

**SAVEETHA SCHOOL OF ENGINEERING**

**SAVEETHA INSTITUTE OF MEDICAL AND TECHNICAL SCIENCES**

**CAPSTONE PROJECT REPORT**

**PROJECT TITLE**

Development of a Real-Time News Website Using Agile Kanban Methodology.

**REPORT SUBMITTED BY**

Laxmi Narayanan D-192324054

**COURSE CODE / NAME**

CSA1086 / SOFTWARE ENGINEERING FOR PROGRAMMING

**Abstract:**

This capstone project aims to develop a real-time news website using an Agile methodology based on the Kanban framework. The website fetches and displays categorized news using a live API, ensuring responsiveness and user-friendly navigation. Hosted on Netlify and managed through GitHub, the project focuses on delivering incremental updates and continuous workflow improvement. The Kanban framework facilitates visual task management and ensures seamless collaboration, while challenges such as API rate limits and responsiveness are addressed with innovative solutions. The project highlights the effectiveness of Kanban in managing a streamlined and efficient development process.

**Introduction:**

The modern world thrives on timely and accurate information, making real-time news delivery essential. This project aims to develop a news website that provides users with categorized and up-to-date news articles. The website integrates a news API for live updates and ensures accessibility across devices. Leveraging the Kanban framework of Agile methodology, the project emphasizes a continuous workflow, enabling flexibility and improved task management. Hosted on Netlify and version-controlled with GitHub, the project demonstrates the practical application of Agile principles in software development.

**Project Scope:**

The project encompasses the following objectives:

1. Build a responsive news website.
2. Integrate an external API to fetch and display categorized news.
3. Implement features such as news filtering and categorization.
4. Ensure compatibility across multiple devices.
5. Host the website on Netlify for public access.
6. Maintain the codebase with version control using GitHub.

The scope focuses on delivering a user-friendly experience while maintaining flexibility for future enhancements. Tasks are broken into manageable work items, facilitating incremental progress and efficient project execution.

**Methodology Overview:**

The project adopts the Kanban framework of Agile methodology, which emphasizes continuous delivery, flexibility, and improved task visualization.

**Kanban Board:** The board consists of four columns: To Do, In Progress, Code Review, and Done.

**Work-in-Progress (WIP) Limits:** WIP limits ensure that the team focuses on a manageable number of tasks simultaneously, avoiding bottlenecks.

**Incremental Updates:** Tasks move through the workflow incrementally, promoting early feedback and continuous improvement.

This approach ensures that tasks are prioritized effectively and progress is made steadily toward project completion.

**Backlog Creation:**

The backlog comprises tasks broken into actionable work items, organized by priority:

1. Set up the GitHub repository and initialize the project.
2. Design the website’s homepage and navigation structure.
3. Integrate the news API for fetching live updates.
4. Add functionality to filter and categorize news articles.
5. Optimize the website’s layout for mobile and desktop devices.
6. Host the project on Netlify and verify deployment.

Each task is added to the Kanban board and moved through the workflow as it progresses.

**Epics, User Stories, and Tasks:**

**Epic 1: Frontend Design**

Story 1: Build the Website Structure

* Task 1: Set up the basic layout in index.html.
* Task 2: Include sections for the header, body, and footer.

Story 2: Apply Website Styling

* Task 1: Style the layout using styles.css.
* Task 2: Add responsive styles for mobile and desktop devices.

Story 3: Integrate SVG Assets

* Task 1: Add darksvg.svg for the dark mode UI.
* Task 2: Add lightsvg.svg for the light mode UI.

**Epic 2: Backend Development**

Story 1: Set Up Server and Database

* Task 1: Set up a Node.js
* Task 2: Create a database schema for news articles.
* Task 3: Configure the connection between the server and database
* Task 4: Test the server and database setup with sample data.

Story 2: Create API Endpoints for News Data

* Task 1: Develop a GET endpoint to fetch all news articles.
* Task 2: Develop a POST endpoint to add a new article.
* Task 3: Develop a DELETE endpoint to remove an article.
* Task 4: Develop a PUT endpoint to update an existing article.
* Task 5: Test all endpoints using tools like Postman or cURL.

Story 3: Integrate News API (If Using Third-Party Data)

* Task 1: Research and choose a suitable news API
* Task 2: Set up an API key and integrate it with your backend.
* Task 3: Write a script to fetch and store news data in the database.
* Task 4: Set up a scheduler to periodically refresh the news content.
* Task 5: Test API integration for errors like rate limits or invalid keys.

**Epic 3: Light/Dark Mode Functionality**

Story 1: Implement Dark Mode

* Task 1: Develop toggling functionality in lightmode.js.
* Task 2: Integrate dark mode styles with the website.

Story 2: Test Light/Dark Mode

* Task 1: Ensure compatibility across major browsers.
* Task 2: Verify smooth transitions between modes.

**Epic 4: JavaScript Functionality**

Story 1: Add Interactivity to the Website

* Task 1: Implement navigation functionality in index.js.
* Task 2: Add dynamic behaviors like animations or hover effects.

Story 2: Debug and Optimize Code

* Task 1: Fix JavaScript errors in index.js.
* Task 2: Optimize the code for performance and responsiveness.

**Epic 5: Deployment and Hosting**

Story 1: Host the Project Online

* Task 1: Set up a GitHub repository for the project.
* Task 2: Deploy the website on Netlify.

Story 2: Test the Hosted Site

* Task 1: Verify all links and features work as expected.
* Task 2: Test responsiveness and browser compatibility.

**Epic 6: Documentation**

Story 1: Complete Project Documentation

* Task 1: Update README.md with usage instructions.
* Task 2: Add contributor guidelines.

Story 2: Document Code

* Task 1: Add comments to index.js.
* Task 2: Document the light/dark mode implementation.

**Workflow:**

The Kanban workflow facilitates a streamlined and transparent process:

**To Do:** Tasks that are ready to be started

* Documentation

**In Progress:** Tasks actively being worked on, like creating responsive layouts or API integration.

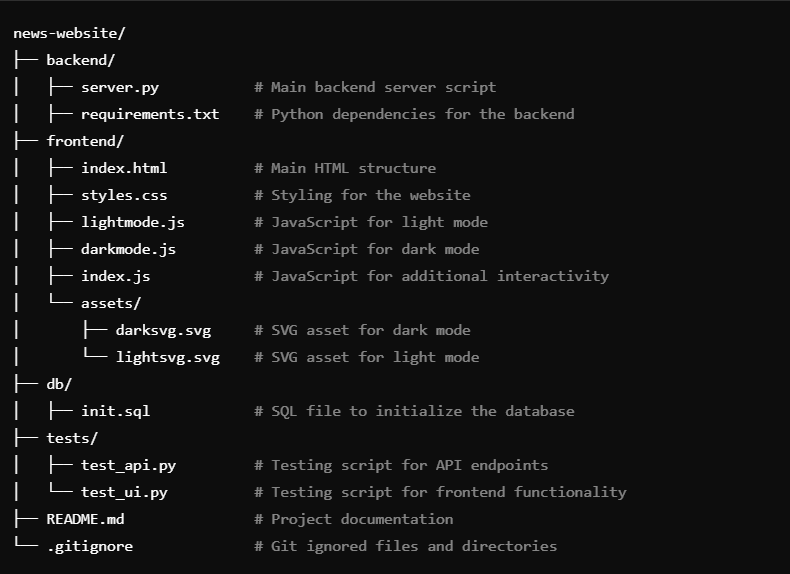
* Backend Development
* Deployment and Hosting

**Done:** Verified and completed tasks are moved to this column.

* Frontend Design
* JavaScript Functionality
* Light/Dark Mode Functionality

**Code Review:** Completed tasks undergo review to ensure quality and functionality.

**Directory Structure:**



This workflow supports incremental progress, continuous delivery, and early identification of potential issues.

**Challenges and Solutions:**

**API Rate Limits:**

* **Challenge:** Limited API calls affecting the retrieval of live news data.
* **Solution:** Implement caching mechanisms to store data temporarily and display fallback content.

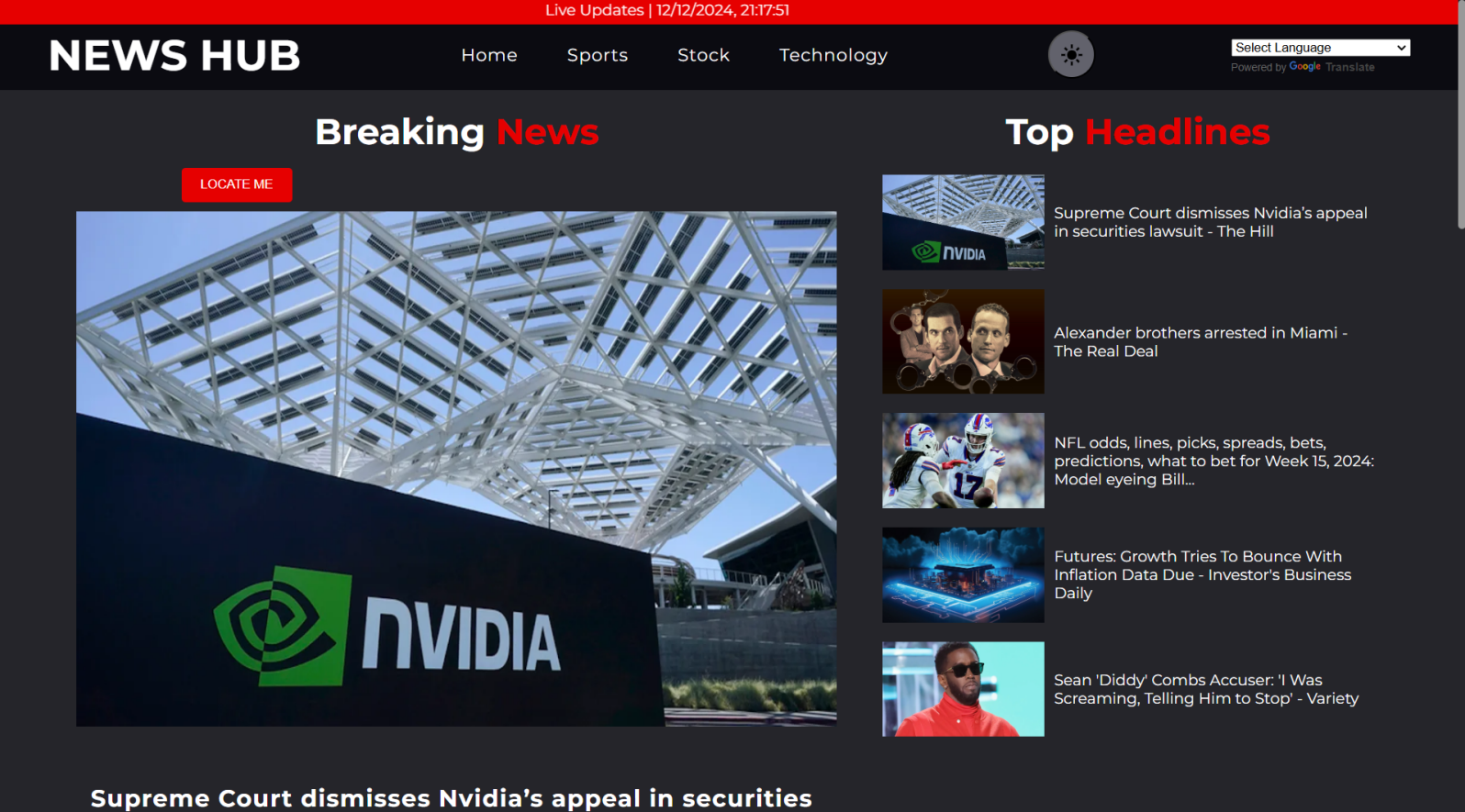
**Responsive Design:**

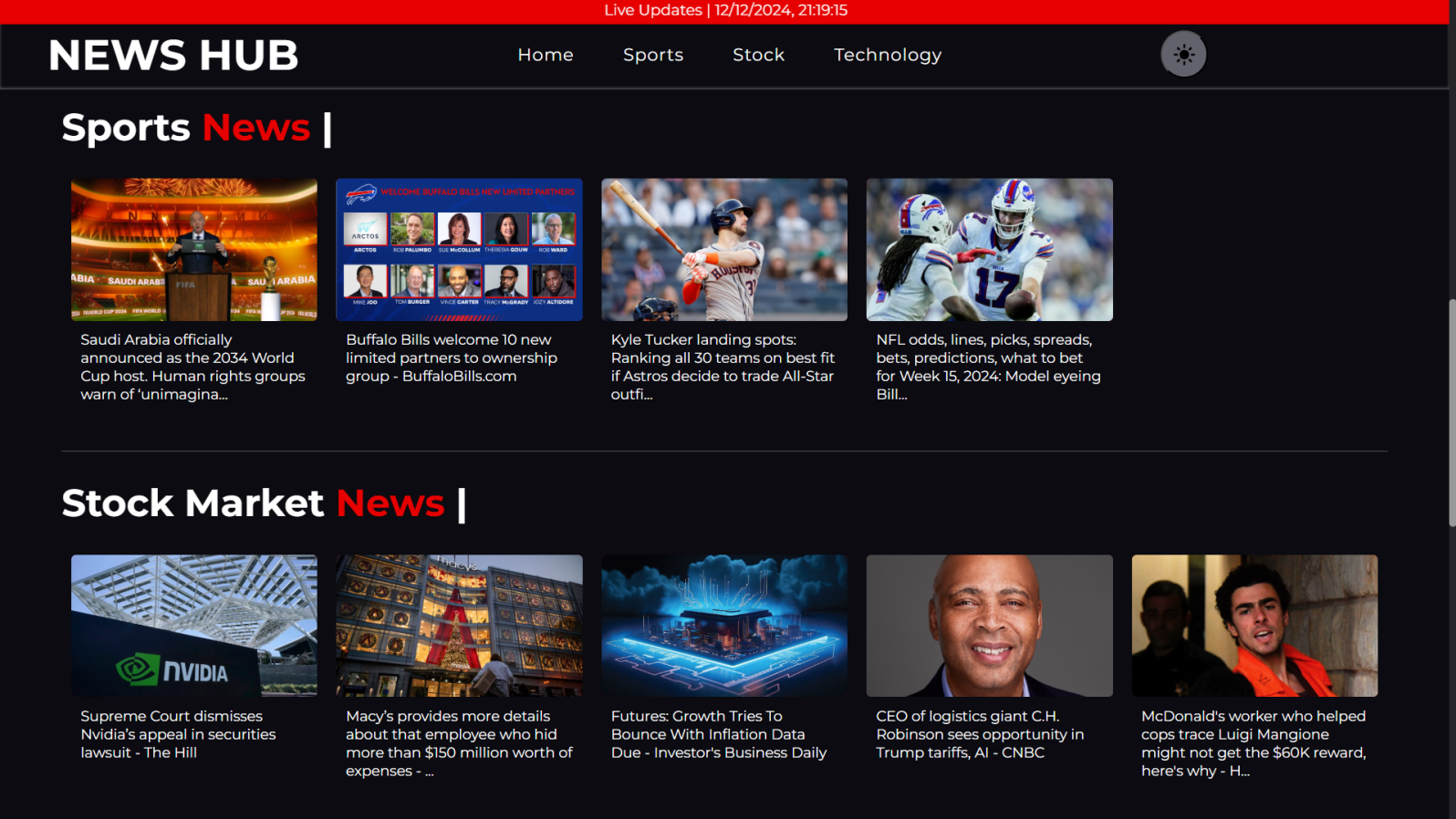
* **Challenge:** Ensuring the website displays consistently across devices.
* **Solution:** Use media queries and conduct thorough testing on multiple screen sizes.

**Workflow Bottlenecks:**

* **Challenge:** Delays due to too many tasks in progress.
* **Solution:** Implement strict WIP limits to maintain focus and avoid task overload.

**Website Visualization:**





**Conclusion:**

This project successfully demonstrates the application of the Kanban framework to develop a real-time news website. The Agile approach ensured flexibility, transparency, and incremental delivery, enabling the efficient execution of tasks. Challenges such as API rate limits and responsive design were addressed effectively, resulting in a functional and user-friendly platform. The project highlights the importance of continuous workflow management and provides a foundation for future enhancements, reaffirming the significance of Kanban in software development projects.