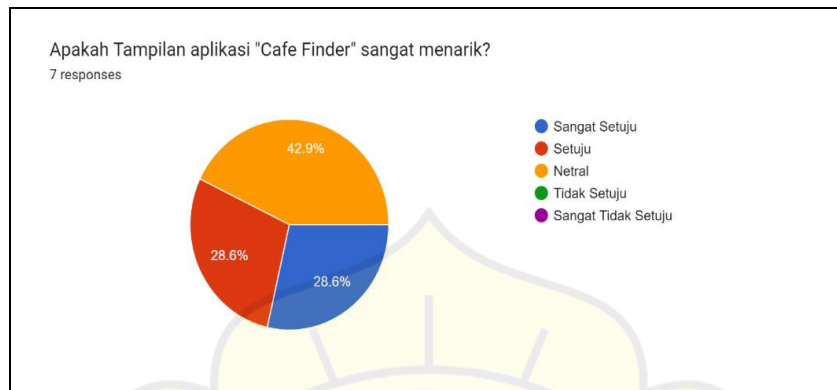
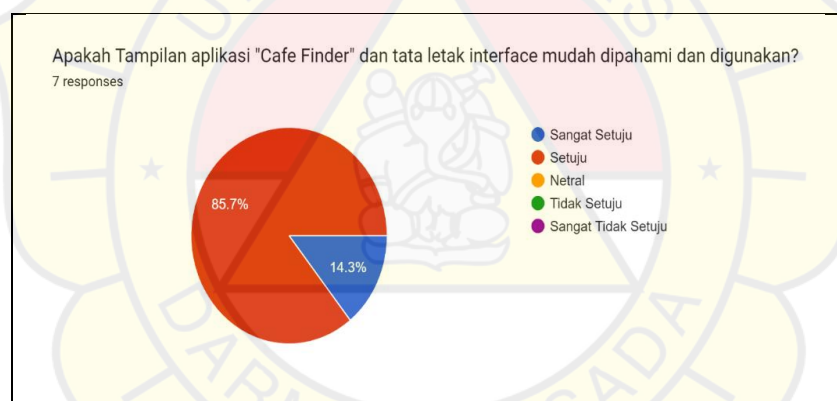


LAMPIRAN 1

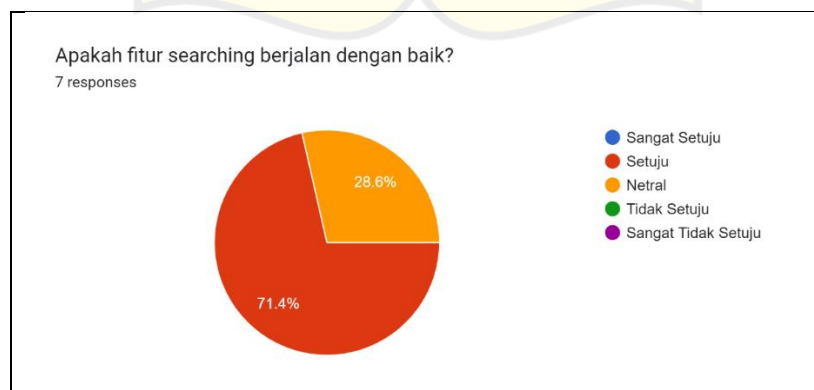
Lampiran Hasil Kuisisioner



Gambar Lampiran 1.1 Hasil Kuisisioner Pertanyaan 1



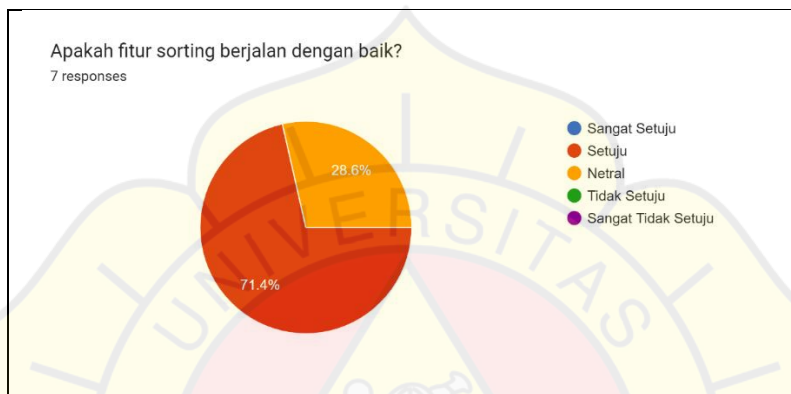
Gambar Lampiran 1.2 Hasil Kuisisioner Pertanyaan 2



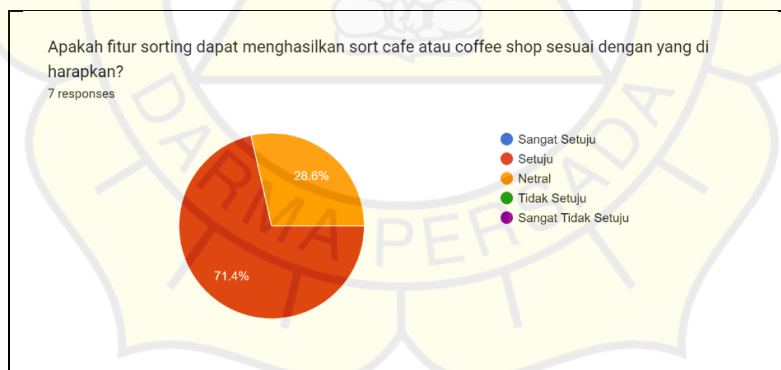
Gambar Lampiran 1.3 Hasil Kuisisioner Pertanyaan 3



Gambar Lampiran 1.4 Hasil Kuisisioner Pertanyaan 4



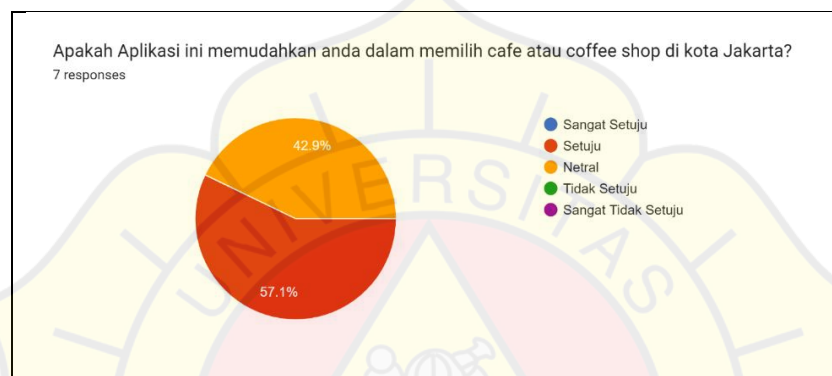
Gambar Lampiran 1.5 Hasil Kuisisioner Pertanyaan 5



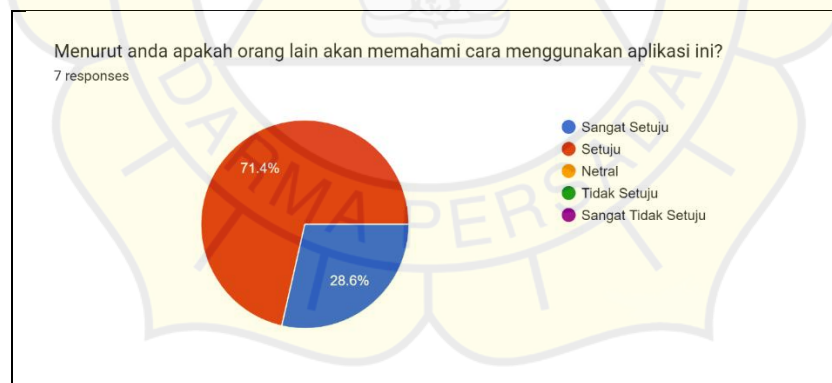
Gambar Lampiran 1.6 Hasil Kuisisioner Pertanyaan 6



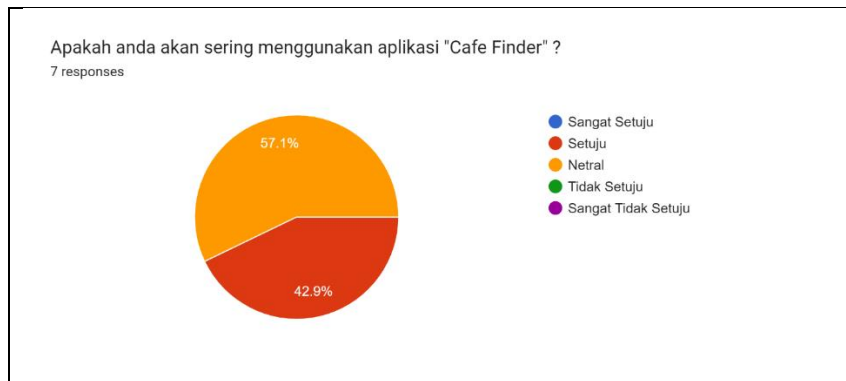
Gambar Lampiran 1.7 Hasil Kuisisioner Pertanyaan 7



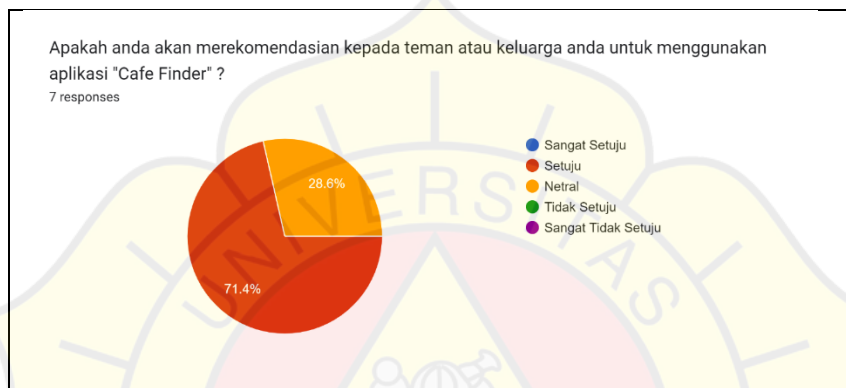
Gambar Lampiran 1.8 Hasil Kuisisioner Pertanyaan 8



Gambar Lampiran 1.9 Hasil Kuisisioner Pertanyaan 9



Gambar Lampiran 1.10 Hasil Kuisisioner Pertanyaan 10



Gambar Lampiran 1.11 Hasil Kuisisioner Pertanyaan 11

Pengujian UAT(*User Acceptance Test*)

Tabel Lampiran 1.1 Data Jawaban Responden

No	Pertanyaan	Jawaban					Presentase				
		STS	TS	N	S	SS	STS	TS	N	S	SS
1.	Apakah Tampilan aplikasi "Cafe Finder" sangat menarik?	0	0	3	2	2	0%	0%	42,9%	28,6%	28,6%
2.	Apakah Tampilan aplikasi "Cafe Finder" dan tata letak interface mudah dipahami dan digunakan?	0	0	0	6	1	0%	0%	0%	85,7%	14,3%
3.	Apakah fitur searching berjalan dengan baik?	0	0	2	5	0	0%	0%	28,6%	71,4%	0%
4.	Apakah fitur searching dapat menghasilkan jawaban sesuai dengan yang di harapkan?	0	0	3	4	0	0%	0%	42,9%	57,1%	0%
5.	Apakah fitur sorting berjalan dengan baik?	0	0	2	5	0	0%	0%	28,6%	71,4%	0%
6.	Apakah fitur sorting dapat menghasilkan sort cafe atau coffee shop sesuai dengan yang di harapkan?	0	0	2	5	0	0%	0%	28,6%	71,4%	0%
7.	Apakah halaman rekomendasi menghasilkan perankingan cafe atau coffee shop sesuai dengan hasil yang di harapkan?	0	0	4	3	0	0%	0%	57,1%	42,9%	0%
8.	Apakah Aplikasi ini memudahkan anda dalam	0	0	3	4	0					

	memilih cafe atau coffee shop di kota Jakarta?						0%	0%	42,9%	57,1%	0%
9.	Menurut anda apakah orang lain akan memahami cara menggunakan aplikasi ini?	0	0	0	5	2	0%	0%	0%	71,4%	28,6%
10.	Apakah anda akan sering menggunakan aplikasi "Cafe Finder" ?	0	0	3	4	0	0%	0%	57,1%	42,9%	0%
11.	Apakah anda akan merekomendasikan kepada teman atau keluarga anda untuk menggunakan aplikasi "Cafe Finder" ?	0	0	2	5	0	0%	0%	28,6%	71,4%	0%

Tabel Lampiran 1.2 Hasil Penilaian

NO	PERTANYAAN	Nilai					Jumlah
		SS x 5	S x 4	N x 3	TS x 2	STS x 1	
1.	Apakah Tampilan aplikasi "Cafe Finder" sangat menarik?	10	8	9	0	0	27
2.	Apakah Tampilan aplikasi "Cafe Finder" dan tata letak interface mudah dipahami dan digunakan?	5	24	0	0	0	29
3.	Apakah fitur searching	0	20	6	0	0	26

	berjalan dengan baik?						
4.	Apakah fitur searching dapat menghasilkan jawaban sesuai dengan yang di harapkan?	0	16	9	0	0	25
5.	Apakah fitur sorting berjalan dengan baik?	0	20	6	0	0	26
6.	Apakah fitur sorting dapat menghasilkan sort cafe atau coffee shop sesuai dengan yang di harapkan?	0	20	6	0	0	26
7.	Apakah halaman rekomendasi menghasilkan perankingan cafe atau coffee shop sesuai dengan hasil yang di 3harapkan?	0	12	12	0	0	24
8.	Apakah Aplikasi ini memudahkan anda dalam memilih cafe atau coffee shop di kota Jakarta?	0	16	9	0	0	25
9.	Menurut anda apakah orang lain akan memahami cara menggunakan aplikasi ini?	10	20	0	0	0	30
10.	Apakah anda akan sering menggunakan	0	16	9	0	0	25

	aplikasi "Cafe Finder" ?						
11.	Apakah anda akan merekomendasikan kepada teman atau keluarga anda untuk menggunakan aplikasi "Cafe Finder" ?	0	20	6	0	0	26

1. Analisa Pertanyaan Pertama

Dari Tabel di atas dapat dilihat bahwa jumlah nilai dari 7 responden untuk pertanyaan pertama adalah 27. Nilai rata-ratanya adalah $27/7 = 3,8$. Presentase nilai nya adalah $3,8/5 \times 100 = 76\%$.

2. Analisa Pertanyaan Kedua

Dari Tabel di atas dapat dilihat bahwa jumlah nilai dari 7 responden untuk pertanyaan kedua adalah 29. Nilai rata-ratanya adalah $29/7 = 4,1$. Presentase nilai nya adalah $4,1/5 \times 100 = 82\%$.

3. Analisa Pertanyaan Ketiga

Dari Tabel di atas dapat dilihat bahwa jumlah nilai dari 7 responden untuk pertanyaan ketiga adalah 26. Nilai rata-ratanya adalah $26/7 = 3,7$. Presentase nilai nya adalah $3,7/5 \times 100 = 74\%$.

4. Analisa Pertanyaan Keempat

Dari Tabel di atas dapat dilihat bahwa jumlah nilai dari 7 responden untuk pertanyaan keempat adalah 25. Nilai rata-ratanya adalah $25/7 = 3,5$. Presentase nilai nya adalah $3,5/5 \times 100 = 70\%$.

5. Analisa Pertanyaan Kelima

Dari Tabel di atas dapat dilihat bahwa jumlah nilai dari 7 responden untuk pertanyaan kelima adalah 26. Nilai rata-ratanya adalah $26/7 = 3,7$. Presentase nilai nya adalah $3,7/5 \times 100 = 74\%$.

6. Analisa Pertanyaan Keenam

Dari Tabel di atas dapat dilihat bahwa jumlah nilai dari 7 responden untuk pertanyaan keenam adalah 26. Nilai rata-ratanya adalah $26/7 = 3,7$. Presentase nilai nya adalah $3,7/5 \times 100 = 74\%$.

7. Analisa Pertanyaan Ketujuh

Dari Tabel di atas dapat dilihat bahwa jumlah nilai dari 7 responden untuk pertanyaan ketujuh adalah 24. Nilai rata-ratanya adalah $24/7 = 3,4$. Presentase nilai nya adalah $3,4/5 \times 100 = 68\%$.

8. Analisa Pertanyaan Kedelapan

Dari Tabel di atas dapat dilihat bahwa jumlah nilai dari 7 responden untuk pertanyaan kedelapan adalah 25. Nilai rata-ratanya adalah $25/7 = 3,5$. Presentase nilai nya adalah $3,5/5 \times 100 = 70\%$.

9. Analisa Pertanyaan Kesembilan

Dari Tabel di atas dapat dilihat bahwa jumlah nilai dari 7 responden untuk pertanyaan kesembilan adalah 30. Nilai rata-ratanya adalah $30/7 = 4,3$. Presentase nilai nya adalah $4,3/5 \times 100 = 86\%$.

10. Analisa Pertanyaan Kesepuluh

Dari Tabel di atas dapat dilihat bahwa jumlah nilai dari 7 responden untuk pertanyaan kesepuluh adalah 25. Nilai rata-ratanya adalah $25/7 = 3,5$. Presentase nilai nya adalah $3,5/5 \times 100 = 70\%$.

11. Analisa Pertanyaan Kesebelas

Dari Tabel di atas dapat dilihat bahwa jumlah nilai dari 7 responden untuk pertanyaan adalah 26. Nilai rata-ratanya adalah $26/7 = 3,7$. Presentase nilai nya adalah $3,7/5 \times 100 = 74\%$.

Perhitungan Keseluruhan:

Perhitungan Keseluruhan merupakan hasil perhitungan rata-rata skor dari seluruh pertanyaan.

Adapun hasil perhitungan keseluruhan pada *user acceptance test* oleh responden ini adalah sebagai berikut:

$$(76\% + 82\% + 74\% + 70\% + 74\% + 74\% + 68\% + 70\% + 86\% + 70\% + 74\%) / 11 = 74,3\%$$

LAMPIRAN 2

Source Code

Kode Program file main.dart

```
1.  import 'package:cafe_finder_fireb/firebase_options.dart';
2.  import 'package:cafe_finder_fireb/src/features/core/bloc/blocs.dart';
3.  import 'package:cafe_finder_fireb/src/features/controllers/screen_controller.dart';
4.  import 'package:cafe_finder_fireb/src/routes/approuter.dart';
5.  import 'package:cafe_finder_fireb/src/utils/theme/this_theme.dart';
6.  import 'package:shared_preferences/shared_preferences.dart';
7.  import 'package:firebase_core/firebase_core.dart';
8.  import 'package:flutter/material.dart';
9.  import 'package:flutter_bloc/flutter_bloc.dart';
10. import 'package:flutter_dotenv/flutter_dotenv.dart';
11. import 'src/features/core/repository/export_repo.dart';
12. late SharedPreferences sharedPreferences;
13. void main() async {
14.   WidgetsFlutterBinding.ensureInitialized();
15.   sharedPreferences = await SharedPreferences.getInstance();
16.   await dotenv.load(fileName: "assets/config/.env");
17.   await Firebase.initializeApp(
18.     options: DefaultFirebaseOptions.currentPlatform,
19.   );
20.   runApp(const MyApp());
21. }
22. class MyApp extends StatelessWidget {
23.   const MyApp({
24.     Key? key,
25.   }) : super(key: key);
26.   @override
27.   Widget build(BuildContext context) {
28.     return MultiRepositoryProvider(
29.       providers: [
30.         RepositoryProvider(
31.           create: (context) => Auth(),
32.         ),
33.         RepositoryProvider(
34.           create: (context) => CoffeeRepo(),
35.         ),
36.         RepositoryProvider(
37.           create: (context) => CoffeeRepo2(),
38.         ),
39.       ],
40.       child: MultiBlocProvider(
41.         providers: [
42.           BlocProvider(
```

```

43.   create: (context) => AuthuserBloc(),
44.   ),
45.   BlocProvider(
46.     create: (context) => CoffeeshopsBloc(
47.       coffeeRepo: context.read<CoffeeRepo>(),
48.     ),
49.   ),
50.   BlocProvider(
51.     create: (context) => ByRateBloc(
52.       coffeeRepo: context.read<CoffeeRepo2>(),
53.     ),
54.   ),
55.   BlocProvider(
56.     create: (context) => ByDistanceBloc(
57.       coffeeRepo: context.read<CoffeeRepo>(),
58.     ),
59.   ),
60.   BlocProvider(
61.     create: (context) => ByPriceBloc(
62.       coffeeRepo: context.read<CoffeeRepo>(),
63.     ),
64.   ),
65.   BlocProvider(
66.     create: (context) => MabacMethodBloc(
67.       coffeeRepo: context.read<CoffeeRepo>(),
68.     ),
69.   ),
70. ],
71. child: MaterialApp(
72.   theme: TAppTheme.lightTheme,
73.   darkTheme: TAppTheme.darkTheme,
74.   themeMode: ThemeMode.system,
75.   debugShowCheckedModeBanner: false,
76.   onGenerateRoute: AppRouter.onGenerateRoute,
77.   initialRoute: ScreenController.routeName,
78. ),
79. );
80. };
81. }
82. }

```

Kode Program file base_repo.dart

```

83.   import '../models/coffeestore_model.dart';
84.   abstract class BaseRepo {
85.     Stream<List<Coffeeshops>> getCoffeeshops();
86.   }

```

Kode Program file coffeeshop_repository.dart

```
87. class CoffeeRepo extends BaseRepo {
88.   final FirebaseFirestore
89.   _firebaseFirestore; //inisiasi variabel untuk menyimpan objek FirebaseFirestore
90.   CoffeeRepo({
91.     //menginisiasi variabel _firebaseFirestore dengan objek firebaseFirestore atau objek
     FirebaseFirestore.instance
92.     FirebaseFirestore? firebaseFirestore,
93.   }) : _firebaseFirestore = firebaseFirestore ?? FirebaseFirestore.instance;
94.   @override
95.   ///fungsi untuk mengambil daftar coffee shops
96.   Stream<List<Coffeeshops>> getCoffeeshops() {
97.     //mengambil data dari koleksi 'places' di Cloud Firestore dan mengembalikan stream data
     dalam bentuk List<Coffeeshops>
98.     return _firebaseFirestore.collection('places').snapshots().map((snapshot) {
99.       //mapping data ke model Coffeeshops
100.      return snapshot.docs.map((doc) => Coffeeshops.fromJson(doc)).toList();
101.    });
102.  }
103. }
```

Kode Program file coffeeshops_bloc.dart

```
104. import 'dart:async';
105. import 'package:bloc/bloc.dart';
106. import
    'package:cafe_finder_fireb/src/features/core/repository/store/coffeeshop_repositories.dart';
107. import 'package:equatable/equatable.dart';
108. import '../models/coffeeshop_model.dart';
109. part 'coffeeshops_event.dart';
110. part 'coffeeshops_state.dart';
111. class CoffeeshopsBloc extends Bloc<CoffeeshopsEvent, CoffeeshopsState> {
112.   final CoffeeRepo _coffeeRepo;
113.   StreamSubscription? _coffeeSubscription; //objek StreamSubscription yang digunakan untuk
     mendengarkan perubahan data pada server Firebase.
114.   CoffeeshopsBloc({
115.     required CoffeeRepo coffeeRepo, // instance dari class CoffeeRepo, yang digunakan untuk
     melakukan koneksi ke Firebase untuk mengambil data coffeeshop
116.   }) : _coffeeRepo = coffeeRepo,
117.       super(CoffeeshopLoading()) {
118.     on<LoadCoffeeshops>(_onLoadCoffeeshops);
119.     _coffeeSubscription = _coffeeRepo.getCoffeeshops().listen(
120.       (coffeeshops) => add(LoadCoffeeshops(coffeeshops)),
121.     );
122.   }
123.   /** Method ini dipanggil ketika event LoadCoffeeshops terjadi.
124.   Method ini akan memperbarui state dari CoffeeshopsBloc menjadi CoffeeshopLoaded
     dengan daftar toko kopi yang diterima sebagai payload dari event. */
125.   void _onLoadCoffeeshops(
126.     LoadCoffeeshops event,
```

```

127. Emitter<CoffeeshopsState> emit,
128. ){
129. emit(CoffeeshopLoaded(event.coffeeshops));
130. }
131. @override
132. /** Method close() yang dipanggil ketika state management sudah tidak diperlukan lagi.
133. method close() akan membatalkan langganan _coffeeSubscription dan menghapus state
    yang ada.
134. */
135. Future<void> close() async {
136. _coffeeSubscription?.cancel();
137. super.close();
138. }
139. }

```

Kode Program file place_list.dart

```

140. import 'package:cafe_finder_fireb/src/common_widgets/animation/loading_widget.dart';
141. import 'package:cafe_finder_fireb/src/constants/color.dart';
142. import
    'package:cafe_finder_fireb/src/features/screens/home/widgets/home_searchbox_widget.dart';
143. import 'package:flutter/material.dart';
144. import 'package:flutter_bloc/flutter_bloc.dart';
145. import 'package:cafe_finder_fireb/src/constants/sizes.dart';
146. import 'package:cafe_finder_fireb/src/constants/text_string.dart';
147. import 'package:cafe_finder_fireb/src/features/screens/home/list-page/extends/list-
    card.dart';
148. // import
    'package:cafe_finder_fireb/src/features/core/screens/home/widgets/header_image_widget.
    dart';
149. import '../..../core/bloc/places_bloc/by_distance/by_distance_bloc.dart';
150. import '../..../core/bloc/places_bloc/by_price/by_price_bloc.dart';
151. import '../..../core/bloc/places_bloc/by_rate/byrate_bloc.dart';
152. import '../..../core/bloc/places_bloc/coffeeshops_bloc.dart';
153. class PlacesList extends StatefulWidget {
154. const PlacesList({
155. Key? key,
156. }) : super(key: key);
157. @override
158. State<PlacesList> createState() => _PlacesListState();
159. }
160. class _PlacesListState extends State<PlacesList> with TickerProviderStateMixin {
161. @override
162. Widget build(BuildContext context) {
163. var mediaQuery = MediaQuery.of(context);
164. var brightness = mediaQuery.platformBrightness;
165. final isDarkMode = brightness == Brightness.dark;
166. TabController tabController = TabController(
167. animationDuration: const Duration(seconds: 2), length: 4, vsync: this);
168. return SingleChildScrollView(

```



```

169. child: Container(
170.   padding: const EdgeInsets.all(tDashboardPadding),
171.   child: Column(
172.     crossAxisAlignment: CrossAxisAlignment.start,
173.     children: [
174.       Text(
175.         tDashboardTitle,
176.         style: Theme.of(context).textTheme.bodyText2,
177.       ),
178.       Text(
179.         tDashboardHeading,
180.         style: Theme.of(context).textTheme.headline2,
181.       ),
182.       const SizedBox(
183.         height: tDashboardPadding + 10,
184.       ),
185.       SizedBox(
186.         height: 55,
187.         child: ListView(
188.           scrollDirection: Axis.horizontal,
189.           physics: const NeverScrollableScrollPhysics(),
190.           children: const [
191.             // ButtonAddField(),
192.             DashboardSearchBox(),
193.             SizedBox(
194.               height: tDashboardPadding,
195.             ),
196.           ],
197.         ),
198.       ),
199.       const SizedBox(
200.         height: tDashboardPadding,
201.       ),
202.       Container(
203.         color: Colors.transparent,
204.         child: Align(
205.           alignment: Alignment.centerLeft,
206.           child: TabBar(
207.             indicatorColor: isDarkMode ? Colors.amber.shade600 : tEals,
208.             labelPadding: const EdgeInsets.only(
209.               left: tDashboardPadding, right: tDashboardPadding),
210.             labelColor: isDarkMode ? white : black,
211.             unselectedLabelColor:
212.               isDarkMode ? Colors.white24 : Colors.black26,
213.             controller: tabController,
214.             isScrollable: true,
215.             labelStyle: Theme.of(context).textTheme.headline4,
216.             tabs: const [
217.               Tab(
218.                 text: 'All',
219.               ),

```



```

220. Tab(
221.   text: 'Top Rated',
222. ),
223. Tab(
224.   text: 'Nearest',
225. ),
226. Tab(
227.   text: 'Lowest Price',
228. ),
229. ],
230. ),
231. ),
232. ),
233. const SizedBox(
234.   height: tDashboardCardPadding,
235. ),
236. Container(
237.   height: 400,
238.   width: double.maxFinite,
239.   color: Colors.transparent,
240.   child: TabBarView(controller: tabController, children: [
241.     // ---- All Place
242.     BlocBuilder<CoffeeshopsBloc, CoffeeshopsState>(
243.       builder: (context, state) {
244.         if (state is CoffeeshopLoading) {
245.           return const Center(
246.             child: LoadingWidget(),
247.           );
248.         }
249.         if (state is CoffeeshopLoaded) {
250.           return SizedBox(
251.             width: double.infinity,
252.             height: 400,
253.             child: Padding(
254.               padding:
255.                 const EdgeInsets.only(right: 0, bottom: 5.0),
256.               child: ListView.builder(
257.                 shrinkWrap: true,
258.                 scrollDirection: Axis.vertical,
259.                 itemCount: state.coffeeshops.length,
260.                 itemBuilder: (context, index) {
261.                   final newStateOf = state.coffeeshops;
262.                   return CardWidget(
263.                     coffeeshops: newStateOf[index],
264.                   );
265.                 },
266.               ),
267.             );
268.         } else {
269.           return Container();
270.         }

```



```

271. },
272. ),
273. //---- Sort by Rate
274. BlocBuilder<ByRateBloc, ByRateState>(
275.   builder: (context, state) {
276.     if (state is ByRateLoading) {
277.       return const Center(
278.         child: LoadingWidget(),
279.       );
280.     }
281.     if (state is ByRateLoaded) {
282.       return SizedBox(
283.         width: double.infinity,
284.         height: 400,
285.         child: Padding(
286.           padding:
287.             const EdgeInsets.only(right: 0, bottom: 5.0),
288.           child: ListView.builder(
289.             shrinkWrap: true,
290.             scrollDirection: Axis.vertical,
291.             itemCount: state.coffeeshops.length,
292.             itemBuilder: (context, index) {
293.               return CardWidget(
294.                 coffeeshops: state.coffeeshops[index],
295.               );
296.             }},
297.           ),
298.         );
299.     } else {
300.       return const Text('Something is missing');
301.     }
302.   },
303. ),
304. //---- Sort by Distance
305. BlocBuilder<ByDistanceBloc, ByDistanceState>(
306.   builder: (context, state) {
307.     if (state is ByDistanceLoading) {
308.       return const Center(
309.         child: LoadingWidget(),
310.       );
311.     }
312.     if (state is ByDistanceLoaded) {
313.       return SizedBox(
314.         width: double.infinity,
315.         height: 400,
316.         child: Padding(
317.           padding:
318.             const EdgeInsets.only(right: 0, bottom: 5.0),
319.           child: ListView.builder(
320.             shrinkWrap: true,
321.             scrollDirection: Axis.vertical,

```

```

322. itemCount: state.coffeeshops.length,
323. itemBuilder: (context, index) {
324.   final dist =
325.     state.coffeeshops[index].distance ??
326.     0.0.toDouble();
327.   final newStateOfDistance = state.coffeeshops;
328.   if (dist < 2.0) {
329.     newStateOfDistance.sort(((a, b) =>
330.       a.distance!.compareTo(b.distance!)));
331.     return CardWidget(
332.       coffeeshops: newStateOfDistance[index],
333.     );
334.   }
335.   return Container();
336. }},
337. ),
338. );
339. } else {
340.   return Container();
341. }
342. },
343. ),
344. //---- Sort by Lowest Price
345. BlocBuilder<ByPriceBloc, ByPriceState>{
346.   builder: (context, state) {
347.     if (state is ByPriceLoading) {
348.       return const Center(
349.         child: LoadingWidget(),
350.       );
351.     }
352.     if (state is ByPriceLoaded) {
353.       return SizedBox(
354.         width: double.infinity,
355.         height: 400,
356.         child: Padding(
357.           padding:
358.             const EdgeInsets.only(right: 0, bottom: 5.0),
359.           child: ListView.builder(
360.             shrinkWrap: true,
361.             scrollDirection: Axis.vertical,
362.             itemCount: state.coffeeshops.length,
363.             itemBuilder: (context, index) {
364.               final newStateOfPrice = state.coffeeshops;

365.               newStateOfPrice[index].forTwo ?? 0.toInt();
366.               newStateOfPrice.sort(((a, b) =>
367.                 a.forTwo!.compareTo(b.forTwo!)));
368.               return CardWidget(
369.                 coffeeshops: newStateOfPrice[index],
370.               );
371.             }},

```

```
372. ),  
373. );  
374. } else {  
375.   return Container();  
376. }  
377. },  
378. ),  
379. ])),  
380. ],  
381. ),  
382. ),  
383. );
```



Unsada Perpustakaan

Muhammad Noval Rifani

-  Quick Submit
-  Quick Submit
-  Universitas Darma Persada

Document Details

Submission ID

trn:oid::1:3121221029

Submission Date

Dec 23, 2024, 10:19 AM GMT+7

Download Date

Dec 23, 2024, 11:16 AM GMT+7

File Name

2015230033_Muhammad_Noval_Rifani_-_Muhammad_Noval_Rifani.docx

File Size

2.4 MB

65 Pages

7,083 Words

44,528 Characters

22% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Top Sources

- 22%  Internet sources
- 10%  Publications
- 0%  Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

