

## SmartBridge Assignment-4

Fareed Durgam 21BCE3532 Vellore Vit

### App Development

### Restaurant App

```
34 class MainActivity : AppCompatActivity() {
35     override fun onCreate(savedInstanceState: Bundle?) {
36         super.onCreate(savedInstanceState)
37         setContent {
38             val navController = rememberNavController()
39             NavHost(navController, startDestination = "Login") { this: NavGraphBuilder
40                 composable(route: "login") { LoginScreen(navController) }
41                 composable(route: "home") { HomeScreen(navController) }
42                 composable(route: "details/{imageId}") { backStackEntry ->
43                     val imageId = backStackEntry.arguments?.getString(key: "imageId")
44                     when (imageId) {
45                         "shawarma" -> ScreenOne(navController)
46                         "pizza" -> ScreenTwo(navController)
47                         "biryani" -> ScreenThree(navController)
48                     } else -> {
49
50                     }
51                 }
52             }
53         }
54     }
55 }
56
57 }
```

```
61 @OptIn(ExperimentalMaterial3Api::class)
62 @Composable
63 fun LoginScreen(navController: NavHostController) {
64     val context = LocalContext.current
65     var username by remember { mutableStateOf(value: "") }
66     var password by remember { mutableStateOf(value: "") }
67
68     Box(
69         modifier = Modifier
70             .fillMaxSize()
71             .background(Color.DarkGray)
72     ) { this: BoxScope
73         // Background Image
74         Image(
75             painter = painterResource(id = R.drawable.food),
76             contentDescription = "Image",
77             modifier = Modifier.fillMaxSize(),
78             contentScale = ContentScale.Crop
79         )
80
81         Column(
82             modifier = Modifier
83                 .fillMaxSize()
84                 .padding(16.dp),
85             horizontalAlignment = Alignment.CenterHorizontally,
86             verticalArrangement = Arrangement.Center
87         )
88     }
```

```

85         horizontalAlignment = Alignment.CenterHorizontally,
86         verticalArrangement = Arrangement.Center
87     ) { this: ColumnScope
88         TextField(
89             value = username,
90             onValueChange = { username = it },
91             label = { Text(text: "Username") },
92             modifier = Modifier.padding(16.dp)
93         )
94
95         TextField(
96             value = password,
97             onValueChange = { password = it },
98             label = { Text(text: "Password") },
99             keyboardOptions = KeyboardOptions.Default.copy(imeAction = ImeAction.Done),
100             keyboardActions = KeyboardActions(
101                 onDone = { this: KeyboardActionScope
102                     navController.navigate(route: "home")
103                 }
104             ),
105             modifier = Modifier.padding(16.dp)
106         )
107
108         Button(
109             onClick = {
110                 navController.navigate(route: "home")

```

```

109             onClick = {
110                 navController.navigate(route: "home")
111             },
112             modifier = Modifier.padding(16.dp)
113         ) { this: RowScope
114             Text(text = "Login")
115         }
116     }
117 }
118 }
119
120 @Composable
121 fun HomeScreen(navController: NavHostController) {
122     Column(
123         modifier = Modifier
124             .fillMaxSize()
125             .background(Color.Gray)
126     ) { this: ColumnScope
127         Text(
128             text = "Welcome to Food4U",
129             modifier = Modifier
130                 .fillMaxWidth()
131                 .background(Color.Red)
132                 .padding(16.dp),
133             style = TextStyle(
134                 fontWeight = FontWeight.Bold

```

```

136         color = Color.White
137     ),
138     textAlign = TextAlign.Center
139 )
140
141 BasicTextField(
142     value = "",
143     onValueChange = {},
144     singleLine = true,
145     textStyle = TextStyle(fontSize = 16.sp),
146     modifier = Modifier
147         .fillMaxWidth()
148         .padding(16.dp)
149         .background(Color.DarkGray)
150         .padding(8.dp)
151 )
152
153
154 val imageLabelPairs = listOf(
155     R.drawable.shawarma to "Al Taza",
156     R.drawable.pizza to "Pizza Hut",
157     R.drawable.biryani to "Paradise"
158 )
159
160 imageLabelPairs.forEach { (imageUrl, label) ->
161     Row(

```

```

163         modifier = Modifier
164             .padding(16.dp)
165     ) { this: RowScope
166         Image(
167             painter = painterResource(id = imageUrl),
168             contentDescription = null,
169             modifier = Modifier
170                 .size(100.dp)
171                 .clickable { navController.navigate(route = "details/$imageUrl") }
172         )
173
174         Spacer(modifier = Modifier.width(16.dp))
175
176         Text(
177             text = label,
178             style = TextStyle(
179                 fontWeight = FontWeight.Bold,
180                 fontSize = 16.sp,
181                 color = Color.Black
182             )
183         )
184     }
185 }
186 }
187 }

```

```

187 }
188 @Composable
189 fun ScreenOne(navController: NavHostController) {
190     val imageId = navController.currentBackStackEntry?.arguments?.getString(key: "imageId")
191     Column(
192         modifier = Modifier
193             .fillMaxSize()
194             .background(Color.White)
195     ) { this: ColumnScope
196
197         Box(
198             modifier = Modifier
199                 .fillMaxWidth()
200                 .padding(16.dp)
201         ) { this: BoxScope
202             Text(
203                 text = "Image ID: $imageId",
204                 modifier = Modifier
205                     .align(Alignment.TopStart),
206                 color = Color.Red
207             )
208         }
209
210         Box(
211             modifier = Modifier
212                 .fillMaxWidth()

```

```

211             modifier = Modifier
212                 .fillMaxWidth()
213                 .height(100.dp)
214                 .background(Color.Red)
215         ) { this: BoxScope
216             Text(
217                 text = "Al taza",
218                 modifier = Modifier
219                     .align(Alignment.Center)
220                     .padding(16.dp),
221                 style = TextStyle(
222                     fontWeight = FontWeight.Bold,
223                     fontSize = 18.sp,
224                     color = Color.White
225                 ),
226             )
227         }
228
229         Box(
230             modifier = Modifier
231                 .fillMaxWidth()
232                 .weight(1f)
233                 .padding(16.dp)
234         ) { this: BoxScope
235             Image(

```

```

238         contentDescription = null,
239         modifier = Modifier
240             .fillMaxSize()
241             .align(Alignment.Center)
242             .size(150.dp)
243     )
244 }
245
246 Text(
247     text = "Recommended",
248     modifier = Modifier
249         .fillMaxWidth()
250         .padding(16.dp),
251     style = TextStyle(
252         fontWeight = FontWeight.Bold,
253         fontSize = 18.sp,
254         color = Color.Black
255     ),
256     textAlign = TextAlign.Center
257 )
258
259
260 Image(
261     painter = painterResource(id = R.drawable.special),
262     contentDescription = null,
263     modifier = Modifier

```

```

259
260 Image(
261     painter = painterResource(id = R.drawable.special),
262     contentDescription = null,
263     modifier = Modifier
264         .fillMaxWidth()
265         .height(150.dp)
266 )
267
268 Image(
269     painter = painterResource(id = R.drawable.speciall),
270     contentDescription = null,
271     modifier = Modifier
272         .fillMaxWidth()
273         .height(150.dp)
274 )
275
276 Image(
277     painter = painterResource(id = R.drawable.plate),
278     contentDescription = null,
279     modifier = Modifier
280         .fillMaxWidth()
281         .height(150.dp)
282 )
283 }

```

```

291 @Composable
292 fun ScreenTwo(navController: NavHostController) {
293     val imageId = navController.currentBackStackEntry?.arguments?.getString(key: "imageId")
294     Column(
295         modifier = Modifier
296             .fillMaxSize()
297             .background(Color.White)
298     ) { this: ColumnScope
299
300         Box(
301             modifier = Modifier
302                 .fillMaxWidth()
303                 .padding(16.dp)
304         ) { this: BoxScope
305             Text(
306                 text = "Image ID: $imageId",
307                 modifier = Modifier
308                     .align(Alignment.TopStart),
309                 color = Color.Red
310             )
311         }
312
313         Box(
314             modifier = Modifier

```

```

313         Box(
314             modifier = Modifier
315                 .fillMaxWidth()
316                 .height(100.dp)
317                 .background(Color.Red)
318         ) { this: BoxScope
319             Text(
320                 text = "Pizza Hut",
321                 modifier = Modifier
322                     .align(Alignment.Center)
323                     .padding(16.dp),
324                 style = TextStyle(
325                     fontWeight = FontWeight.Bold,
326                     fontSize = 18.sp,
327                     color = Color.White
328                 ),
329             )
330         }
331
332         Box(
333             modifier = Modifier
334                 .fillMaxWidth()
335                 .weight(1f)
336                 .padding(16.dp)
337         ) { this: BoxScope

```

```

340 painter = painterResource(id = R.drawable.pizzah),
341 contentDescription = null,
342 modifier = Modifier
343     .fillMaxSize()
344     .align(Alignment.Center)
345     .size(150.dp)
346 )
347 }
348
349 Text(
350     text = "Recommended",
351     modifier = Modifier
352         .fillMaxWidth()
353         .padding(16.dp),
354     style = TextStyle(
355         fontWeight = FontWeight.Bold,
356         fontSize = 18.sp,
357         color = Color.Black
358     ),
359     textAlign = TextAlign.Center
360 )
361
362
363 Image(
364     painter = painterResource(id = R.drawable.chicken),
365     contentDescription = null,

```

```

364 painter = painterResource(id = R.drawable.chicken),
365 contentDescription = null,
366 modifier = Modifier
367     .fillMaxWidth()
368     .height(150.dp)
369 )
370
371 Image(
372     painter = painterResource(id = R.drawable.corn),
373     contentDescription = null,
374     modifier = Modifier
375         .fillMaxWidth()
376         .height(150.dp)
377 )
378
379 Image(
380     painter = painterResource(id = R.drawable.triple),
381     contentDescription = null,
382     modifier = Modifier
383         .fillMaxWidth()
384         .height(150.dp)
385 )
386 }
387 }
388
389

```

Component: Android Emulator  
Update...

```

388
389
390 @Composable
391 fun ScreenThree(navController: NavHostController) {
392     val imageId = navController.currentBackStackEntry?.arguments?.getString( key: "imageId")
393     Column(
394         modifier = Modifier
395             .fillMaxSize()
396             .background(Color.White)
397     ) { this: ColumnScope
398
399         Box(
400             modifier = Modifier
401                 .fillMaxWidth()
402                 .padding(16.dp)
403         ) { this: BoxScope
404             Text(
405                 text = "Image ID: $imageId",
406                 modifier = Modifier
407                     .align(Alignment.TopStart),
408                 color = Color.Red
409             )
410         }
411
412         Box(
413             modifier = Modifier

```

```

412         Box(
413             modifier = Modifier
414                 .fillMaxWidth()
415                 .height(100.dp)
416                 .background(Color.Red)
417         ) { this: BoxScope
418             Text(
419                 text = "Paradise Biryani",
420                 modifier = Modifier
421                     .align(Alignment.Center)
422                     .padding(16.dp),
423                 style = TextStyle(
424                     fontWeight = FontWeight.Bold,
425                     fontSize = 18.sp,
426                     color = Color.White
427                 ),
428             )
429         }
430
431
432         Box(
433             modifier = Modifier
434                 .fillMaxWidth()
435                 .weight(1f)
436                 .padding(16.dp)
437         ) { this: BoxScope

```



```

436         .padding(16.dp)
437     ) { this: BoxScope
438         Image(
439             painter = painterResource(id = R.drawable.paradiselogo),
440             contentDescription = null,
441             modifier = Modifier
442                 .fillMaxSize()
443                 .align(Alignment.Center)
444                 .size(150.dp)
445         )
446     }
447
448     Text(
449         text = "Recommended",
450         modifier = Modifier
451             .fillMaxWidth()
452             .padding(16.dp),
453         style = TextStyle(
454             fontWeight = FontWeight.Bold,
455             fontSize = 18.sp,
456             color = Color.Black
457         ),
458         textAlign = TextAlign.Center
459     )
460

```

```

461
462     Image(
463         painter = painterResource(id = R.drawable.dum),
464         contentDescription = null,
465         modifier = Modifier
466             .fillMaxWidth()
467             .height(150.dp)
468     )
469
470     Image(
471         painter = painterResource(id = R.drawable.minibiryani),
472         contentDescription = null,
473         modifier = Modifier
474             .fillMaxWidth()
475             .height(150.dp)
476     )
477
478     Image(
479         painter = painterResource(id = R.drawable.veqbiryani),
480         contentDescription = null,
481         modifier = Modifier
482             .fillMaxWidth()
483             .height(150.dp)
484     )
485

```

```

478     Image(
479         painter = painterResource(id = R.drawable.veqbiryani),
480         contentDescription = null,
481         modifier = Modifier
482             .fillMaxWidth()
483             .height(150.dp)
484     )
485 }
486
487
488
489
490 @Composable
491 @Preview
492 fun LoginScreenPreview() {
493     LoginScreen(rememberNavController())
494 }
495
496
497
498

```





8:26



## Al Taza Shawarma



### Recommended



#### Al Taza Special Chicken Jumbo

★★★★★ 4 ratings

#500

Search Your City for the Jumbo Menu! Enjoy the Al Taza Shawarma... [read more](#)



ADD

add to cart



#### Al Taza Special Chicken

★★★★★ 59 ratings

#215

Chef's Recommendation, Lebanese flavored chicken, shawarma flavored with... [read more](#)



ADD

add to cart



#### Plate Shawarma Chicken

★★★★★ 5 ratings

#505

A Grand variety of Lebanese flavored Chicken slices, fresh lettuce... [read more](#)



ADD

add to cart

8:41



## Pizza Hut



### Recommended



Available

#### Margherita Pizza

★★★★☆ 145 ratings

#169

Pizza topped with our herb-infused signature pom sauce and 100% mozzarella. [read more](#)



ADD



Available

#### Corn & Cheese

★★★★☆ 24 ratings

#219

A combination of juicy Sweet Corn & 100% mozzarella cheese with Doritos. [read more](#)



ADD

customizable



Available

#### Triple Chicken Feast

#409

Spicy Sizzizen Chicken, Marinated, Breadcrumbs, Harvest Chicken, Juicy Chik. [read more](#)



ADD

customizable

8:50



## Paradise Biryani



### Recommended



#### Chicken Dum Biryani Family Pack (Serves 3-4)

★★★★☆ 240 ratings

₹799

The Family Pack comes with the finest of long grain Basmati Rice with... [read more](#)



ADD



#### 2 Mini Chicken Dum Biryani

★★★★☆ 124 ratings

₹549

2 small portions of biryani rice size of 450 grams each accompanied with... [read more](#)



ADD



#### Nirami Mutton Dum Biryani (Serves 1-2)

★★★★☆ 16 ratings

₹429

Fresh & Cooked Mutton Dum Biryani made with the finest of long grain Basmati... [read more](#)



ADD

[View all >](#)

