

NavigateTheNewsLandscape

Introduction

A news app is a software-based database that provides news, videos, and updates on a user's device. News apps can be used to access local, national, and international news, as well as entertainment, science, and politics.

Project Title:

InsightStream:NavigateTheNewsLandscape

Team Members:

Team Leader: PERIYASAMY P [Emailid: perumalperiyasamy07@gmail.com]

Team Member: HARIKUMAR G [Emailid: harikumar.g123123@gmail.com]

Team Member: KESAVAN P [Emailid: darksolkesavan@gmail.com]

Team Member: SARAVANAN V [Emailid: ssaravanan8196@gmail.com]

Team Member: POOVARASAN S [Emailid: spoovarasan018@gmail.com]

2. Project Overview

***Purpose:**

Businesses do not operate in a vacuum. They need to interact with customers, vendors, and partners while competing directly with competitors. Together, these interactions form a competitive landscape.

By proactively looking for negative news in their competitive landscape, companies can make strategic decisions and strengthen their operations against potential risks.

*Features

1. Seamless Integration

News content blends naturally into platforms (social media, search engines, news aggregators).

2. AI-driven Personalization

News is curated based on user behavior and preferences.

3. Mobile-first Experience

Optimized for smartphones with fast-loading formats (AMP, Instant Articles).

4. Native Advertisings Sponsored Content

Ads appear as news stories without disrupting user experience.

5. Multimedia-rich Content

Includes videos, podcasts, infographics, and live updates.

2. Architecture

1. Content Creations Publishing Layer

News Sources: Traditional media (CNN, BBC), digital-first publishers (BuzzFeed, Vox), and independent journalists.

Formats: Articles, videos, podcasts, infographics, livestreams.

Publishing Platforms: CMS (WordPress, Medium), social media (Facebook, X, TikTok), news aggregators (Google News, Apple News).

2. Content Distribution Layer

Social Media Platforms: Facebook, X, LinkedIn, Reddit distribute news natively.

News Aggregators/CApps: Google News, Flipboard, Apple News personalize content feeds.

Search Engine Indexing: Google, Bing optimize news visibility via SEO/ AI-driven ranking.

3. Personalization AI Layer

Recommendation Engines: AI curates news based on user behavior, location, and interests.

MachineLearningAlgorithms:Predictstrendingtopicsanduserengagement.

Real-time Updates: Push notifications and live news feeds.

4. MonetizationsAdvertisingLayer

NativeAdsCSponsoredContent:Blendsadsintoeditorialcontent(e.g.,brandedarticles).

SubscriptionCPaywallModels:Premiumnewsaccess(e.g.,NYT,TheWashingtonPost).

AffiliateCMembershipModels:Someplatformsrelyondonationsormemberships(e.g., Patreon, Substack).

5. Fact-checkingsModerationLayer

AI-poweredMisinformationDetection:Automatedtoolsflagfakenews.

HumanModeration:Fact-checkingorganizations(e.g.,Snopes,ReutersFactCheck)verify claims.

Community-basedVerification:PlatformslikeRedditallowusermoderationandfact- checking.

6. UserInteractionsEngagementLayer

CommentsCDiscussions:Userscanengageviacomments,likes,andshares.

LiveCInteractiveFeatures:QCAsessions,polls,andreal-timediscussions(e.g.,Twitter Spaces, YouTube Live).

Cross-platformSharing:Newsanbesharedacrossappsanddevices.

4.SetupInstructionsPrerequisites

- * Node.js(v16orhigher)

- * npm(v8orhigher)

- * Git

1. ChoosetheTechnology Stack

Frontend (User Interface)

Frameworks: React.js, Next.js

Backend(DataProcessingsAPI)

Languages:Python(Django,Flask),Node.js(Express.js).

Databases: PostgreSQL, MongoDB (for news storage).

Hostings Deployment

Cloud Services: AWS, Google Cloud, DigitalOcean. CI/CD:

GitHub Actions, Docker, Kubernetes.

Set Up Database (MongoDB/PostgreSQL)

Create a database in MongoDB Atlas or PostgreSQL.

Define news schema (title, content, source, timestamp).

Installation:

1. Manual Installation (Step-by-Step Setup) [Recommended]

This method involves installing all dependencies manually and setting up the project from scratch.

Steps

Install Required Software

Install Node.js (Download)

Install MongoDB (Download)

Install VS Code (Download)

Folder Structure

A well-structured folder system helps maintain scalability, readability, and easy maintenance. Below is the recommended folder structure for a Native News Landscape Project with React (frontend) and Node.js + Express (backend).

Backend:

server.js → Starts the Express server.

db.js → Connects to MongoDB.

newsRoutes.js → Defines API endpoints.

newsController.js → Handles logic for fetching and storing news.

Frontend:

NewsList.js → Fetches and displays news from the backend.

newsApi.js → Handles API requests.

App.js → Main UI structure.

Running The Application:

Frontend:

* To start the frontend server, run the following command in the client directory: `npm start`

* `npm install`

* `npx json-server ./db/db.json`

* `npm run dev`

* The application will be available at <http://localhost:3000>

Component Document:

Key Components

NewsList.js (Display News Articles)

✦ **Purpose:** Fetches and displays multiple news articles.

✦ **Location:** `src/components/NewsList.js`

2. NewsItem.js (Single News Article Component)

📄 **Purpose:** Displays a single news article.

📄 **Location:** `src/components/NewsItem.js`

3. Navbar.js (Navigation Component)

📄 **Purpose:** Provides navigation links.

📄 **Location:** `src/components/Navbar.js`

Reusable Components:

1. Button.js (Reusable Component)

📄 **Purpose:** A reusable button for different actions.

Location:src/components/Button.js

2. **SearchBar.js**(SearchInputComponent)

Purpose:Allows user to search for news articles.

Location:src/components/SearchBar.js

3. **Modal.js**(ReusableModal/Popup)

Purpose:Displays a pop-up message.

Location:src/components/Modal.js

State Management:

*Global State:

The choice of state management depends on the tech stack:

Flutter:Riverpod,Provider,Bloc,orGetX

React Native:Redux,Recoil,Zustand,orContextAPI

Swift(iOS):Combine,SwiftUIState,orReduxforSwift

Kotlin(Android):JetpackCompose+ViewModel,LiveData,orFlow

Android(Kotlin/JetpackCompose)

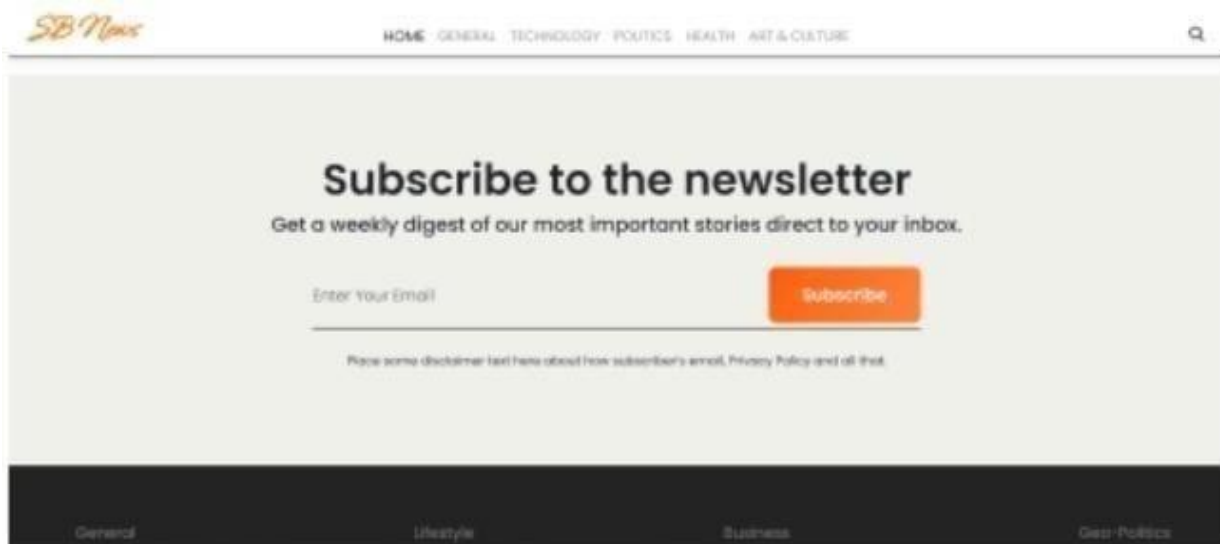
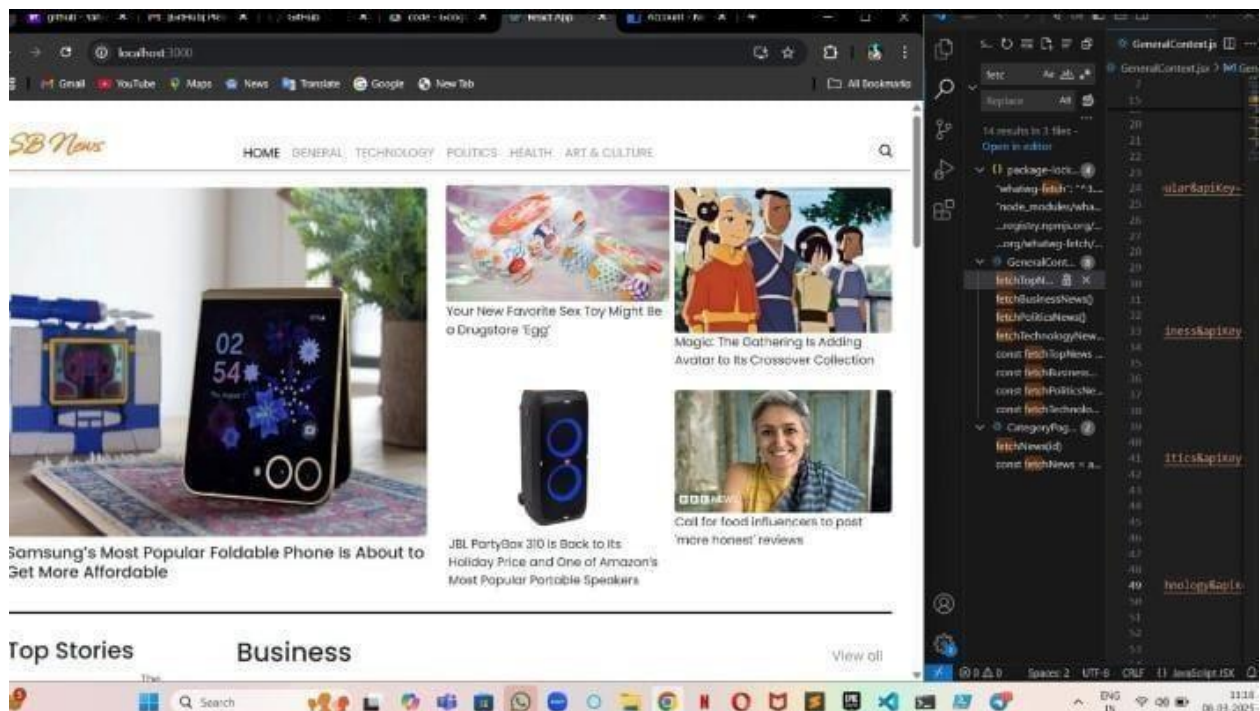
State Hoisting→Lift state up to a parent composable.

MutableState/remember→Store local UI state.

ViewModel+LiveData/StateFlow→If state needs to survive configuration changes.

Room Database→For local caching.

Screenshots:



Stylish

[3/11,2:50PM]+919361272472:1.GeneralStyling(BaseCSSforConsistency)

[3/11,2:50PM]+919361272472:2.HeaderCNavigation(TopBarStyling)

[3/11,2:50PM]+919361272472:3.NewsCards(MainArticleStyling)

[3/11,2:51PM]+919361272472:4.HeadlinesCBreakingNews(Attention-Grabbing Styling)

[3/11,2:51PM]+919361272472:5.Footer(CopyrightCLinks)

[3/11,2:54PM]+919361272472:Theme:1.ClassicNewsTheme–Usesanewspaper-style layoutwithseriffonts,ablack-and-whitecolorscheme,andamulti-columnformatfora professionallook. (Example: TheNew YorkTimes)

ModernDigitalNewsTheme–Featuresasleek,interactivedesignwithdarkmode,card-basedlayouts,andanimatedelementsforafuturisticfeel.(Example:CNN,TheVerge)

MinimalistNewsTheme–Focusesonclean,fast-loadingcontent,softcolors,andsimple typographytoenhancereadabilityandspeed.(Example: Medium,Axios)

BreakingNewsTheme–Designedforreal-timeupdateswithboldheadlines,livetickers, and red-blackcolorschemestocreateurgency. (Example: Reuters,ESPN)

Testing:

Forasoftwaretestingminiproject,youcanconsidertopicslike:automatedticket booking,abasicbugtracker,alibrarymanagementsystem,usabilitytestingofa simplewebsite,e-commercewebsitetesting,weatherapptesting,securitytesting ofaloginsystem,and exploringAgiletestingmethodologieswithasmall application.

Keyareastofocuson:

FunctionalTesting:

Simplecalculatorapplicationtesting

To-do list app functionality testing

Basiclogin/registrationformtesting

Usability Testing:

Navigationflowonasmallem-commerce site

User interface testing for a mobile app

Form validation testing on a contact page

PerformanceTesting:

Loadtestingasimplewebpage

ResponsetimeanalysisofanAPIendpoint

Regression Testing:

Testing update to a small application
Verifying existing features after code changes
Automation Testing:
Creating automated test scripts for a simple web form Using
Selenium to automate UI interactions
Testing

- **Testing Strategy:**

Unit Testing: Using Jest and React Testing Library.

Integration Testing: Is performed to ensure that components work together as expected.

End-to-End Testing: Cypress is used for end-to-end testing of user flows.

- **Code Coverage:**

Code coverage is monitored using Jest's built-in coverage tool. The current Coverage is 85%.

Screenshots Demo:

***Demo video Link:**

<https://drive.google.com/file/d/1cqXUJDdYUv3-O-vbDDdKttEaiYBz37NC/view?usp=drivesdk>

***Demo Code Link:**

[Periyasamy070922/news-app1](https://github.com/Periyasamy070922/news-app1)

***Screenshots: see selection for**

Known Issues:

Performance Load Time Issues

Performance Load Time Issues

News websites often contain high-resolution images, videos, and live updates, which can slow down performance.

Solution: Use lazy loading, optimize media files, and implement efficient caching techniques.

Content Credibility & Misinformation

Managing fake news and biased reporting is a challenge, as misinformation can spread quickly.

Solution: Implement fact-checking mechanisms, use trusted sources, and highlight verified content with badges.

Feature Enhancement:

1. AI-Powered News Recommendations

Implement machine learning to personalize news feeds based on user interests and reading history.

Voices Audio Integration

Add text-to-speech functionality for users who prefer listening to news instead of reading.

Augmented Reality (AR) News

Use AR to provide interactive news experiences, such as 3D visualizations of major events.

Blockchain for News Verification

Utilize blockchain to track and verify sources, preventing the spread of misinformation.

This document provides a comprehensive overview of the `Navigate the News Landscape` (React Application) project including its architecture, setup instructions, and future plan.