Frontend Development with React.js - News App Documentation

---

1. **Introduction**

   The News App is designed to provide users with an easy-to-navigate platform to stay informed with the latest news. It fetches articles from a third-party news API and displays them in a categorized, user-friendly format. Users can browse articles, filter by different categories (such as World, Technology, Sports, etc.), search for specific topics, and read the full content of each article.

   - **Project Title**: News App
   - **Team Members**:

   Team Leader : K.Sarathi

   Team member :M.Sachin

   Team member :D.Sanjay

   Team member : M.Sathish

   Team member : G.Tamizh selvan

---

2. Project Overview

- **Purpose**:
  The News App allows users to view the latest news articles, filter them by categories (e.g., Sports, Technology, Business), and access detailed information on each article.
- **Features**:
  - Display a list of news articles fetched from an API.
  - Categorized sections (e.g., World, Business, Technology, etc.).
  - Search functionality to find articles by keywords.
  - Article details page for reading the full content.

- o Pagination to navigate between pages of news.
- o Responsive design to ensure accessibility on various screen sizes.

---

3. Architecture

- **Component Structure**:
  - o **App**: Main wrapper component containing routing logic.
  - o **NewsList**: Displays a list of articles.
  - o **ArticleCard**: A reusable component to display brief article info (title, description).
  - o **ArticleDetail**: Displays full article details.
  - o **SearchBar**: Allows users to search for specific articles.
  - o **CategoryFilter**: A dropdown for filtering news by categories.
- **State Management**:
  - o **React Context API** is used for global state management (e.g., storing the list of articles, categories).
  - o **Local state** is managed using `useState` and `useReducer` for individual component logic (e.g., search input, selected category).
- **Routing**:
  - o **React Router** is used to manage the routing structure.
  - o `/` - Home page displaying the news list.
  - o `/article/:id` - Detailed view of the selected article.
  - o `/search` - Displays articles based on search criteria.

---

4. Setup Instructions

- **Prerequisites**:
  - o Node.js (version 14 or above)
  - o npm (version 6 or above)
- **Installation**:
  1. Clone the repository:
     `git clone https://github.com/yourusername/news-app.git`
  2. Navigate to the project directory:
     `cd news-app`
  3. Install dependencies:
     `npm install`
  4. Configure environment variables:
     - Create a `.env` file in the root directory and add your API key from the news API provider.
  5. Start the development server: `npm start`

---

5. Folder Structure

- **Client**:
  - `src/`
    - `components/` – All the React components.
    - `pages/` – Contains main pages (Home, ArticleDetail).
    - `assets/` – Static assets like images, logos, etc.
    - `utils/` – Utility functions, helper hooks, etc.
    - `styles/` – All CSS/SCSS files.
    - `App.js` – Root component that defines routes.
    - `index.js` – Entry point for the React app.
- **Utilities**:
  - Custom hooks: `useFetchArticles` for fetching news data.
  - Utility functions: `formatDate` for formatting article dates.

---

6. Running the Application

- To start the frontend server locally:
  1. In the terminal, navigate to the project's root folder.
  2. Run the following command:
     `npm start`
  3. Open your browser and go to `http://localhost:3000` to view the app.

---

7. Component Documentation

- **Key Components**:
  - **NewsList**:
    - Purpose: Displays a list of news articles.
    - Props:
      - `articles`: An array of article objects.
  - **ArticleCard**:
    - Purpose: Displays a card view of an article.
    - Props:
      - `title`: The article's title.
      - `description`: The article's description.
      - `url`: The article's detailed URL.
- **Reusable Components**:
  - **SearchBar**:
    - Purpose: Allows users to search for articles.
    - Props:
      - `onSearch`: Callback function to handle search input changes.
  - **CategoryFilter**:

▪ Purpose: Filters the articles by selected category.
▪ Props:
  ▪ `categories`: List of categories to filter by.
  ▪ `onCategoryChange`: Callback function when category is changed.

---

8. State Management

- **Global State**:
  o Articles are stored in global state using the Context API. All components that need access to articles (e.g., NewsList, ArticleCard) consume the global state.
- **Local State**:
  o Each component that requires internal state (e.g., SearchBar's input, selected category in CategoryFilter) uses `useState` or `useReducer`.

---

9. User Interface

- **UI Screenshots/GIFs**:
  o **Home Page**: Screenshot of the main news feed.
  o **Article Detail Page**: Screenshot showcasing full article content.
  o **Search Results Page**: GIF showing search functionality.

---

10. Styling

- **CSS Frameworks/Libraries**:
  o **Styled-Components** is used to manage scoped styles within each React component.
  o **Sass** is used for more complex styles and theming.
- **Theming**:
  o A custom light/dark mode theme is implemented using `styled-components` for a seamless user experience across different preferences.
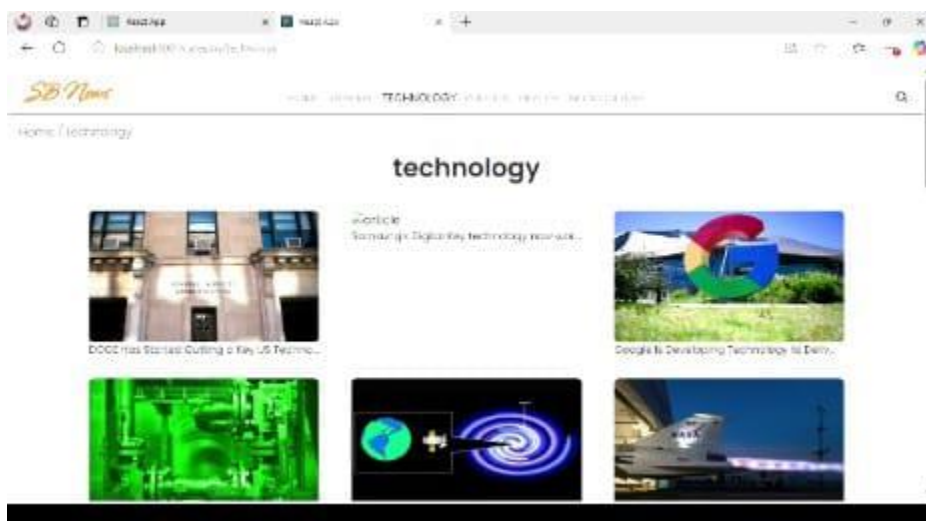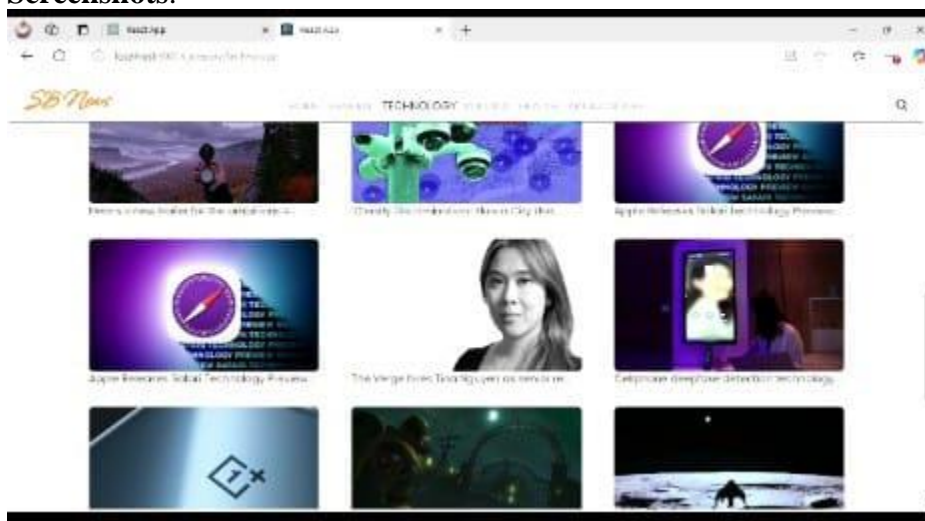
---

11. Testing

- **Testing Strategy**:
  o Unit tests for individual components (e.g., ArticleCard, SearchBar) using **Jest**.
  o Integration tests for component interactions (e.g., submitting search queries) using **React Testing Library**.
  o End-to-end tests with **Cypress** to simulate real user interactions.

- **Code Coverage**:
  - Code coverage is tracked using Jest and displayed via a coverage report.

---

12. Screenshots or Demo

- **Screenshots**:

-

---

13. Known Issues

- **Bug**: Pagination sometimes fails to load the next set of news articles.
- **Fix**: Currently investigating this issue; will be addressed in future updates.

---

14. Future Enhancements

- **Features to add**:
  - Push notifications for breaking news.
  - Bookmarking articles to save them for later.
  - Ability to share articles on social media.
  - Offline mode to allow users to read news when no internet connection is available.

This format should give you a comprehensive idea of how to document your News App built with React.js. Let me know if you need more details!