

File: lib/main.dart

```
import 'package:flutter/material.dart';
import 'screens/questionnaire_screen.dart';

void main() {
  runApp(const WellnessApp());
}

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

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Wellness Questionnaire',
      theme: ThemeData(
        primarySwatch: Colors.blue,
        useMaterial3: true,
      ),
      home: const QuestionnaireScreen(),
      debugShowCheckedModeBanner: false,
    );
  }
}
```

File: lib/models/result_model.dart

// Simple data model to hold each questionnaire result.

```
class QuestionnaireResult {
  final DateTime date;
  final String resultLabel;
  final String stage;
  final String region;
  final double sleepHours;

  QuestionnaireResult({
    required this.date,
    required this.resultLabel,
    required this.stage,
    required this.region,
    required this.sleepHours,
```

```
});  
}
```

File: lib/screens/questionnaire_screen.dart

```
import 'package:flutter/material.dart';  
import '../models/result_model.dart';  
import 'results_screen.dart';
```

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

```
class _QuestionnaireScreenState extends State<QuestionnaireScreen> {  
  final TextEditingController _sleepController = TextEditingController();
```

```
  String _selectedStage = 'Select your current stage';  
  String _selectedRegion = 'Select your region';
```

```
  bool? _appetite;  
  bool? _mood;  
  bool? _support;  
  bool? _history;
```

```
  final List<QuestionnaireResult> _historyList = [];
```

```
  final List<String> _stages = [  
    'Select your current stage',  
    'Preconception',  
    'Pregnancy',  
    'Postpartum',  
    'Other',  
  ];
```

```
  final List<String> _regions = [  
    'Select your region',  
    'North',  
    'South',  
    'East',  
    'West',
```

```

    'Central',
];

@override
void dispose() {
    _sleepController.dispose();
    super.dispose();
}

String _calculateRisk() {
    int score = 0;
    if (_appetite == true) score++;
    if (_mood == true) score++;
    if (_support == false) score++;
    if (_history == true) score++;

    return (score >= 2) ? 'Possible PPD Risk' : 'Low Risk';
}

void _onSubmit() {
    double sleep = double.tryParse(_sleepController.text) ?? 0;

    if (_selectedStage == _stages[0] || _selectedRegion == _regions[0]) {
        ScaffoldMessenger.of(context).showSnackBar(const SnackBar(content: Text('Please fill all
fields.')));
        return;
    }

    if (_appetite == null || _mood == null || _support == null || _history == null) {
        ScaffoldMessenger.of(context).showSnackBar(const SnackBar(content: Text('Please answer
all questions.')));
        return;
    }

    final result = QuestionnaireResult(
        date: DateTime.now(),
        resultLabel: _calculateRisk(),
        stage: _selectedStage,
        region: _selectedRegion,
        sleepHours: sleep,
    );

    setState(() => _historyList.insert(0, result));
}

```

```

Navigator.push(
  context,
  MaterialPageRoute(builder: (_) => ResultsScreen(latest: result, history: _historyList)),
);
}

Widget _questionCard({required IconData icon, required String text, required bool? value,
required Function(bool?) onChanged}) {
  return Container(
    decoration: BoxDecoration(
      borderRadius: BorderRadius.circular(12),
      color: Colors.white,
      border: Border.all(color: Colors.grey.shade300),
    ),
    padding: const EdgeInsets.all(18),
    margin: const EdgeInsets.only(bottom: 16),
    child: Column(crossAxisAlignment: CrossAxisAlignment.start, children: [
      Row(children: [
        Icon(icon, size: 22, color: Colors.black87),
        const SizedBox(width: 12),
        Expanded(child: Text(text, style: const TextStyle(fontSize: 16, fontWeight:
FontWeight.w600))),
      ]),
      const SizedBox(height: 12),
      Row(children: [
        Radio(value: true, groupValue: value, onChanged: onChanged), const Text('Yes'),
        const SizedBox(width: 30),
        Radio(value: false, groupValue: value, onChanged: onChanged), const Text('No'),
      ]),
    ]),
  );
}

```

```

@override
Widget build(BuildContext context) {
  return Scaffold(
    backgroundColor: const Color(0xffff4f4f),
    body: SafeArea(
      child: SingleChildScrollView(
        padding: const EdgeInsets.symmetric(horizontal: 20, vertical: 20),
        child: Column(crossAxisAlignment: CrossAxisAlignment.start, children: [
          const Text('Wellness Questionnaire', style: TextStyle(fontSize: 32, fontWeight:
FontWeight.w700, fontFamily: 'Inter')),
          const SizedBox(height: 6),

```

```

const Text('Please answer the following questions honestly. Your privacy is important to
us.', style: TextStyle(color: Colors.black54, fontSize: 15, fontFamily: 'Inter')),
const SizedBox(height: 28),(height: 6),
const Text('Please answer the following questions honestly. Your privacy is important to
us.', style: TextStyle(color: Colors.black54, fontSize: 14)),
const SizedBox(height: 28),

Row(children: [
  Expanded(
    child: Column(crossAxisAlignment: CrossAxisAlignment.start, children: [
      const Text('Current Stage', style: TextStyle(fontWeight: FontWeight.w600)),
      const SizedBox(height: 6),
      DropdownButtonFormField(
        value: _selectedStage,
        items: _stages.map((e) => DropdownMenuItem(value: e, child: Text(e))).toList(),
        onChanged: (v) => setState(() => _selectedStage = v!),
        decoration: InputDecoration(
          border: OutlineInputBorder(borderRadius: BorderRadius.circular(10)),
        ),
      ),
    ],
  ),
  const SizedBox(width: 14),
  Expanded(
    child: Column(crossAxisAlignment: CrossAxisAlignment.start, children: [
      const Text('Region', style: TextStyle(fontWeight: FontWeight.w600)),
      const SizedBox(height: 6),
      DropdownButtonFormField(
        value: _selectedRegion,
        items: _regions.map((e) => DropdownMenuItem(value: e, child: Text(e))).toList(),
        onChanged: (v) => setState(() => _selectedRegion = v!),
        decoration: InputDecoration(
          border: OutlineInputBorder(borderRadius: BorderRadius.circular(10)),
        ),
      ),
    ],
  ),
  const SizedBox(height: 28),
  const Text('Average sleep per night (hours)?', style: TextStyle(fontWeight:
FontWeight.w600)),
  const SizedBox(height: 6),
  TextField(

```

```

        controller: _sleepController,
        keyboardType: TextInputType.number,
        decoration: InputDecoration(
          hintText: 'e.g., 7',
          border: OutlineInputBorder(borderRadius: BorderRadius.circular(10)),
        ),
      ),

      const SizedBox(height: 28),
      _questionCard(icon: Icons.restaurant, text: 'Have you experienced significant changes in
your appetite?', value: _appetite, onChanged: (v) => setState(() => _appetite = v)),
      _questionCard(icon: Icons.sentiment_very_dissatisfied, text: 'Have you been
experiencing frequent mood swings?', value: _mood, onChanged: (v) => setState(() => _mood =
v)),
      _questionCard(icon: Icons.favorite, text: 'Do you feel you have adequate support from
your family/partner?', value: _support, onChanged: (v) => setState(() => _support = v)),
      _questionCard(icon: Icons.psychology, text: 'Have you had a history of depression or
other mental health concerns?', value: _history, onChanged: (v) => setState(() => _history = v)),

      const SizedBox(height: 10),
      SizedBox(
        width: double.infinity,
        child: ElevatedButton(
          style: ElevatedButton.styleFrom(
            backgroundColor: const Color(0xff5ba6f0),
            padding: const EdgeInsets.symmetric(vertical: 16),
            shape: RoundedRectangleBorder(borderRadius: BorderRadius.circular(8)),
          ),
          onPressed: _onSubmit,
          child: const Text('Submit & See Results', style: TextStyle(fontSize: 16, color:
Colors.white)),
        ),
      ),
    ],
  ),
);
}
}

```

File: lib/screens/results_screen.dart

```

import 'package:flutter/material.dart';
import '../models/result_model.dart';

```

```

class ResultsScreen extends StatelessWidget {
  final QuestionnaireResult latest;
  final List<QuestionnaireResult> history;

  const ResultsScreen({super.key, required
import 'package:flutter/material.dart';
import '../models/result_model.dart';

class ResultsScreen extends StatelessWidget {
  final QuestionnaireResult latest;
  final List<QuestionnaireResult> history;

  const ResultsScreen({super.key, required this.latest, required this.history});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: const Color(0xff7f7f7f),
      appBar: AppBar(title: const Text('Results History')),
      body: Padding(
        padding: const EdgeInsets.all(16.0),
        child: Column(crossAxisAlignment: CrossAxisAlignment.start, children: [
          const Text('Here are all the previous questionnaire results.', style: TextStyle(color:
Colors.black54)),
          const SizedBox(height: 20),

          Row(children: const [
            Expanded(child: Text('Date', style: TextStyle(fontWeight: FontWeight.w600))),
            Expanded(child: Text('Result', style: TextStyle(fontWeight: FontWeight.w600))),
            Expanded(child: Text('Stage', style: TextStyle(fontWeight: FontWeight.w600))),
            Expanded(child: Text('Sleep (hrs)', style: TextStyle(fontWeight: FontWeight.w600))),
          ]),
          const Divider(),

          Expanded(
            child: history.isEmpty
              ? const Center(child: Text('No results yet.'))
              : ListView.builder(
                  itemCount: history.length,
                  itemBuilder: (_, i) {
                    final r = history[i];
                    return Padding(
                      padding: const EdgeInsets.symmetric(vertical: 8.0),

```

```

        child: Row(children: [
          Expanded(child: Text('N/A')), // Matching screenshot
          Expanded(
            child: Container(
              padding: const EdgeInsets.symmetric(horizontal: 10, vertical: 4),
              decoration: BoxDecoration(
                color: r.resultLabel == 'Possible PPD Risk' ? Colors.red.shade300 :
Colors.green.shade300,
                borderRadius: BorderRadius.circular(20),
              ),
              child: Center(
                child: Text(r.resultLabel, style: const TextStyle(color: Colors.white,
fontWeight: FontWeight.w600)),
              ),
            ),
          ),
          Expanded(child: Text(r.stage)),
          Expanded(child: Text(r.sleepHours.toString())),
        ]),
      );
    }, />
import 'package:flutter/material.dart';
import '../models/result_model.dart';

class ResultsScreen extends StatelessWidget {
  final QuestionnaireResult latest;
  final List<QuestionnaireResult> history;

  const ResultsScreen({super.key, required this.latest, required this.history});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(title: const Text('Results')),
      body: Padding(
        padding: const EdgeInsets.all(16.0),
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.start,
          children: [
            Text('Your Result', style: Theme.of(context).textTheme.headlineSmall),
            const SizedBox(height: 12),
            Card(
              child: ListTile(
                title: Text(latest.resultLabel, style: const TextStyle(fontWeight: FontWeight.bold)),

```



```

        subtitle: Text('Stage: ${latest.stage} • Sleep: ${latest.sleepHours} hrs'),
      ),
    ),

    const SizedBox(height: 20),
    const Text('Results History', style: TextStyle(fontSize: 18, fontWeight: FontWeight.bold)),
    const SizedBox(height: 8),

    Expanded(
      child: history.isEmpty
        ? const Center(child: Text('No previous results found.'))
        : ListView.builder(
            itemCount: history.length,
            itemBuilder: (context, index) {
              final item = history[index];
              return Card(
                margin: const EdgeInsets.symmetric(vertical: 6),
                child: ListTile(
                  title: Text(item.resultLabel),
                  subtitle: Text('${item.date.toLocal().toString().split(".").first} • ${item.stage}'),
                  trailing: Text(item.sleepHours.toString()),
                ),
              );
            },
          ),
    ),
  ],
),
),
);
}
}

```

File structure :

```
wellness_app/  
├─ android/  
├─ ios/  
├─ linux/  
├─ macos/  
├─ web/  
├─ windows/  
├─ pubspec.yaml  
├─ lib/  
│   ├── main.dart  
│   ├── screens/  
│   │   ├── questionnaire_screen.dart  
│   │   └─ results_screen.dart  
│   └─ models/  
│       └─ result_model.dart  
└─ test/
```