

```

// Automatic FlutterFlow imports
import '/backend/backend.dart';
import '/backend/schema/structs/index.dart';
import '/backend/schema/enums/enums.dart';
import '/actions/actions.dart' as action_blocks;
import '/flutter_flow/flutter_flow_theme.dart';
import '/flutter_flow/flutter_flow_util.dart';
import '/custom_code/widgets/index.dart'; // Imports other custom
widgets
import '/custom_code/actions/index.dart'; // Imports custom actions
import '/flutter_flow/custom_functions.dart'; // Imports custom
functions
import 'package:flutter/material.dart';
// Begin custom widget code
// DO NOT REMOVE OR MODIFY THE CODE ABOVE!

// Set your widget name, define your parameter, and then add the
// boilerplate code using the green button on the right!

class EmailDropdownButton extends StatefulWidget {
  const EmailDropdownButton({
    super.key,
    this.width,
    this.height,
    required this.emailList,
  });

  final double? width;
  final double? height;
  final List<String> emailList;

  @override
  State<EmailDropdownButton> createState() =>
    _EmailDropdownButtonState();
}

```

```

class _EmailDropDownButtonState extends State<EmailDropDownButton> {
  String? _selectedValue = null;
  static String _filter = "";
  GlobalKey _dropdownKey = GlobalKey();

  @override
  Widget build(BuildContext context) {
    // Provides an alternative dropdown for email search and
    selection.

    return Container(
      // This top margin ensures that the label text is right over
      // the dropdown widget's border.
      margin: EdgeInsets.only(top: 8),
      width: widget.width,
      height: widget.height,
      foregroundDecoration: BoxDecoration(
        borderRadius: BorderRadius.circular(8),
        border: Border(
          top: BorderSide(width: 0.5),
          left: BorderSide(width: 0.5),
          right: BorderSide(width: 0.5),
          bottom: BorderSide(width: 0.5),
        ),
      ),
      // The following Row is required for the Expanded widget.
      child: Row(children: [
        Expanded(
          flex: 3,
          child: DropdownButton<String>(
            key: _dropdownKey,
            isExpanded: true,
            isDense: true,
            dropdownColor: Colors.white,
            value: _selectedValue,
            hint: Text('Search for an Email address...'),

```

```

padding: EdgeInsets.only(left: 8, right: 8),
onChanged: (String? value) async {
  setState(() {
    _selectedValue = value!;
  });
},
items: [
  // This is the first element in the list used for
inserting a search
  // text which then is applied as a filter to the list of
emails.
  DropdownMenuItem<String>(
    value: "",
    child: Container(
      height: 40,
      padding: EdgeInsets.only(right: 8),
      color: Color(0),
      child: TextField(
        decoration: InputDecoration(
          border: OutlineInputBorder(
            borderRadius: BorderRadius.circular(8),
          ),
        ),
        // Set filter
        onChanged: (String newText) {
          setState(() {
            _filter = newText.toLowerCase();
          });
        },
      ),
    ),
  ),
  ..._buildDropdownItems()
],
),
),

```

```

    ]),
  );
}

List<DropdownMenuItem<String>> _buildDropdownItems() {
  // Build filtered email list.
  return widget.emailList
    .where((email) =>
email.toLowerCase().contains(_filter.toLowerCase()))
    .map<DropdownMenuItem<String>>((String value) {
      print(value);
      return DropdownMenuItem<String>(
        value: value,
        child: Text(value, style: TextStyle(fontSize: 14)),
      );
    }).toList();
}
}

```