### InsightStream: Navigate the News Landscape

Project Documentation

A React.js Based Web Application

ST JOSEPH’S COLLEGE OF ARTS AND SCIENCE FOR WOMENS,HOSUR

PREPARED BY :

TEAM LEADER : RUBIKA S

TEAM MEMBERS :

SARANYA S

SARUTHI B

SATHYA M

1. General Project Description

## 1.1 Abstract

This project is a news website built using React.js that delivers the latest articles and top stories to users in a clean, responsive, and modern design. The system is structured to provide category-based news browsing, newsletters, and interactive features like navigation, search, and subscriptions. The objective of this project is to create a lightweight, dynamic, and scalable platform for publishing and consuming news content.

## 1.2 Objectives

- To design a modern, user-friendly news website.  
- To implement reusable and modular React.js components.  
- To display categorized news content dynamically.  
- To enable users to subscribe to newsletters.  
- To create a responsive layout suitable for desktop and mobile users.

## 1.3 Features

- Homepage with top stories and highlighted articles.  
- Categorized news pages.  
- Newsletter subscription system.  
- Responsive Navbar and Footer.  
- Search-friendly layout with SEO-ready structure.  
- Styled using modular CSS for maintainability.

## 1.4 Results

The resulting project is a functional React.js web application that can be deployed on platforms like Netlify, Vercel, or any Node.js supported hosting service. The application provides seamless navigation, organized news presentation, and a smooth user experience.

## 1.5 Conclusion & Future Scope

This project demonstrates how React.js can be leveraged to build modular, scalable, and visually appealing web applications. In the future, features like real-time API integration for fetching live news, user accounts, personalized recommendations, and an admin panel for managing content can be added to enhance functionality.

# 2. Technical Documentation

## 2.1 Technology Stack

- \*\*Frontend Framework\*\*: React.js  
- \*\*Language\*\*: JavaScript (ES6+)  
- \*\*Styling\*\*: CSS Modules  
- \*\*Build Tool\*\*: Node.js with npm  
- \*\*Deployment\*\*: Supports platforms like Netlify, Vercel, or GitHub Pages

## 2.2 Project Structure

The project contains the following major directories and files:  
  
• public/ - Contains static assets (HTML, icons, manifest).  
• src/ - Contains the source code.  
 - index.js: Entry point of the React app.  
 - App.js: Root component that defines routing and layout.  
 - components/: Holds reusable React components (Navbar, Footer, Newsletter, TopStories).  
 - styles/: CSS files for modular styling.  
• package.json - Manages dependencies and scripts.  
• README.md - Documentation and usage guide.

## 2.3 Component Overview

- \*\*Navbar.jsx\*\*: Provides navigation across categories.  
- \*\*Hero Section\*\*: Displays highlighted news.  
- \*\*TopStories.jsx\*\*: Lists trending or top articles.  
- \*\*NewsLetter.jsx\*\*: Allows users to subscribe.  
- \*\*Footer.jsx\*\*: Contains website footer information.

## 2.4 Workflow

1. User visits the homepage served by index.html.  
2. React.js initializes and renders App.js.  
3. Navbar and Footer are loaded globally.  
4. The homepage displays top stories and featured articles.  
5. Users can navigate to categories, view news, and subscribe to newsletters.  
6. The app responds dynamically without full page reloads thanks to React’s virtual DOM.