**NEWS APP**

PROJECT DEVELOPMENT PHASE

|  |  |
| --- | --- |
| **Date** | **6th November 2023** |
| **Team ID** |  |
| **Project Name** | **An Android Application for keeping up with latest headlines** |
| **Maximum Marks** | **15 marks** |

CategoryTabRow.kt

package com.example.newsroom.presentation.component  
  
import androidx.compose.foundation.ExperimentalFoundationApi  
import androidx.compose.foundation.layout.padding  
import androidx.compose.foundation.pager.PagerState  
import androidx.compose.material3.MaterialTheme  
import androidx.compose.material3.ScrollableTabRow  
import androidx.compose.material3.Tab  
import androidx.compose.material3.Text  
import androidx.compose.runtime.Composable  
import androidx.compose.ui.Modifier  
import androidx.compose.ui.unit.dp  
  
@OptIn(ExperimentalFoundationApi::class)  
@Composable  
fun CategoryTabRow(  
 pagerState: PagerState,  
 categories: List<String>,  
 onTabSelected: (Int) -> Unit  
) {  
 ScrollableTabRow(  
 selectedTabIndex = pagerState.currentPage,  
 edgePadding = 0.*dp*,  
 containerColor = MaterialTheme.colorScheme.primaryContainer,  
 contentColor = MaterialTheme.colorScheme.onPrimaryContainer,  
 ) **{** categories.*forEachIndexed* **{** index, category **->** Tab(  
 selected = pagerState.currentPage == index,  
 onClick = **{** onTabSelected(index) **}**,  
 content = **{** Text(  
 text = category,  
 modifier = Modifier.*padding*(vertical = 8.*dp*, horizontal = 2.*dp*)  
 )  
 **}** )  
 **}  
 }**}

Color.kt

package com.example.newsroom.presentation.theme  
  
import androidx.compose.ui.graphics.Color  
import androidx.compose.ui.graphics.Color.Companion.*White*val *RoyalBlue* = *Color*(0xFFFFFF)

Theme.kt

package com.example.newsroom.presentation.theme  
  
import android.app.Activity  
import android.os.Build  
import androidx.compose.foundation.isSystemInDarkTheme  
import androidx.compose.material3.MaterialTheme  
import androidx.compose.material3.darkColorScheme  
import androidx.compose.material3.dynamicDarkColorScheme  
import androidx.compose.material3.dynamicLightColorScheme  
import androidx.compose.material3.lightColorScheme  
import androidx.compose.runtime.Composable  
import androidx.compose.runtime.SideEffect  
import androidx.compose.ui.graphics.Color  
import androidx.compose.ui.graphics.toArgb  
import androidx.compose.ui.platform.*LocalContext*import androidx.compose.ui.platform.*LocalView*import androidx.core.view.WindowCompat  
  
private val *DarkColorScheme* = *darkColorScheme*(  
 primaryContainer = Color.Black,  
 onPrimaryContainer = Color.White  
)  
  
private val *LightColorScheme* = *lightColorScheme*(  
 primaryContainer = *RoyalBlue*,  
 onPrimaryContainer = Color.White  
)  
  
@Composable  
fun NewsroomTheme(  
 darkTheme: Boolean = isSystemInDarkTheme(),  
 // Dynamic color is available on Android 12+  
 dynamicColor: Boolean = true,  
 content: @Composable () -> Unit  
) {  
 val colorScheme = when {  
 dynamicColor && Build.VERSION.*SDK\_INT* >= Build.VERSION\_CODES.*S* -> {  
 val context = *LocalContext*.current  
 if (darkTheme) *dynamicDarkColorScheme*(context) else *dynamicLightColorScheme*(context)  
 }  
  
 darkTheme -> *DarkColorScheme* else -> *LightColorScheme* }  
 val view = *LocalView*.current  
 if (!view.*isInEditMode*) {  
 SideEffect **{** val window = (view.*context* as Activity).*window* window.*statusBarColor* = colorScheme.primaryContainer.*toArgb*()  
 WindowCompat.getInsetsController(window, view).*isAppearanceLightStatusBars* = darkTheme  
 **}** }  
  
 MaterialTheme(  
 colorScheme = colorScheme,  
 typography = *Typography*,  
 content = content  
 )  
}

Type.kt

package com.example.newsroom.presentation.theme  
  
import androidx.compose.material3.Typography  
import androidx.compose.ui.text.TextStyle  
import androidx.compose.ui.text.font.FontFamily  
import androidx.compose.ui.text.font.FontWeight  
import androidx.compose.ui.unit.sp  
  
val *Typography* = Typography(  
 bodyLarge = TextStyle(  
 fontFamily = FontFamily.Default,  
 fontWeight = FontWeight.Normal,  
 fontSize = 16.*sp*,  
 lineHeight = 24.*sp*,  
 letterSpacing = 0.5.*sp* )  
)

ImageHolder.kt

package com.example.newsroom.presentation.component  
  
import androidx.compose.foundation.layout.aspectRatio  
import androidx.compose.foundation.layout.fillMaxWidth  
import androidx.compose.foundation.shape.RoundedCornerShape  
import androidx.compose.runtime.Composable  
import androidx.compose.ui.Modifier  
import androidx.compose.ui.draw.clip  
import androidx.compose.ui.layout.ContentScale  
import androidx.compose.ui.platform.*LocalContext*import androidx.compose.ui.res.painterResource  
import androidx.compose.ui.unit.dp  
import coil.compose.AsyncImage  
import coil.request.ImageRequest  
import com.example.newsroom.R  
import com.example.newsroom.domain.model.Article  
  
@Composable  
fun ImageHolder(  
 imageUrl: String?,  
 modifier: Modifier = Modifier  
) {  
 AsyncImage(  
 model = ImageRequest  
 .Builder(*LocalContext*.current)  
 .data(imageUrl)  
 .crossfade(true)  
 .build(),  
 contentDescription = "Image",  
 contentScale = ContentScale.Crop,  
 modifier = modifier  
 .*clip*(*RoundedCornerShape*(4.*dp*))  
 .*fillMaxWidth*()  
 .*aspectRatio*(16 / 9f),  
 placeholder = painterResource(R.drawable.*placeholder\_loading*),  
 error = painterResource(R.drawable.*placeholder\_news*)  
 )  
}

NewsArticleCard.kt

package com.example.newsroom.presentation.component  
  
import androidx.compose.foundation.background  
import androidx.compose.foundation.clickable  
import androidx.compose.foundation.layout.Arrangement  
import androidx.compose.foundation.layout.Column  
import androidx.compose.foundation.layout.Row  
import androidx.compose.foundation.layout.Spacer  
import androidx.compose.foundation.layout.aspectRatio  
import androidx.compose.foundation.layout.fillMaxWidth  
import androidx.compose.foundation.layout.height  
import androidx.compose.foundation.layout.padding  
import androidx.compose.foundation.shape.RoundedCornerShape  
import androidx.compose.material3.Card  
import androidx.compose.material3.MaterialTheme  
import androidx.compose.material3.Text  
import androidx.compose.runtime.Composable  
import androidx.compose.ui.Modifier  
import androidx.compose.ui.draw.clip  
import androidx.compose.ui.graphics.Color  
import androidx.compose.ui.layout.ContentScale  
import androidx.compose.ui.platform.*LocalContext*import androidx.compose.ui.res.painterResource  
import androidx.compose.ui.text.style.TextOverflow  
import androidx.compose.ui.unit.dp  
import coil.compose.AsyncImage  
import coil.request.ImageRequest  
import com.example.newsroom.R  
import com.example.newsroom.domain.model.Article  
import com.example.newsroom.util.dateFormatter  
  
@Composable  
fun NewsArticleCard(  
 modifier: Modifier = Modifier.*background*(color = Color.White),  
 article: Article,  
 onCardClicked: (Article) -> Unit  
) {  
 val date = *dateFormatter*(article.publishedAt)  
 Card(  
 modifier = modifier.*clickable* **{** onCardClicked(article) **}** .*background*(color = Color.White)  
 ) **{** Column() **{** ImageHolder(imageUrl = article.urlToImage)  
 Spacer(modifier = Modifier.*height*(8.*dp*))  
 Column(modifier = Modifier.*padding*(12.*dp*)) **{** Text(  
 text = article.title,  
 style = MaterialTheme.typography.titleMedium,  
 maxLines = 1,  
 overflow = TextOverflow.Ellipsis  
 )  
 **}** Row(  
 modifier = Modifier  
 .*fillMaxWidth*()  
 .*padding*(12.*dp*),  
 horizontalArrangement = Arrangement.SpaceBetween  
 ) **{** Text(  
 text = article.source.name ?: "",  
 style = MaterialTheme.typography.bodySmall  
 )  
 Text(  
 text = date,  
 style = MaterialTheme.typography.bodySmall  
 )  
 **}  
 }  
 }**}

ArticleScreen.kt

package com.example.newsroom.presentation.article\_screen  
  
import android.webkit.WebView  
import android.webkit.WebViewClient  
import androidx.compose.foundation.layout.Box  
import androidx.compose.foundation.layout.fillMaxSize  
import androidx.compose.foundation.layout.padding  
import androidx.compose.material.icons.Icons  
import androidx.compose.material.icons.filled.*ArrowBack*import androidx.compose.material.icons.filled.*Search*import androidx.compose.material3.CircularProgressIndicator  
import androidx.compose.material3.ExperimentalMaterial3Api  
import androidx.compose.material3.Icon  
import androidx.compose.material3.IconButton  
import androidx.compose.material3.MaterialTheme  
import androidx.compose.material3.Scaffold  
import androidx.compose.material3.Text  
import androidx.compose.material3.TopAppBar  
import androidx.compose.material3.TopAppBarDefaults  
import androidx.compose.runtime.Composable  
import androidx.compose.runtime.mutableStateOf  
import androidx.compose.runtime.remember  
import androidx.compose.runtime.setValue  
import androidx.compose.runtime.getValue  
import androidx.compose.ui.Alignment  
import androidx.compose.ui.Modifier  
import androidx.compose.ui.platform.*LocalContext*import androidx.compose.ui.text.font.FontWeight  
import androidx.compose.ui.viewinterop.AndroidView  
  
@OptIn(ExperimentalMaterial3Api::class)  
@Composable  
fun ArticleScreen(  
 url: String?,  
 onBackPressed: () -> Unit  
) {  
 val context = *LocalContext*.current  
 var isLoading by remember **{** *mutableStateOf*(true) **}** Scaffold(  
 modifier = Modifier.*fillMaxSize*(),  
 topBar = **{** TopAppBar(  
 title = **{** Text(text = "Article", fontWeight = FontWeight.Bold) **}**,  
 navigationIcon = **{** IconButton(onClick = onBackPressed) **{** Icon(imageVector = Icons.Default.*ArrowBack*, contentDescription = "Back")  
 **}  
 }**,  
 colors = TopAppBarDefaults.topAppBarColors(  
 containerColor = MaterialTheme.colorScheme.primaryContainer,  
 navigationIconContentColor = MaterialTheme.colorScheme.onPrimaryContainer,  
 titleContentColor = MaterialTheme.colorScheme.onPrimaryContainer  
 )  
 )  
 **}** ) **{** padding **->** Box(  
 modifier = Modifier  
 .*fillMaxSize*()  
 .*padding*(padding),  
 contentAlignment = Alignment.Center  
 ) **{** AndroidView(factory = **{** WebView(context).*apply* **{** *webViewClient* = object : WebViewClient() {  
 override fun onPageFinished(view: WebView?, url: String?) {  
 isLoading = false  
 }  
 }  
 loadUrl(url ?: "")  
 **}  
 }**)  
 if (isLoading && url != null) {  
 CircularProgressIndicator()  
 }  
 **}  
 }**}

NewsScreenTopBar.kt

package com.example.newsroom.presentation.component  
  
import androidx.compose.material.icons.Icons  
import androidx.compose.material.icons.filled.*Search*import androidx.compose.material3.ExperimentalMaterial3Api  
import androidx.compose.material3.Icon  
import androidx.compose.material3.IconButton  
import androidx.compose.material3.MaterialTheme  
import androidx.compose.material3.Text  
import androidx.compose.material3.TopAppBar  
import androidx.compose.material3.TopAppBarDefaults  
import androidx.compose.material3.TopAppBarScrollBehavior  
import androidx.compose.runtime.Composable  
import androidx.compose.ui.text.font.FontWeight  
  
@OptIn(ExperimentalMaterial3Api::class)  
@Composable  
fun NewsScreenTopBar(  
 scrollBehavior: TopAppBarScrollBehavior,  
 onSearchIconClicked: () -> Unit  
) {  
 TopAppBar(  
 scrollBehavior = scrollBehavior,  
 title = **{** Text(text = "NEWSAPP", fontWeight = FontWeight.ExtraBold)**}**,  
 actions = **{** IconButton(onClick = onSearchIconClicked) **{** Icon(imageVector = Icons.Default.*Search*, contentDescription = "Search")  
 **}  
 }**,  
 colors = TopAppBarDefaults.topAppBarColors(  
 containerColor = MaterialTheme.colorScheme.primaryContainer,  
 titleContentColor = MaterialTheme.colorScheme.onPrimaryContainer,  
 actionIconContentColor = MaterialTheme.colorScheme.onPrimaryContainer  
 )  
 )  
}

BottomSheetContent.kt

package com.example.newsroom.presentation.component  
  
import androidx.compose.foundation.layout.Arrangement  
import androidx.compose.foundation.layout.Column  
import androidx.compose.foundation.layout.Row  
import androidx.compose.foundation.layout.Spacer  
import androidx.compose.foundation.layout.fillMaxWidth  
import androidx.compose.foundation.layout.height  
import androidx.compose.foundation.layout.padding  
import androidx.compose.material3.Button  
import androidx.compose.material3.MaterialTheme  
import androidx.compose.material3.Surface  
import androidx.compose.material3.Text  
import androidx.compose.runtime.Composable  
import androidx.compose.ui.Alignment  
import androidx.compose.ui.Modifier  
import androidx.compose.ui.text.font.FontWeight  
import androidx.compose.ui.unit.dp  
import com.example.newsroom.domain.model.Article  
  
@Composable  
fun BottomSheetContent(  
 article: Article,  
 onReadFullStoryButtonClicked: () -> Unit  
) {  
 Surface(  
 modifier = Modifier.*padding*(16.*dp*)  
 ) **{** Column(horizontalAlignment = Alignment.CenterHorizontally) **{** Text(  
 text = article.title,  
 style = MaterialTheme.typography.titleMedium  
 )  
 Spacer(modifier = Modifier.*height*(8.*dp*))  
 Text(  
 text = article.description ?: "",  
 style = MaterialTheme.typography.bodyMedium  
 )  
 Spacer(modifier = Modifier.*height*(8.*dp*))  
 ImageHolder(imageUrl = article.urlToImage)  
 Spacer(modifier = Modifier.*height*(8.*dp*))  
 Text(  
 text = article.content ?: "",  
 style = MaterialTheme.typography.bodyMedium  
 )  
 Spacer(modifier = Modifier.*height*(8.*dp*))  
 Row(  
 modifier = Modifier.*fillMaxWidth*(),  
 horizontalArrangement = Arrangement.SpaceBetween  
 ) **{** Text(  
 text = article.author ?: "",  
 style = MaterialTheme.typography.bodySmall,  
 fontWeight = FontWeight.Bold  
 )  
 Text(  
 text = article.source.name ?: "",  
 style = MaterialTheme.typography.bodySmall,  
 fontWeight = FontWeight.Bold  
 )  
 **}** Spacer(modifier = Modifier.*height*(8.*dp*))  
 Button(  
 modifier = Modifier.*fillMaxWidth*(),  
 onClick = onReadFullStoryButtonClicked  
 ) **{** Text(text = "Read Full Story")  
 **}  
 }  
 }**}

RetryContent.kt

package com.example.newsroom.presentation.component  
  
import androidx.compose.foundation.layout.Column  
import androidx.compose.foundation.layout.Spacer  
import androidx.compose.foundation.layout.height  
import androidx.compose.material3.Button  
import androidx.compose.material3.Text  
import androidx.compose.runtime.Composable  
import androidx.compose.ui.Alignment  
import androidx.compose.ui.Alignment.Companion.*CenterHorizontally*import androidx.compose.ui.Modifier  
import androidx.compose.ui.graphics.Color  
import androidx.compose.ui.unit.dp  
import androidx.compose.ui.unit.sp  
  
@Composable  
fun RetryContent(  
 error: String,  
 onRetry: () -> Unit,  
 modifier: Modifier = Modifier  
) {  
 Column(  
 modifier = modifier  
 ) **{** Text(text = error, color = Color.Red, fontSize = 18.*sp*)  
 Spacer(modifier = Modifier.*height*(8.*dp*))  
 Button(  
 onClick = onRetry,  
 modifier = Modifier.*align*(*CenterHorizontally*)  
 ) **{** Text(text = "Retry")  
 **}  
 }**}

NavGraphSetup.kt

package com.example.newsroom.util  
  
import androidx.compose.runtime.Composable  
import androidx.hilt.navigation.compose.hiltViewModel  
import androidx.navigation.NavHostController  
import androidx.navigation.NavType  
import androidx.navigation.compose.NavHost  
import androidx.navigation.compose.composable  
import androidx.navigation.navArgument  
import com.example.newsroom.presentation.article\_screen.ArticleScreen  
import com.example.newsroom.presentation.news\_screen.NewsScreen  
import com.example.newsroom.presentation.news\_screen.NewsScreenViewModel  
  
@Composable  
fun NavGraphSetup(  
 navController: NavHostController  
) {  
 val argKey = "web\_url"  
  
 NavHost(  
 navController = navController,  
 startDestination = "news\_screen"  
 ) **{** *composable*(route = "news\_screen") **{** val viewModel: NewsScreenViewModel = hiltViewModel()  
 NewsScreen(  
 state = viewModel.state,  
 onEvent = viewModel::onEvent,  
 onReadFullStoryButtonClick = **{** url **->** navController.navigate("article\_screen?$argKey=$url")  
 **}** )  
 **}** *composable*(  
 route = "article\_screen?$argKey={$argKey}",  
 arguments = *listOf*(*navArgument*(name = argKey) **{** type = NavType.StringType  
 **}**)  
 ) **{** backStackEntry **->** ArticleScreen(  
 url = backStackEntry.arguments?.getString(argKey),  
 onBackPressed = **{** navController.navigateUp() **}** )  
 **}  
 }**}

MainActivity.kt

package com.example.newsroom  
  
import android.os.Bundle  
import androidx.activity.ComponentActivity  
import androidx.activity.compose.setContent  
import androidx.hilt.navigation.compose.hiltViewModel  
import androidx.navigation.compose.rememberNavController  
import com.example.newsroom.presentation.news\_screen.NewsScreen  
import com.example.newsroom.presentation.news\_screen.NewsScreenViewModel  
import com.example.newsroom.presentation.theme.NewsroomTheme  
import com.example.newsroom.util.NavGraphSetup  
import dagger.hilt.android.AndroidEntryPoint  
  
@AndroidEntryPoint  
class MainActivity : ComponentActivity() {  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 *setContent* **{** NewsroomTheme **{** val navController = rememberNavController()  
 NavGraphSetup(navController = navController)  
 **}  
 }** }  
}

MyApplication.kt

package com.example.newsroom  
  
import android.app.Application  
import dagger.hilt.android.HiltAndroidApp  
  
@HiltAndroidApp  
class MyApplication: Application()

**IMAGES OF FINAL APP –**

