20, color: Colors.white),
      ),
    ],
  ),
),
Container(
  padding: EdgeInsets.all(8),
  child: Row(
    children: [
      Text(
        "${widget.origin} ",
        style: TextStyle(fontSize: 15, color: Colors.green[900]),
      ),
      Icon(Icons.arrow_circle_right_outlined),
      Text(
        " ${widget.destination}",
        style: TextStyle(fontSize: 15, color: Colors.green[900]),
      ),
    ],
  ),
),
],
),
),
Card(

```

```
margin: EdgeInsets.all(10),
child: Column(
  children: [
    Container(
      padding: EdgeInsets.all(8),
      decoration: BoxDecoration(
        borderRadius: BorderRadius.only(
          topRight: Radius.circular(8),
          topLeft: Radius.circular(8),
        ),
      ),
      child: Column(
        children: [
          Row(
            children: [
              Text(
                "06:25 ",
                style: TextStyle(fontSize: 20, color: Colors.green[900]),
              ),
              Icon(Icons.arrow_circle_right_outlined, color: Colors.green[900]),
              Text(
                " 07:55",
                style: TextStyle(fontSize: 20, color: Colors.green[900]),
              ),
            ],
          ),
          SizedBox(height: 20),
          TextField(
            controller: _seat,
            decoration: InputDecoration(
              icon: Icon(Icons.event_seat, color: Colors.green[900]),
              suffixIcon: Icon(Icons.arrow_drop_down),
              labelText: "เลือกที่นั่ง",
              labelStyle: TextStyle(fontSize: 14),
              contentPadding: EdgeInsets.symmetric(vertical: 10.0, horizontal: 10.0),
              border:
                OutlineInputBorder(borderRadius: BorderRadius.circular(40)),
            ),
          ),
        ],
      ),
    ),
  ],
),
```

```

onTap: () {
  _showDropdown(context);
},
readOnly: true,
),
 SizedBox(height: 20),
 ElevatedButton(
  onPressed: () {
    showDialog(
      context: context,
      builder: (BuildContext context) {
        return AlertDialog(
          title: Text('รายละเอียดการเดินทาง'),
          content: Column(
            mainAxisAlignment: MainAxisAlignment.min,
            children: <Widget>[
              Text('คุณ ${widget.name}'),
              Text('เดินทางจาก ${widget.origin} ไป ${widget.destination}'),
              SizedBox(height: 20),
              Container(
                width: 100, // กำหนดความกว้างของรูป
                height: 100, // กำหนดความสูงของรูป
                child: Image.asset('image/qrcode.png'), // เปลี่ยนเส้นทางไปยังรูปภาพตามที่คุณ
              ),
            ],
          ),
        );
      },
    );
    actions: <Widget>[
      TextButton(
        onPressed: () {
          Navigator.push(context, MaterialPageRoute(
            builder: (BuildContext context) {
              return MyApp();
            }));
        },
        child: Text('เสร็จสิ้น'),
      ),
    ],
  ),

```

นำเข้า

```

        );
    },
    );
    },
    child: Text('ชำระเงิน'),

    ),
    ],
    ),
    ),
    ],
    ),
    ),
    ],
    ),
    ),
);
}

void _showDropdown(BuildContext context) {
  showDialog(
    context: context,
    builder: (BuildContext context) {
      return AlertDialog(
        title: Text('ที่นั่ง'),
        content: DropdownButton<String>(
          value: dropdownValue,
          onChanged: (String? newValue) {
            setState(() {
              dropdownValue = newValue!;
              _seat.text = newValue; // Update text field value
            });
            Navigator.of(context).pop();
          },
          items: list.map<DropdownMenuItem<String>>((String value) {
            return DropdownMenuItem<String>(
              value: value,
              child: Text(value),
            );
          });

```

```

        }).toList(),
      ),
    );
  },
);
}
}

```

หน้า history\_ticket.dart

```

import 'package:chapter4_cloud_firestore/page/sign_in_screen.dart';
import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:date_field/date_field.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:flutter/material.dart';

```

```

List<String> list = <String>[
  'สถานีขนส่งผู้โดยสาร จ.พะเยา',
  'สถานีขนส่งผู้โดยสาร จ.พิษณุโลก',
  'สถานีขนส่งผู้โดยสาร จ.ลำปาง',
  'สถานีขนส่งผู้โดยสาร จ.เชียงราย',
  'สถานีขนส่งผู้โดยสาร จ.เชียงใหม่',
  'สถานีขนส่งผู้โดยสาร จ.แพร่',
];

```

```

List<String> list1 = <String>[
  'สถานีขนส่งผู้โดยสาร จ.พะเยา',
  'สถานีขนส่งผู้โดยสาร จ.พิษณุโลก',
  'สถานีขนส่งผู้โดยสาร จ.ลำปาง',
  'สถานีขนส่งผู้โดยสาร จ.เชียงราย',
  'สถานีขนส่งผู้โดยสาร จ.เชียงใหม่',
  'สถานีขนส่งผู้โดยสาร จ.แพร่',
];

```

```

class History extends StatefulWidget {
  const History({super.key});

  @override

```



```

State<History> createState() => _History();
}

class _History extends State<History> {
  CollectionReference busCollection =
    FirebaseFirestore.instance.collection("bus");

  bool _checkState = true;

  TextEditingController _name = TextEditingController(); //ชื่อ
  TextEditingController _id = TextEditingController(); //รหัสบัตร ปชช
  TextEditingController _origin = TextEditingController(); //เมืองต้นทาง
  TextEditingController _destination = TextEditingController(); //เมืองปลายทาง
  TextEditingController _numticket = TextEditingController();
  TextEditingController _selecteddate = TextEditingController();
  var selectedDate;

  String dropdownValue = list.first;
  String dropdownValue1 = list1.first;

  //จำนวนตัว

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text("หน้าประวัติการจอง"),
        actions: [
          CircleAvatar(
            // substring หั่นตัวอักษร เพื่อโชว์อักขระตัวแรก โดยตำแหน่ง เริ่มที่ 0 จบที่ 1
            child: Text(FirebaseAuth.instance.currentUser!.email.toString().substring(0,1)),
          ),
          IconButton(
            onPressed: () async {
              FirebaseAuth.instance.signOut().then((value) {
                Navigator.pushAndRemoveUntil(context,
                  MaterialPageRoute(builder: (BuildContext context) {
                    return SignInScreen();

```

```

        }), (route) => false);
    });
  },
  icon: Icon(Icons.logout)
),
],
),
body: Container(
  child: StreamBuilder(
    stream: busCollection.snapshots(),
    builder: (BuildContext context, AsyncSnapshot<dynamic> snapshot) {
      if (snapshot.connectionState == ConnectionState.waiting) {
        return Center(
          child: CircularProgressIndicator(),
        );
      } else if (snapshot.hasError) {
        return Center(
          child: Text("Error : ${snapshot.hasError.toString()}"),
        );
      } else if (snapshot.data.docs.length == 0) {
        return Center(
          child: Text("ไม่มีข้อมูล..."),
        );
      }
      return ListView.builder(
        itemCount: snapshot.data.docs.length,
        itemBuilder: (BuildContext context, int index) {
          DocumentSnapshot bus = snapshot.data.docs[index];
          return Container(
            margin: EdgeInsets.all(10),
            child: Card(
              elevation: 6,
              shadowColor: Colors.grey,
              child: ListTile(
                title: Text(bus['name'].toString()),
                subtitle: Text("เดินทางจาก " + bus['origin'].toString() + " ไป " + bus['origin'].toString()),
                trailing: Row(
                  mainAxisAlignment: MainAxisAlignment.min,

```

```

children: [
  IconButton(
    onPressed: () {
      showDialog(
        context: context,
        builder: (BuildContext context) {
          _name.text = bus['name'];
          _origin.text = bus['origin'];
          _destination.text = bus['destination'];
          _numticket.text = bus['numticket'];
          _selecteddate.text = bus['selecteddate'];
          return StatefulBuilder(
            builder: (BuildContext context,
              void Function(void Function())
              setState) {
              return AlertDialog(
                actions: [
                  OutlinedButton(
                    onPressed: () {
                      Navigator.pop(context);
                    },
                    child: Text('ยกเลิก',
                      style: TextStyle(
                        color: Colors.white,
                        fontSize: 15)),
                    style: OutlinedButton.styleFrom(
                      backgroundColor:
                        Colors.red.shade700),
                  ),
                  OutlinedButton(
                    onPressed: () async {
                      setState(() {
                        _checkState = false;
                      });
                      var name =
                        _name.text.toString();
                      var origin =
                        _origin.text.toString();

```

```

var destination =
  _destination.text.toString();
var numticket =
  _numticket.text.toString();
var selecteddate =
  _selecteddate.text.toString();

await busCollection
  .doc(bus.id)
  .update({
    "name": name,
    "origin": origin,
    "destination": destination,
    "numticket": numticket,
    "selecteddate": selecteddate
  }).then((value) {
    setState(() {
      _checkState = true;
    });
    Navigator.pop(context);
    ScaffoldMessenger.of(context)
      .showSnackBar(SnackBar(
        content: Text(
          "แก้ไขข้อมูลสำเร็จ"),
        backgroundColor:
          Colors.green,
      )); // callbackFunction
  });
},
child: (_checkState)?Text(
  'แก้ไข',
  style: TextStyle(
    color: Colors.white,
    fontSize: 15),
):SizedBox(
  width: 16,
  height: 16,
  child:

```

```

CircularProgressIndicator(
  color: Colors.white,
  strokeWidth: 1.5,
),
),
style: OutlinedButton.styleFrom(
  backgroundColor: Colors
    .yellowAccent.shade700,
),
],
//หน้าต่างสีเทา
title: Text("แก้ไขข้อมูล"),
content: Column(
  mainAxisAlignment: MainAxisAlignment.min, //ขนาดเท่ากับฟอร์ม
  children: [
    TextField(
      controller: _name,
      decoration: InputDecoration(
        icon: Icon(Icons.people, color: Colors.green[900]),
        labelText: "ชื่อ-นามสกุล",
        border:
          OutlineInputBorder(borderRadius: BorderRadius.circular(40)),
      ),
    ),
    SizedBox(
      height: 10,
    ),
    TextField(
      controller: _id,
      decoration: InputDecoration(
        icon: Icon(Icons.person_remove_alt_1_rounded, color:
Colors.green[900]),

        labelText: "รหัสบัตรประชาชน",
        border:
          OutlineInputBorder(borderRadius: BorderRadius.circular(40)),
      ),
    ),
    SizedBox(

```

```

        height: 10,
      ),
      TextField(
        controller: _origin,
        decoration: InputDecoration(
          icon: Icon(Icons.location_on_sharp, color: Colors.green[900]),
          suffixIcon: Icon(Icons.arrow_drop_down),
          labelText: "ต้นทาง",
          border:
            OutlineInputBorder(borderRadius: BorderRadius.circular(40)),
        ),
        onTap: () {
          _showDropdown(context);
        },
        readOnly: true,
      ),
      SizedBox(
        height: 10,
      ),
      TextField(
        controller: _destination,
        decoration: InputDecoration(
          icon: Icon(Icons.location_on_sharp, color: Colors.green[900]),
          suffixIcon: Icon(Icons.arrow_drop_down),
          labelText: "ปลายทาง",
          border:
            OutlineInputBorder(borderRadius: BorderRadius.circular(40)),
        ),
        onTap: () {
          _showDropdown1(context);
        },
        readOnly: true,
      ),
      SizedBox(
        height: 10,
      ),
      DateTimeFormField(
        decoration: InputDecoration(

```

```

        icon: Icon(Icons.date_range, color: Colors.green[900]),
        labelText: 'วันเดินทางไป',
        border: OutlineInputBorder(
          borderRadius: BorderRadius.circular(40),
        ),
      ),
      firstDate: DateTime.now().add(const Duration(days: 10)),
      lastDate: DateTime.now().add(const Duration(days: 40)),
      mode: DateTimeFieldPickerMode.date,
      onChanged: (DateTime? value) {
        selectedDate = value;
      },
    ),
    SizedBox(
      height: 10,
    ),
    TextField(
      controller: _numticket,
      decoration: InputDecoration(
        icon: Icon(Icons.airplane_ticket_rounded, color: Colors.green[900]),
        labelText: "จำนวนตั๋ว",
        border:
          OutlineInputBorder(borderRadius: BorderRadius.circular(40)),
      ),
    ),
    SizedBox(
      height: 10,
    ),
  ],
),
);
},
);
},
icon: Icon(
  Icons.edit,

```

```

        color: Colors.amberAccent,
      )),
    IconButton(
      onPressed: () {
        showDialog(
          context: context,
          builder: (BuildContext context) {
            return StatefulBuilder(
              builder: (BuildContext context,
                void Function(void Function())
                setState) {
              return AlertDialog(
                title: Text("คุณต้องการลบหรือไม่?"),
                content: Text(" "),
                actions: [
                  TextButton(
                    onPressed: () {
                      Navigator.pop(context);
                    },
                    child: Text(
                      "ยกเลิก",
                      style: TextStyle(
                        color: Colors.red,
                      ),
                    ),
                  TextButton(
                    onPressed: () async {
                      setState(() {
                        _checkState = false;
                      });
                      await busCollection
                        .doc(bus.id)
                        .delete()
                        .then((value) {
                          ScaffoldMessenger.of(
                            context)
                            .showSnackBar(SnackBar(
                              content: Text(
                                "ลบข้อมูลสำเร็จ"),

```



```

        backgroundColor:
          Colors.green,
      ));
      Navigator.pop(context);
    });
  },
  child: (_checkState)
    ? Text(
      "ยืนยัน",
      style: TextStyle(
        color:
          Colors.green),
    )
    : SizedBox(
      width: 16,
      height: 16,
      child:
        CircularProgressIndicator(
          color: Colors.white,
          strokeWidth: 1.5,
        ),
    ),
  ],
);
},
);
},
);
},
);
},
icon: Icon(
  Icons.delete,
  color: Colors.pink,
))
],
),
),
),
);

```

```

        },
    );
},
),
),
);
}

void _showDropdown(BuildContext context) {
  showDialog(
    context: context,
    builder: (BuildContext context) {
      return AlertDialog(
        title: Text('ค้นหา'),
        content: DropdownButton<String>(
          value: dropdownValue,
          onChanged: (String? newValue) {
            setState(() {
              dropdownValue = newValue!;
              _origin.text = newValue; // Update text field value
            });
            Navigator.of(context).pop();
          },
          items: list.map<DropdownMenuItem<String>>((String value) {
            return DropdownMenuItem<String>(
              value: value,
              child: Text(value),
            );
          }).toList(),
        ),
      );
    },
  );
}

```

```

void _showDropdown1(BuildContext context) {
  showDialog(
    context: context,
    builder: (BuildContext context) {

```

```
return AlertDialog(  
  title: Text('ปายทาง'),  
  content: DropdownButton<String>(  
    value: dropdownValue1,  
    onChanged: (String? newValue) {  
      setState(() {  
        dropdownValue1 = newValue!  
        _destination.text = newValue; // Update text field value  
      });  
      Navigator.of(context).pop();  
    },  
    items: list1.map<DropdownMenuItem<String>>((String value) {  
      return DropdownMenuItem<String>(  
        value: value,  
        child: Text(value),  
      );  
    }).toList(),  
  ),  
);  
},  
);  
}
```