IST 440W

Wonseok (Ryan) Song

Modification Code

On Pharmacy.dart I modified and enhanced my code into:

factory Pharmacy.fromJson(Map<String, dynamic> *json*) {  
 return Pharmacy(  
 *name*: *json*['name'],  
 *lat*: *json*['lat'],  
 *lon*: *json*['lon'],  
 *medicineQuantity*: *json*['medicineQuantity'],  
 *id*: *json*['id'],  
 *phoneNumber*: *json*['phoneNumber'],  
 *medicines*: List<MedicineModel.Medicine>.from(*json*['medicines'].map((*medicineJson*) => MedicineModel.Medicine.fromJson(*medicineJson*))),  
 *medicineName*: *json*['medicineName'],  
 );  
 }  
  
 Map<String, dynamic> toJson() {  
 return {  
 'name': name,  
 'lat': lat,  
 'lon': lon,  
 'medicineQuantity': medicineQuantity,  
 'id': id,  
 'phoneNumber': phoneNumber,  
 'medicines': medicines.map((*medicine*) => *medicine*.toJson()).toList(),  
 'medicineName': medicineName,  
 };  
 }  
}

On Pharmacy\_List\_Screen.dart

The codes that are modified and enhanced:

class PharmacyListScreen extends StatelessWidget {  
 @override  
 Widget build(BuildContext *context*) {  
 List<Pharmacy> pharmacies = Provider.*of*<PharmacyService>(*context*).searchResults;  
  
 return Scaffold(  
 *appBar*: AppBar(  
 *title*: const Text('Pharmacy List'),  
 ),  
 *body*: ListView.builder(  
 *itemCount*: pharmacies.length,  
 *itemBuilder*: (*context*, *index*) {  
 Pharmacy pharmacy = pharmacies[*index*];  
 return ListTile(  
 *title*: Text(pharmacy.name),  
 *subtitle*: Text('Medicine: ${pharmacy.medicines.first.name}'), *// Displaying the first medicine's name  
 trailing*: Text('Quantity: ${pharmacy.medicines[0].currentStock}'),  
 *onTap*: () {  
 Navigator.*push*(  
 *context*,  
 MaterialPageRoute(  
 *builder*: (*context*) => PharmacyDetailScreen(*pharmacy*: pharmacy), *// Pass the selected pharmacy to the detail screen* ),  
 );  
 },  
 );  
 },  
 ),  
 );  
 }  
}

On Screen\_Search.dart

I made changes and enhances in:

class SearchScreen extends StatelessWidget {  
 final TextEditingController \_searchController = TextEditingController();  
  
 @override  
 Widget build(BuildContext *context*) {  
 return Scaffold(  
 *appBar*: AppBar(  
 *title*: const Text('Search'),  
 ),  
 *body*: Padding(  
 *padding*: const EdgeInsets.all(16.0),  
 *child*: Column(  
 *crossAxisAlignment*: CrossAxisAlignment.stretch,  
 *children*: [  
 TextField(  
 *controller*: \_searchController,  
 *decoration*: const InputDecoration(  
 *labelText*: 'Search for medicine',  
 ),  
 ),  
 const SizedBox(*height*: 16.0),  
 ElevatedButton(  
 *onPressed*: () {  
 String medicineName = \_searchController.text;  
 if (medicineName.isNotEmpty) {  
 print('Searching for medicine: $medicineName');  
 Provider.*of*<PharmacyService>(*context*, *listen*: false)  
 .searchPharmaciesByMedicine(medicineName)  
 .then((*\_*) {  
 Navigator.*pushNamed*(*context*, '/list');  
 });  
 } else {  
 ScaffoldMessenger.*of*(*context*).showSnackBar(SnackBar(  
 *content*: Text('Search field is empty!'),  
 ));  
 }  
 },  
 *child*: const Text('List'),  
 ),  
 ],  
 ),  
 ),  
 );  
 }  
}

Database\_service.dart

I made modification and enhancement in:

class DatabaseService {  
 Future<List<PharmacyModel.Pharmacy>> getPharmaciesWithMedicine(String *medicineName*, Position *userPosition*) async {  
 List<PharmacyModel.Pharmacy> filteredPharmacies = dummyPharmacies.where((*pharmacy*) {  
 return *pharmacy*.medicines.any((*medicine*) =>  
 *medicine*.name.toLowerCase() == *medicineName*.toLowerCase());  
 }).toList();  
  
 return filteredPharmacies;  
 }  
}  
  
List<PharmacyModel.Pharmacy> dummyPharmacies = [  
PharmacyModel.Pharmacy(  
*name*: 'Pharmacy 1',  
*lat*: 37.7749,  
*lon*: -122.4194,  
*medicineQuantity*: 10,  
*id*: '1',  
*phoneNumber*: '+123456789',  
*medicines*: [  
MedicineModel.Medicine(*id*: '1', *name*: 'Medicine A', *currentStock*: 5),  
MedicineModel.Medicine(*id*: '2', *name*: 'Medicine B', *currentStock*: 3),  
MedicineModel.Medicine(*id*: '3', *name*: 'Medicine C', *currentStock*: 2),  
MedicineModel.Medicine(*id*: '4', *name*: 'Medicine D', *currentStock*: 4),  
MedicineModel.Medicine(*id*: '5', *name*: 'Medicine E', *currentStock*: 1),  
],  
*medicineName*: 'Medicine A',  
),  
PharmacyModel.Pharmacy(  
*name*: 'Pharmacy 2',  
*lat*: 37.7833,  
*lon*: -122.4167,  
*medicineQuantity*: 5,  
*id*: '2',  
*phoneNumber*: '+987654321',  
*medicines*: [  
 MedicineModel.Medicine(*id*: '1', *name*: 'Medicine A', *currentStock*: 6),  
 MedicineModel.Medicine(*id*: '6', *name*: 'Medicine F', *currentStock*: 6),  
 MedicineModel.Medicine(*id*: '7', *name*: 'Medicine G', *currentStock*: 2),  
 MedicineModel.Medicine(*id*: '8', *name*: 'Medicine H', *currentStock*: 3),  
 MedicineModel.Medicine(*id*: '9', *name*: 'Medicine I', *currentStock*: 1),  
],  
*medicineName*: 'Medicine F',  
),  
PharmacyModel.Pharmacy(  
*name*: 'Pharmacy 3',  
*lat*: 37.7914,  
*lon*: -122.4086,  
*medicineQuantity*: 8,  
*id*: '3',  
*phoneNumber*: '+111222333',  
*medicines*: [  
 MedicineModel.Medicine(*id*: '1', *name*: 'Medicine A', *currentStock*: 7),  
 MedicineModel.Medicine(*id*: '11', *name*: 'Medicine K', *currentStock*: 7),  
 MedicineModel.Medicine(*id*: '12', *name*: 'Medicine L', *currentStock*: 2),  
 MedicineModel.Medicine(*id*: '13', *name*: 'Medicine M', *currentStock*: 3),  
 MedicineModel.Medicine(*id*: '14', *name*: 'Medicine N', *currentStock*: 5),  
],  
*medicineName*: 'Medicine K',  
),  
PharmacyModel.Pharmacy(  
*name*: 'Pharmacy 4',  
*lat*: 37.7519,  
*lon*: -122.4342,  
*medicineQuantity*: 12,  
*id*: '4',  
*phoneNumber*: '+444555666',  
*medicines*: [  
 MedicineModel.Medicine(*id*: '1', *name*: 'Medicine A', *currentStock*: 8),  
 MedicineModel.Medicine(*id*: '15', *name*: 'Medicine O', *currentStock*: 6),  
 MedicineModel.Medicine(*id*: '16', *name*: 'Medicine P', *currentStock*: 2),  
 MedicineModel.Medicine(*id*: '17', *name*: 'Medicine Q', *currentStock*: 3),  
 MedicineModel.Medicine(*id*: '18', *name*: 'Medicine R', *currentStock*: 4),  
],  
*medicineName*: 'Medicine P',  
),  
PharmacyModel.Pharmacy(  
*name*: 'Pharmacy 5',  
*lat*: 37.8085,  
*lon*: -122.4199,  
*medicineQuantity*: 7,  
*id*: '5',  
*phoneNumber*: '+777888999',  
*medicines*: [  
 MedicineModel.Medicine(*id*: '1', *name*: 'Medicine A', *currentStock*: 9),  
 MedicineModel.Medicine(*id*: '19', *name*: 'Medicine S', *currentStock*: 7),  
 MedicineModel.Medicine(*id*: '20', *name*: 'Medicine T', *currentStock*: 3),  
 MedicineModel.Medicine(*id*: '21', *name*: 'Medicine U', *currentStock*: 2),  
 MedicineModel.Medicine(*id*: '22', *name*: 'Medicine V', *currentStock*: 5),  
],  
*medicineName*: 'Medicine U',  
)];

All these modification and enhancement are for removing the “Map” button and making more specified database for dummy pharmacies and medicines to pull out more data to query when user searches. I tried to provide the data on map by finding pharmacies base on users’ current location. However, I failed to do so. I removed the map feature. Focusing on the list feature to have better look of application for users. Putting two buttons with “list” and ‘Map” are unnecessary and inconvenient.

Result of Enhancement

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a medical application

Description automatically generated

Color-Coded Flow Chart

* Design Flow Chart

A diagram of a company

Description automatically generated

* Operation Flow Chart

