

A Project Report on
An Android Application for Keeping Up with the Latest Headlines

BACHELOER OF COMPUTER APPLICATION

Saiva Bhanu Kshatriya Arts & Science College,

(Affiliated to Madurai Kamaraj University)

Aruppukottai-626101

Saiva Bhanu Kshatriya Arts and Science College

(Affiliated to Madurai Kamaraj University)

Aruppukottai-626101

Submitted

BY

M. DIVYALAKSHMI (Reg.No:COS23651)

P. JAYAPRIYA (Reg.No:COS23652)

S. KEERTHANA (Reg.No:COS23653)

J. KUMUTHA (Reg.No:COS23654)

INTRODUCTION:

Android provides simple application structure and requires Java and Mark-up languages knowledge to work with. Such as, an discrete movement delivers a solitary screen for a user interface and a service whole completes work in the contextual [1]. We can work on different module separately and can combine at the end, we can also add future modules easily afterwards. API (Application Programming Interface) which is an intermediate interface between different applications. It provides automation, immediacy, adaption and personalization. News API provides us the source of news articles from many different sources at one place and updates it. To expand the sources old fashioned Admin panel can be used where writers will fill the gap of API. In 2014, a design language has been created by google named Material Design which is based on “cards” uses grind based layouts, responsive animation, padding and depth effects like shadow to create an responsive, attractive and easy user interface. With the use of different libraries and material design it is possible to use attractive UI.

1.1 OVERVIEW

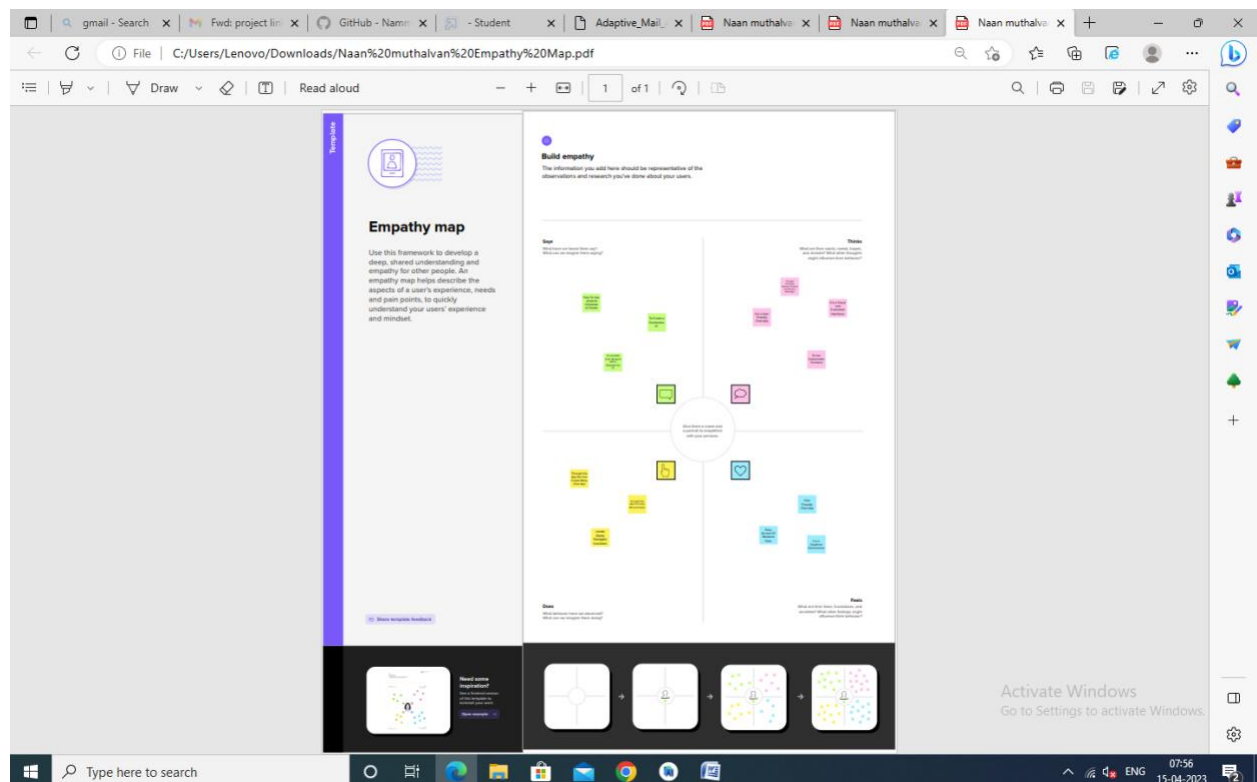
The News and Weather app allows users to browse through news headlines for different categories such as Top Stories, General, Business, Technology, Entertainment, and so on. Furthermore, the app lets users glance through the news and weather conditions. The app makes use of Integration and Orchestration services of Quantum Fabric to fetch data from third-party APIs, and then populates the data to the front end of the app.

1.2 PURPOSE

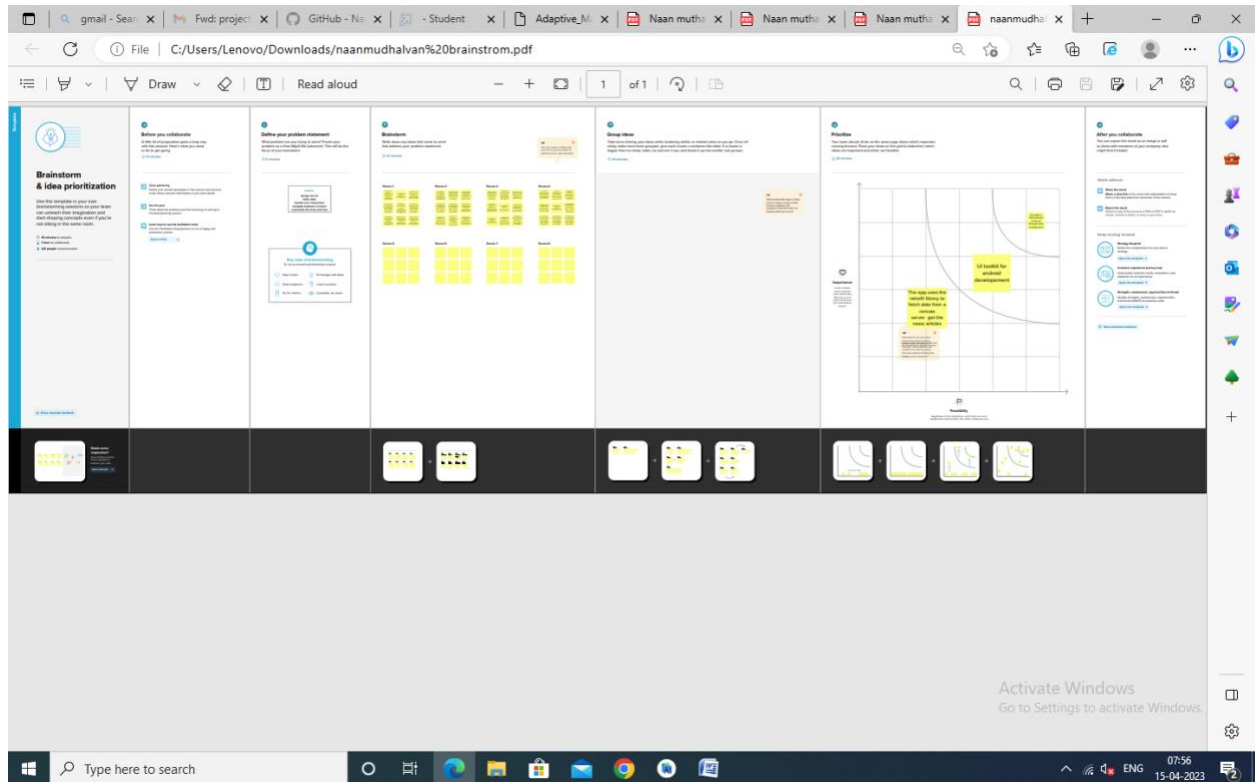
News apps offer a great reading experience and simultaneously display the ads helpfully so they don't annoy users and distract them from the focus on the content they are reading, yet showing the ads simultaneously.

2. Problem Definition & Design Thinking

2.1 Empathy Map



2.2 Ideation & Brainstorming Map2



3. RESULT

Sign Up



 username

 password

 email

Register

Have an account? [Log in](#)



Login

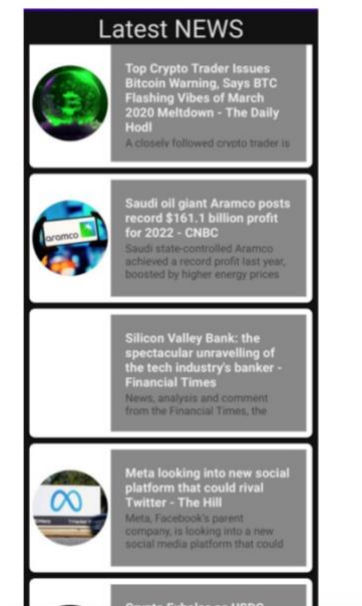
 username

 password

Log in

[Sign up](#)

[Forgot password ?](#)



4. ADVANTAGES & DISADVANTAGES

Advantages:-

1. Revenue production by a new channel

A digital magazine boasts different methods to generate revenue through paid subscriptions, paid articles, and ads. The follower and reader base is straight equivalent to the revenue.

Nevertheless, in terms of news apps in this smartphone era, you should assess the scenario regarding apps' conversion into a new source of profit. News app users reach out to such applications to involve in a captivating and good content associated with their choices.

You must note that readers are not attracted by banner ads and they try to avoid them completely. Hence, you may avoid such pop-up ads because users just concentrate on the news content instead of ads.

News apps offer a great reading experience and simultaneously display the ads helpfully so they don't annoy users and distract them from the focus on the content they are reading, yet showing the ads simultaneously.

2. People's preference for mobile apps over websites

Since almost like mobile apps more than mobile browsers for online shopping, the same reason goes for news apps and online magazines as well.

As per a survey done by Compuware, users choose mobile applications because they are quicker, more feasible and simpler to browse and provide a better user experience also. So the same conveniences they can receive from a news app.

Disadvantages:-

- Require data/wifi to get online
- Companies not making as much money due to free reading for audiences
- News spreads quicker online - people find out news before they should
- Lose money - can't get people to pay for digital
- Older audiences may not access digital platforms
- Costly to maintain
- Errors stay online FOREVER
- Response to pace = errors in reporting
- Fake News!

5. APPLICATIONS

news delivery

Today, news delivery can be done via SMS, by specialized applications, or using mobile versions of media websites. According to a recent market study across six countries (France, Germany, Italy, Spain, UK, and US), 16.9% of consumers access news and information via mobile devices, either via browser, downloaded application, or SMS alerts.^[1]

The demand for mobile news delivery is growing quickly, with 107 percent growth in daily access to mobile news in the last year alone.^[2] For example, the New York Times mobile site registered 19 million views in May 2008, compared to 500,000 in January 2007.^[3]

July 18, 2011, Time Warner announced that news coverage from CNN and Headline News will be streamed live over the Internet and available for people to view on their laptops, smartphones, or tablets if they subscribe to certain paid TV services.^[4]

From 2014 many media companies launched their native mobile application including Newto engage global users by delivering quick and short news of their choice.

Mobile news creation

Mobile news also has the potential to place the power of breaking news reporting in the hands of small communities and facilitate a much better exchange of information among users due to the ease of usage of mobile phones compared with conventional media such as radio, TV or newspapers, though issues of quality, journalistic standards and professionalism are of concern to some critics.^[5]^[citation needed]

Mobile telephony and full featured mobile devices also facilitate activism and citizen journalism. In addition to individual efforts, major media outlets like CNN, Reuters, and Yahoo are attempting to harness the power of citizen journalists.

The creation of mobile news was fuelled first by the popularity of receiving text alerts, and then hugely accelerated when mobile companies embraced social media, making content creation easy and accessible.

6.CONCLUSION

Over the centuries, newspaper institutions have become hegemonic institutions, and the multimodal design of newspapers is an enactment of the social (re-)production of power relations: the power of newspapers lies in their discourse, and this discourse is fundamentally multimodal. In the early 21st century, newspapers face discursive competition as never before. In the commercial and ideological 'attention economy', online newspapers must compete not only with a much greater number of newspapers in a globalised market, but also with other fora accessed through other media, both traditional (e.g. television, radio) and emerging (e.g. mobile phones and other portable devices). The numbers of social networking and other user-generated-content sites, on which anyone with an internet connection can construct and mass-distribute information and opinion quickly and cheaply, are growing, and the scale of this growth brings innovation, failure, and success, or in short, evolution. The meaning potential of humanity is expanding rapidly. And with this expansion comes a potential realignment of power. Some see these changes as a great democratising movement, and power relations are undoubtedly at stake in a way that 494 Chapter Nine: Conclusion they were not before the world wide web. But hegemonic institutions such as online newspapers are not sitting idly by. Indeed, they are scrambling to stake their claim on the web in order to maintain, or extend their hegemony. This is reflected in the rapid evolution of new forms of news discourse (Chapters 7 and 8). Newspapers have always depended for their survival on being at the edge of information construction and distribution, and the institutionalised power and adaptability of newspapers remain as strengths. The speed of the evolution of the macro-genre of the online newspaper is remarkable. New genres such as newsbites, newsbits, 'videobites', slide shows, blogs, and image galleries are commonplace. Not only this, but story pages often allow readers to link these pages to the very fora that may collectively threaten the newspapers' existence (e.g. del.icio.us,

Digg, Facebook, twitter), fora that represent methods of communication that literally did not exist when this research project began. In this way, newspapers become integrated in readers' web-mediated social networks. But whether they do it by means of print newspapers, so-called 'digital editions', online newspapers, or otherwise, newspaper institutions will need to do a better job than the authors who publish on user-generated-content sites, social networking sites, in personal blogs, and in whatever communicative environments evolve in the future. The Bangkok Post, People's Daily, Sydney Morning Herald, and the thousands of newspapers published around the world will need to continue attracting attention. 495

Chapter Nine: Conclusion 496 Newspapers came to the internet with an advantage, and the established, shared position of power they have held is reflected in the consistent semiotic structures by which they construe the world and relate to their readers (Chapters 5 and 6). But the environment has changed, and continues to do so. The 'rivers of gold' of advertising revenue are drying up; the relative monopoly on the ability to massdistribute information is gone; rapid evolution on surprisingly short timescales, such as that documented in this thesis, appears to be the new norm. The construal of ideology is moving from extended argumentation in 'regulated' texts through which readers progress from beg

7.FUTURE SCOPE

The modern smartphone is 7 years old and yet, when it comes to designing mobile applications, we are still barely scratching the surface. Today we'll see how harnessing technology already embedded in a phone can unleash great potential.

A mobile news app has simple goals: Capture and retain reader attention, and repeat the process, several times a day. Pretty straightforward. But not that simple in the real world. For a news provider, the smartphone screen is the most challenging environment ever seen. There, chances are that a legacy media or a pure-player will find itself in direct competition, not only with the usual players in its field, but also with Facebook, Snapchat, Instagram and scores of gaming applications. Distraction is just one icon away; any weakness in functional or graphic design can be lethal.

If mobile is to become the dominant vector for news, retaining readers will be much more challenging than it is on a PC or tablet (though the latter tends to engage readers 10x or sometimes 20x more). A news app needs to be steered with precision. Today's digital marketing tools allow publishers to select multiple parameters monitoring the use of a application: They can measure how long the app is used, when, for how long, why and where people tend to drop it, what kind of news they like, if they hit a paywall and give up, and why they do so, etc. Similarly, when an app remains unopened for too long, smart tools can pinpoint the user and remind her of the product's benefits. These tools are as good as the people who (a) set the parameters, (b) monitor them on a daily basis, and (c) take appropriate

action such as launching a broadside of super-targeted emails. But these are incremental measures, they don't breed exponential growth in viewership (and revenue).

8. APPENDIX

Source code

```
packagecom.example.newsapplication
```

```
importandroid.content.Context
```

```
importandroid.content.Intent
```

```
importandroid.content.Intent.FLAG_ACTIVITY_NEW_TASK
```

```
importandroid.os.Bundle
```

```
importandroid.util.Log
```

```
importandroid.widget.TextView
```

```
importandroidx.activity.ComponentActivity
```

```
importandroidx.activity.compose.setContent
```

```
importandroidx.activity.viewModels
```

```
importandroidx.compose.foundation.Image
```

```
importandroidx.compose.foundation.background
```

import androidx.compose.foundation.clickable

import androidx.compose.foundation.layout.*

import androidx.compose.foundation.lazy.LazyColumn

import androidx.compose.foundation.lazy.itemsIndexed

import androidx.compose.foundation.selection.selectable

import androidx.compose.foundation.shape.RoundedCornerShape

import androidx.compose.material.Card

import androidx.compose.material.MaterialTheme

import androidx.compose.material.Surface

import androidx.compose.material.Text

import androidx.compose.runtime.*

import androidx.compose.ui.Modifier

import androidx.compose.ui.graphics.Color

import androidx.compose.ui.text.font.FontWeight

import androidx.compose.ui.text.style.TextAlign

import androidx.compose.ui.unit.dp

import androidx.compose.ui.unit.sp

import androidx.compose.ui.viewinterop.AndroidView

import androidx.core.text.HtmlCompat

import coil.compose.rememberImagePainter

import coil.size.Scale

import coil.transform.CircleCropTransformation

```
import com.example.example.Articles
```

```
import com.example.newsapplication.ui.theme.NewsApplicationTheme
```

```
import com.example.newsapplication.MainViewModel
```

```
class MainPage : ComponentActivity() {
```

```
    val mainViewModel by viewModels<MainViewModel>()
```

```
    override fun onCreate(savedInstanceState: Bundle?) {
```

```
        super.onCreate(savedInstanceState)
```

```
        setContent {
```

```
            NewsApplicationTheme {
```

```
                // A surface container using the 'background' color from the theme
```

```
                Surface(color = MaterialTheme.colors.background) {
```

```
                    Column() {
```

```
                        Text(text = "Latest NEWS", fontSize = 32.sp, modifier = Modifier.fillMaxWidth(), textAlign =  
                        TextAlign.Center)
```

```
                        MovieList(applicationContext, movieList = mainViewModel.movieListResponse)
```

```
                        mainViewModel.getMovieList()
```

```
                    }
```

```
                }
```

```
    }  
    }  
    }  
}
```

@Composable

```
funMovieList(context: Context, movieList: List<Articles>) {  
  
    varselectedIndex by remember { mutableStateOf(-1) }  
  
    LazyColumn {  
  
        itemsIndexed(items = movieList) {  
  
            index, item ->  
  
            MovieItem(context, movie = item, index, selectedIndex) { i ->  
  
                selectedIndex = i  
  
            }  
  
        }  
  
    }  
  
}
```

@Composable

```
funMovieItem(context: Context) {
```

```
val movie = Articles(
```

```
    "Coco",
```

```
    "",
```

```
    "articl"
```

```
)
```

```
MovieItem(context, movie = movie, 0, 0) { i ->
```

```
    Log.i("wertytest123abc", "MovieItem: "
```

```
        +i)
```

```
    }
```

```
}
```

```
@Composable
```

```
funMovieItem(context: Context, movie: Articles, index: Int, selectedIndex: Int,
```

```
onClick: (Int) -> Unit)
```

```
{
```

```
    valbackgroundColor = if (index == selectedIndex) MaterialTheme.colors.primary else
```

```
    MaterialTheme.colors.background
```

```
    Card(
```



```

modifier = Modifier

        .padding(8.dp, 4.dp)

        .fillMaxSize()

        .selectable(true, true, null,

onClick = {

Log.i("test123abc", "MovieItem: $index/n$selectedIndex")

        })

        .clickable { onClick(index) }

        .height(180.dp), shape = RoundedCornerShape(8.dp), elevation = 4.dp

    ) {

Surface(color = Color.White) {

Row(

    Modifier

        .padding(4.dp)

        .fillMaxSize()

    )

    {

Image(

    painter = rememberImagePainter(

    data = movie.urlToImage,

```

```
builder = {  
  
    scale(Scale.FILL)  
  
    placeholder(R.drawable.placeholder)  
  
    transformations(CircleCropTransformation())  
  
    }  
  
    ),  
  
    contentDescription = movie.description,  
  
    modifier = Modifier  
  
        .fillMaxHeight()  
  
        .weight(0.3f)  
  
    )
```

```
Column(  
  
    verticalArrangement = Arrangement.Center,  
  
    modifier = Modifier  
  
        .padding(4.dp)  
  
        .fillMaxHeight()  
  
        .weight(0.8f)  
  
        .background(Color.Gray)  
  
        .padding(20.dp)  
  
        .selectable(true, true, null,
```

```
onClick = {

    Log.i("test123abc", "MovieItem: $index/n${movie.description}")

    context.startActivity(

        Intent(context, DisplayNews::class.java)

            .setFlags(Intent.FLAG_ACTIVITY_NEW_TASK)

            .putExtra("desk", movie.description.toString())

            .putExtra("urlToImage", movie.urlToImage)

            .putExtra("title", movie.title)

        )

    })

    ){
```

```
Text(

    text = movie.title.toString(),

    style = MaterialTheme.typography.subtitle1,

    fontWeight = FontWeight.Bold

    )
```

```
HtmlText(html = movie.description.toString())

    }

    }

}
```

```
}
```

```
@Composable
```

```
funHtmlText(html: String, modifier: Modifier = Modifier) {
```

```
    AndroidView(
```

```
        modifier = modifier
```

```
            .fillMaxSize()
```

```
            .size(33.dp),
```

```
        factory = { context ->TextView(context) },
```

```
        update = { it.text = HtmlCompat.fromHtml(html, HtmlCompat.FROM_HTML_MODE_COMPACT) }
```

```
    )
```

```
}
```

```
}
```