**NEWSTORO**

**What is NEWSTORO?**

In today's fast-paced world, staying informed about the latest news and updates is more important than ever. Enter Newstoro, your go-to destination for curated, personalized news content delivered straight to your fingertips. With Newstoro, you can stay ahead of the curve, effortlessly accessing the stories that matter most to you, whenever and wherever you are.

Newstoro takes the hassle out of keeping up with the news by offering a seamless and intuitive user experience. Whether you're interested in global headlines, local events, or niche topics, Newstoro has you covered. Our cutting-edge technology scours the web for the most relevant and engaging news articles, ensuring that you're always in the know.

**Project Summary: Newstoro**

Newstoro is a cutting-edge news application designed to provide users with a personalized and seamless news consumption experience. In an era of information overload, Newstoro aims to simplify the process of staying informed by delivering curated news content tailored to each user's interests and preferences.

**Concepts in App:**

1. News Api
2. MVVM
3. Retrofit
4. Room
5. Nav Graph
6. Fragments
7. Coroutines
8. WebView

**News API:**

News API: <https://newsapi.org/>

* Serves as the backbone for the app, providing access to diverse news articles and headlines.
* Ensures app stays updated with the latest news.

**MVVM (Model-View-View Model):**

* Maintains separation of concerns with Model handling data retrieval, View Model managing UI-related logic, and View displaying news content.
* Promotes a scalable and maintainable code structure.

**Retrofit:**

* Simplifies network requests by connecting to the News API.
* Enables seamless integration of real-time news data.
* Enhances app's responsiveness and user experience.

**Room:**

* Facilitates local storage of news data.
* Ensures users can read previously accessed news content offline.

**Nav Graph:**

* Visually outlines navigation flow in the app.
* Simplifies navigation management and enhances user experience.

**Fragment:**

* Represents modular UI components.
* Enhances app's adaptability to various screen sizes and user interactions.

**Coroutines:**

* Manages asynchronous tasks efficiently.
* Ensures smooth and responsive interactions.

**WebView:**

* Allows seamless access to full articles or external links.

**Functionality of News App:**

1. **Trending** Headlines
2. **Search Functionality**
3. Favourite Headlines
4. Offline Reading

**Trending Headlines**

• Displays curated list of popular, widely discussed news articles.

• Provides quick overview of current hot topics.

• Helps stay informed about latest, relevant news.

**App Search Functionality**

• Allows users to find specific news articles, topics, or keywords.

• Inputs user interests for relevant articles retrieval.

• Enhances user engagement with personalized, efficient access.

**Favourite Headlines**

• Saves and organizes preferred articles.

• Marks articles as favourites.

• Enhances user experience.

• Enables convenient revisiting and enjoyment of preferred content.

**Offline Reading**

* Users can save articles for offline reading
* Allowing them to access news content even when they don't have an internet connection.

**Technical Talks about Project**

1. To, display all these articles are using news API which I’ll show you later. So, with the help of that, it fetches the real-time data or news and displays it on the recycler view.
2. Then, with the help of web view – we, can open the article in the app itself without going to any browser.
3. Then, by clicking on add to fav – that, particular article gets saved in the room database so what you see here is all saved in the room database hence later whenever you open the app again – they, will be still here.
4. Lastly, search uses API only, and other topics like MVVM will help us to make the project organized and segregated in model view and View Model not, just that but also the navigation component will help us to navigate from one screen to another screen effortlessly.