Task: "Sorting Algorithm Visualizer"

Estimated Time: 3 Hours

Goal: Create a **React app** that **visualizes sorting algorithms** (Bubble Sort, Selection Sort, Merge Sort, Quick Sort).

Task Requirements

Generate a random array of numbers
Allow users to pick a sorting algorithm
Animate the sorting process step by step
Show sorting speed control
Display comparisons and swaps visually

How It Works (User Flow)

User clicks "Generate Array" → Creates a new random array of bars.
User selects a sorting algorithm (Bubble, Selection, Merge, Quick Sort).
User clicks "Start Sorting" → The algorithm sorts the array **visually step by step**.
Sorting bars change color to show swaps & comparisons.
Final sorted array is displayed with an animation.

Implementation Details

Array Generation

- Generate a **random array of numbers** (size 10-20).
- Represent numbers as **vertical bars** (height = value).
- Use **CSS animations** for smooth transitions.

Sorting Algorithms to Implement

Bubble Sort (Easiest)

- Compare adjacent elements, swap if necessary.
- **Visual:** Highlight bars being compared, color swaps.

Selection Sort

- Find the smallest element, swap it with the first unsorted position.
- **Visual:** Highlight the minimum and swapping positions.

Merge Sort (Divide & Conquer)

- Recursively divide the array, merge sorted halves.
- **Visual:** Show merging process in steps.

Quick Sort (Efficient)

- Select pivot, partition around pivot, recursively sort.
- **Visual:** Highlight pivot and moving elements.

Animation & Visualization

- **Use colors to show:**
    - Normal state
    - Comparing elements
    - Sorted elements
- **Use `setTimeout()` to slow down animation.**