

**AN ANDRIOD APPLICATION FOR  
KEEPING UP WITH THE  
LATEST HEADLINES**

HARIHARAN.R

ARAVINDAN.K

SEBADHINESH.J

SHAHIL ABSAR.S

MOHAMED NIYASDEEN.S

## 1. INTRODUCTION

### 1.1 Overview

The app's main feature is displaying a list of news articles, each with a title, image, and brief description. Users can scroll through the list of articles and tap on an article to view more details.

The app uses the Jetpack Compose UI toolkit to build the UI and it uses the coil library to load images. The app fetches data from a remote server using Retrofit library and demonstrates how to use the Jetpack Compose UI toolkit for Android development.

### 1.2 PURPOSE

Android is an operating system for mobile devices, and it does not have a specific purpose for keeping up with the latest headlines. However, there are numerous apps available on the Android platform that can help users stay updated with the latest news and headlines.

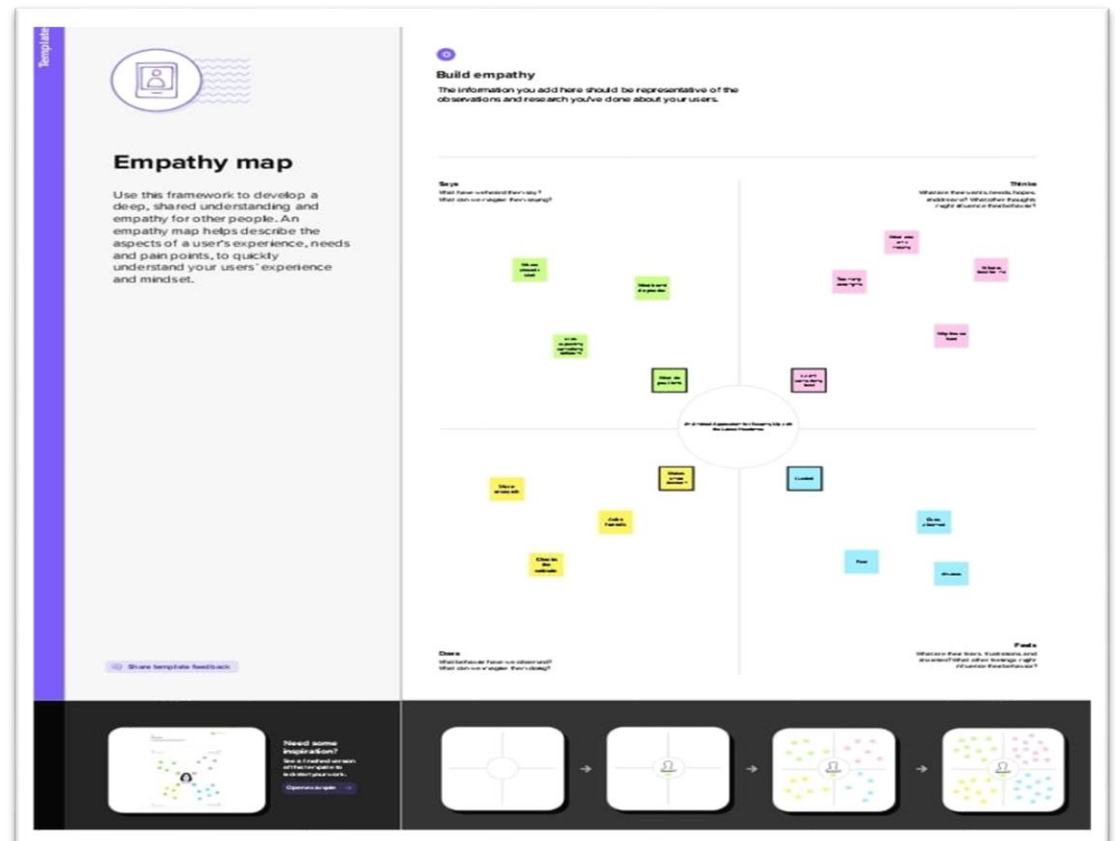
Some of the popular news apps for Android include Google News, CNN, BBC News, The New York Times, and many more. These apps allow users to customize their news feeds based on their interests and preferences, providing them with the latest news and headlines from around the world.

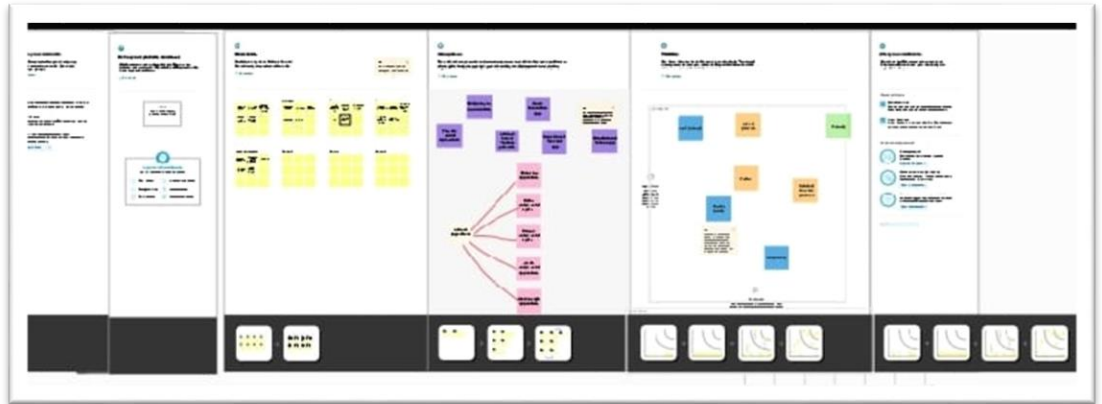
Additionally, Android devices often come with built-in news widgets that can display headlines and updates directly on the home screen, making it easy for users to keep up with the latest news without having to open an app.

Overall, Android provides a platform for users to access a variety of news sources and stay informed about the latest headlines and news stories.


## **2.problem definition&design thinking**

### **2.1 Empathy Map**







### 3.RESULT



Login

 username

 password

[Log in](#)


[Sign up](#) [Forgot password ?](#)

Register Page :

## Sign Up



 username

 password


 email

**Register**


Have an account? [Log in](#)

Main News Headlines Page :

### Latest NEWS




**Top Crypto Trader Issues Bitcoin Warning, Says BTC Flashing Vibes of March 2020 Meltdown - The Daily Hodl**  
A closely followed crypto trader is




**Saudi oil giant Aramco posts record \$161.1 billion profit for 2022 - CNBC**  
Saudi state-controlled Aramco achieved a record profit last year, boosted by higher energy prices

**Silicon Valley Bank: the spectacular unravelling of the tech industry's banker - Financial Times**  
News, analysis and comment from the Financial Times, the



**Meta looking into new social platform that could rival Twitter - The Hill**  
Meta, Facebook's parent company, is looking into a new social media platform that could



**Crypto Exhales as USDC Stablecoin Rebounds**



Display News Page :



#### 4.ADVANDAGES & DISADVANTAGES:

##### *Advantages:*

\*There are several advantages to using an Android device for keeping up with the latest headlines, including:

\*Wide range of news apps: Android devices have access to a wide range of news apps, including popular ones like CNN, BBC, and The New York Times, as well as specialized ones for niche topics. These apps can help you stay up to date with the latest news from around the world.

\*Customization options: Android devices allow you to customize your home screen and widgets, making it easy to see the latest news headlines without even opening an app. You can also customize your news feed to show only the topics that interest you, making it easier to find the news you want.

\*Push notifications: Many news apps on Android devices offer push notifications, which can alert you to breaking news stories as soon as they happen. This means you can stay informed even when you're not actively looking for news.

\*Integration with other apps: Android devices allow for easy integration between different apps, making it easy to share news stories with friends and family or save articles for later reading. This can help you stay connected and informed with those around you.

\*Voice assistance: Android devices offer voice assistants like Google Assistant, which can help you stay up to date with the latest headlines even when your hands are full. You can simply ask your device to read you the news or provide updates on specific topics.

### *Disadvantages:*

While there are many advantages to using an Android device for keeping up with the latest headlines, there are also a few potential disadvantages to consider, including:

Fragmentation: One potential disadvantage of Android is the fragmentation of the platform. There are many different Android devices on the market, each with different hardware specs and software versions.

Security: Android devices are known to have some security vulnerabilities, which can make it easier for hackers to gain access to your personal information

or compromise your device. This can be a concern when using news apps or other apps that require access to your data.

**Ads and pop-ups:** Some news apps on Android devices may have a lot of ads or pop-ups, which can be annoying and distracting. This can make it harder to focus on the news can slow down your device.

**Battery life:** Some news apps on Android devices can be battery-intensive, which can drain your device's battery quickly. This can be frustrating if you're trying to stay up to date with the latest headlines on the go.

**Data usage:** Many news apps on Android devices require a lot of data to download articles and updates. This can be a concern if you have a limited data plan or if you're using your device in an area with poor network coverage.

## **5.APPLICATIONS:**

There are many Android applications that you can use to keep up with the latest headlines, news and current events. Here are a few popular options:

**Google News:** The Google News app is a great choice for staying up-to-date on current events. It offers a personalized news feed based on your interests and location, and includes articles from a wide range of sources.

**Flipboard:** Flipboard is a popular news app that allows you to customize your news feed by selecting your favorite topics and sources. It also features a magazine-style layout that makes reading the news more enjoyable.

**Feedly:** Feedly is an RSS reader app that allows you to follow your favorite websites and blogs in one place. You can organize your feeds by topic, and it

also includes a Discover feature that suggests new content based on your interests.

**BBC News:** The BBC News app offers comprehensive coverage of national and international news, with articles, videos and live broadcasts. It also includes a My News feature that allows you to customize your news feed.

**Reuters:** The Reuters app is a great choice for business and financial news, with real-time updates on stocks, markets, and other financial data. It also includes general news coverage from around the world.

Overall, there are many Android applications available to keep up with the latest headlines and news. It's a matter of personal preference which one you choose, so it's worth trying out a few to see which one works best for you.

About Android studio:

Android Studio is the official integrated development environment (IDE) for Android app development. It is developed by Google and is used by developers to create, test, and deploy Android apps.

Android Studio provides a comprehensive development environment for building Android apps, including a code editor, a visual layout editor, a project management system, and debugging tools. It supports Java, Kotlin, and C++ programming languages.

Some of the features of Android Studio include:

Intelligent code editor with code completion, code highlighting, and code refactoring

Visual layout editor for designing app interfaces

Built-in support for version control systems like Git and GitHub

Gradle-based build system for building and deploying apps

Emulator for testing apps on virtual devices

Android Virtual Device Manager for creating and managing virtual devices

Profiling tools for analyzing app performance and identifying performance bottlenecks

Integration with Firebase for app analytics, testing, and cloud services.

Overall, Android Studio provides a robust and powerful platform for developing Android apps.

## **6.CONCLUSION:**

Android is an excellent platform for keeping up with the latest headlines. With numerous news applications available on the Google Play Store, Android users have access to a vast range of sources, topics, and formats, from traditional newspapers and TV stations to blogs, podcasts, and social media.

Some popular news apps for Android include Google News, Flipboard, Feedly, News360, and SmartNews. These apps use various algorithms, AI technologies, and user preferences to curate personalized news feeds, breaking news alerts, and daily digests that match the user's interests, location, language, and reading habits.

Moreover, Android devices have advanced notification systems, widgets, and home screen customization options that allow users to receive real-time updates, bookmarks, and shortcuts to their favorite news sources and topics. Some Android devices also support voice assistants like Google Assistant or Amazon Alexa, which can read the news out loud, answer questions, and provide recommendations based on voice commands.

Overall, Android provides an intuitive, convenient, and flexible way to stay informed and engaged with the latest headlines, whether you are on the go, at home, or anywhere in between.

## 7. FUTURE SCOPE:

Android is a popular mobile operating system developed by Google, and it plays a significant role in the way we consume news and stay up-to-date with the latest headlines. Here are some potential future developments in Android that could further enhance this experience:

**Increased personalization:** As Android continues to evolve, we can expect to see more personalized news feeds that are tailored to the individual user's interests and preferences. This could be achieved through the use of AI and machine learning algorithms that analyze user behavior and content consumption patterns.

**Integration with other devices:** Android is already available on a range of devices, including smartphones, tablets, and smartwatches. In the future, we may see more seamless integration between these devices, allowing users to receive notifications and updates on the latest headlines across all of their devices.

**Improved accessibility:** Accessibility features are becoming increasingly important, and Android is no exception. Future updates may include enhancements to assist users with visual or hearing impairments to consume news content more easily.

**Integration with augmented reality:** Augmented reality (AR) is already being used in various industries, including gaming and education. It's possible that Android will eventually incorporate AR technology into its news feeds, allowing users to view 3D models or visualizations of news events.

**Increased security:** With cybersecurity becoming a growing concern, Android will likely continue to enhance its security features to protect users' personal information and prevent the spread of fake news.

Overall, the future of Android for keeping up with the latest headlines looks promising, with many potential developments that could improve the user experience and make it easier to stay informed.

## 8.APPENDIX

### A.SOURCE CODE

#### 1.User data class

```
package com.example.newsheadlines

import androidx.room.ColumnInfo
import androidx.room.Entity
import androidx.room.PrimaryKey

@Entity(tableName = "user_table")
data class User(
    @PrimaryKey(autoGenerate = true) val id: Int?,
    @ColumnInfo(name = "first_name") val firstName: String?,
    @ColumnInfo(name = "last_name") val lastName: String?,
    @ColumnInfo(name = "email") val email: String?,
    @ColumnInfo(name = "password") val password: String?,

)
```

#### 2.UserDaoInterface:

```
package com.example.newsheadlines

import androidx.room.*
```

@Dao

interface UserDao {

@Query("SELECT \* FROM user\_table WHERE email = :email")

suspend fun getUserByEmail(email: String): User?

@Insert(onConflict = OnConflictStrategy.REPLACE)

suspend fun insertUser(user: User)

@Update

suspend fun updateUser(user: User)

@Delete

suspend fun deleteUser(user: User)

}

3.userdatabase

package com.example.newsheadlines

import android.content.Context

import androidx.room.Database

import androidx.room.Room

import androidx.room.RoomDatabase

@Database(entities = [User::class], version = 1)



```

abstract class UserDatabase : RoomDatabase() {

    abstract fun userDao(): UserDao

    companion object {

        @Volatile
        private var instance: UserDatabase? = null

        fun getDatabase(context: Context): UserDatabase {
            return instance ?: synchronized(this) {
                val newInstance = Room.databaseBuilder(
                    context.applicationContext,
                    UserDatabase::class.java,
                    "user_database"
                ).build()
                instance = newInstance
                newInstance
            }
        }
    }
}

```

4.userdatabasehelper:

Create Apiservice interface

```
package com.example.newsheadlines
```

```
import retrofit2.Retrofit
```

```
import retrofit2.converter.gson.GsonConverterFactory
```

```
import retrofit2.http.GET
```

```
interface ApiService {
```

```
    // @GET("movielist.json")
```

```
    @GET("top-headlines?country=us&category=business&apiKey=684cb893caf7425abeffad82ac1d0f4e")
```

```
    /// @GET("search?q=chatgpt")
```

```
    suspend fun getMovies() : News
```

```
companion object {
```

```
    var apiService: ApiService? = null
```

```
    fun getInstance() : ApiService {
```

```
        if (apiService == null) {
```

```
            apiService = Retrofit.Builder()
```

```
                // .baseUrl("https://howtodoandroid.com/apis/")
```

```
                .baseUrl("https://newsapi.org/v2/")
```

```
                // .baseUrl("https://podcast-episodes.p.rap
```

```
        }
```

Create news data class

```
package com.example.newsheadlines
```

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

```
import com.google.gson.annotations.SerializedName
```

```
data class News (
```

```
    @SerializedName("status") var status:String?= null,
```

```
    @SerializedName("totalResults") var totalResults : Int? = null,
```

```
    @SerializedName("articles") var articles : ArrayList<Articles> = arrayListOf()
```

```
)
```

Footer

[9:37 AM, 4/21/2023] Susi frd: Create source data class

```
package com.example.example
```

```
import com.google.gson.annotations.SerializedName
```

```
data class Source (
```

```
    @SerializedName("id" ) var id : String? = null,
```

```
        @SerializedName("name" ) var name : String? = null
    )
```

Create Article data class

```
package com.example.example
```

```
import com.google.gson.annotations.SerializedName
```

```
data class Articles (
```

```
    @SerializedName("title" ) var title : String? = null,
    @SerializedName("description" ) var description : String? = null,
    @SerializedName("urlToImage" ) var urlToImage : String? = null,
)
```

Create mainviewmodel class

```
package com.example.newsheadlines
```

```
import android.util.Log
import androidx.compose.runtime.getValue
import androidx.compose.runtime.mutableStateOf
import androidx.compose.runtime.setValue
import androidx.lifecycle.ViewModel
import androidx.lifecycle.viewModelScope
import com.example.example.Articles
```

```

import kotlinx.coroutines.launch

class MainViewModel : ViewModel() {
    var movieListResponse: List<Articles> by mutableStateOf(listOf())
    var errorMessage: String by mutableStateOf("")
    fun getMovieList() {
        viewModelScope.launch {
            val apiService = ApiService.getInstance()
            try {
                val movieList = apiService.getMovies()
                movieListResponse = movieList.articles
            }
            catch (e: Exception) {
                errorMessage = e.message.toString()
            }
        }
    }
}

```

Creating LoginActivity.kt with database

```
package com.example.newsheadlines
```

```

import android.content.Context
import android.content.Intent
import android.os.Bundle
import androidx.activity.ComponentActivity
import androidx.activity.compose.setContent

```

```
import androidx.compose.foundation.Image
import androidx.compose.foundation.background
import androidx.compose.foundation.layout.*
import androidx.compose.foundation.shape.RoundedCornerShape
import androidx.compose.material.*
import androidx.compose.material.icons.Icons
import androidx.compose.material.icons.filled.Lock
import androidx.compose.material.icons.filled.Person
import androidx.compose.runtime.*
import androidx.compose.ui.Alignment
import androidx.compose.ui.Modifier
import androidx.compose.ui.graphics.Color
import androidx.compose.ui.res.painterResource
import androidx.compose.ui.text.font.FontWeight
import androidx.compose.ui.text.input.PasswordVisualTransformation
import androidx.compose.ui.tooling.preview.Preview
import androidx.compose.ui.unit.dp
import androidx.compose.ui.unit.sp
import androidx.core.content.ContextCompat
import androidx.core.content.ContextCompat.startActivity
import com.example.newsheadlines.ui.theme.NewsHeadlinesTheme
```

```
class LoginActivity : ComponentActivity() {
    private lateinit var databaseHelper: UserDatabaseHelper
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        databaseHelper = UserDatabaseHelper(this)
        setContent {
```

```

        LoginScreen(this, databaseHelper)
    }
}

@Composable
fun LoginScreen(context: Context, databaseHelper: UserDatabaseHelper) {
    var username by remember { mutableStateOf("") }
    var password by remember { mutableStateOf("") }
    var error by remember { mutableStateOf("") }

    Column(
        Modifier
            .fillMaxHeight()
            .fillMaxWidth()
            .padding(28.dp),
        horizontalAlignment = Alignment.CenterHorizontally,
        verticalArrangement = Arrangement.Center)

    {
        Image(
            painter = painterResource(id = R.drawable.news),
            contentDescription = "")

        Spacer(modifier = Modifier.height(10.dp))

        Row {
            Divider(color = Color.LightGray, thickness = 2.dp, modifier = Modifier
                .width(155.dp)

```

```

        .padding(top = 20.dp, end = 20.dp))
Text(text = "Login",
    color = Color(0xFF6495ED),
    fontWeight = FontWeight.Bold,
    fontSize = 24.sp, style = MaterialTheme.typography.h1)
Divider(color = Color.LightGray, thickness = 2.dp, modifier = Modifier
    .width(155.dp)
    .padding(top = 20.dp, start = 20.dp))

}

Spacer(modifier = Modifier.height(10.dp))

TextField(
    value = username,
    onChange = { username = it },
    leadingIcon = {
        Icon(
            imageVector = Icons.Default.Person,
            contentDescription = "personIcon",
            tint = Color(0xFF6495ED)
        )
    },
    placeholder = {
        Text(
            text = "username",
            color = Color.Black
        )
    },

```



```

        colors = TextFieldDefaults.textFieldColors(
            backgroundColor = Color.Transparent
        )
    )

    Spacer(modifier = Modifier.height(20.dp))

    TextField(
        value = password,
        onChange = { password = it },
        leadingIcon = {
            Icon(
                imageVector = Icons.Default.Lock,
                contentDescription = "lockIcon",
                tint = Color(0xFF6495ED)
            )
        },
        placeholder = { Text(text = "password", color = Color.Black) },
        visualTransformation = PasswordVisualTransformation(),
        colors = TextFieldDefaults.textFieldColors(backgroundColor = Color.Transparent)
    )

    Spacer(modifier = Modifier.height(12.dp))

    if (error.isNotEmpty()) {
        Text(

```

```

        text = error,
        color = MaterialTheme.colors.error,
        modifier = Modifier.padding(vertical = 16.dp)
    )
}

```

```

Button(
    onClick = {
        if (username.isNotEmpty() && password.isNotEmpty()) {
            val user = databaseHelper.getUserByUsername(username)
            if (user != null && user.password == password) {
                error = "Successfully log in"
                context.startActivity(
                    Intent(
                        context,
                        MainPage::class.java
                    )
                )
                //onLoginSuccess()
            } else {
                error = "Invalid username or password"
            }
        } else {
            error = "Please fill all fields"
        }
    },
    shape = RoundedCornerShape(20.dp),
    colors = ButtonDefaults.buttonColors(backgroundColor = Color(0xFF77a2ef)),
    modifier = Modifier.width(200.dp)
)

```

```

        .padding(top = 16.dp)
    ) {
        Text(text = "Log In", fontWeight = FontWeight.Bold)
    }

    Row(modifier = Modifier.fillMaxWidth()) {
        TextButton(onClick = {
            context.startActivity(
                Intent(
                    context,
                    RegistrationActivity::class.java
                )))
        { Text(text = "Sign up",
            color = Color.Black
        )}

        Spacer(modifier = Modifier.width(100.dp))

        TextButton(onClick = { /* Do something! */ })
        { Text(text = "Forgot password ?",
            color = Color.Black
        )}
    }

}

private fun startMainPage(context: Context) {
    val intent = Intent(context, MainPage::class.java)
    ContextCompat.startActivity(context, intent, null)
}

```

```
}
```

Androidmanifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">
    <uses-permission android:name="android.permission.INTERNET"/>
    <uses-permission android:name="android.permission.ACCESS_WIFI_STATE"/>
    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@drawable/news_app_icon"
        android:label="@string/app_name"
        android:supportsRtl="true"
        android:theme="@style/Theme.NewsHeadlines"
        tools:targetApi="31">
        <activity
            android:name=".DisplayNews"
            android:exported="false"
            android:label="@string/title_activity_display_news"
            android:theme="@style/Theme.NewsHeadlines" />
        <activity
            android:name=".RegistrationActivity"
            android:exported="false"
            android:label="@string/title_activity_registration"
            android:theme="@style/Theme.NewsHeadlines" />
```

```
<activity
    android:name=".MainPage"
    android:exported="false"
    android:label="@string/title_activity_main_page"
    android:theme="@style/Theme.NewsHeadlines" />
<activity
    android:name=".LoginActivity"
    android:exported="true"
    android:label="@string/app_name"
    android:theme="@style/Theme.NewsHeadlines">
    <intent-filter>
        <action android:name="android.intent.action.MAIN" />

        <category android:name="android.intent.category.LAUNCHER" />
    </intent-filter>
</activity>
</application>

</manifest>
```