JavaScript course

Assignment 2/ Henna-Riina Anttila

**Summary of Thoughts to Tasks website**

**Key features:**

* Task creation: users can add tasks with a title, due date and priority level (low, medium, high).
* Visual Styling: each priority level has its own backround color to help organize tasks visually.
* Mark tasks as completed: tasks can be marked as done and moved to a separate completed list. They can also be undone.
* Delete tasks: tasks can be removed from the list with ”Delete” button.
* Filter tasks: task list can be filtered by priority.
* Drag and Drop: tasks can be rearranged using drag and drop functionality.
* Task Persistence: All tasks are saved to localStorage so they stay on the page even after refresing.
* Animated feedback: After adding a task a short message appears for two seconds using setTimeout() and fade animation.
* Insights & tools page: a second page includes a motivational quote (fetched from an API using a free proxy because the API does not allow direct access from the browser (CORS problem)), and a section for users to write and save their thoughts. I tested the page using Live Server in VS Code.
* Layout: the pages use CSS for layout including clear spacing, fonts and section styling.

**JavaScript Concepts Used**

* Control Structures: used in task validation, condition checks, and filtering.
* Functions: Reusable functions like task creation, validation, and button handling.
* Objects and Classes: A Task class is used to structure task data.
* Arrays: Used to manage task lists in memory and in storage.
* JSON: Used to store and retrieve data from localStorage.
* DOM Manipulation: The interface is updated dynamically when tasks are added, deleted, or marked as done.
* Event Handling: Used to respond to user actions like clicking buttons, submitting forms, and dragging tasks.
* Asynchronous JavaScript: the quote feature uses fetch() to load a new quote from an API.
* Currying and Higher-Order Functions: used to build flexible validators and dynamic button creation.
* React: used on the Insights & Tools page to create interactive components like QuoteFetcher and ThoughtDump with useState and useEffect. I tested the page using Live Server in VS Code. Babel was used to run JSX in the browser without extra tools.