

**Problem Statement:** This project addresses the need of an interactive and visually appealing platform to show application of various data structures (primarily queues) and algorithms (sorting and searching) in CPU scheduling. CPU scheduling algorithms gets difficult if learnt manually and application of data structures can not be understood in trivial way. The application bypasses this problem by providing a web application where any number of processes can be visualized for any algorithm along with analysis of various DSA aspects.

### **Objectives:**

- To develop an interactive web application covering 5 major CPU scheduling algorithms and plotting Gantt charts against input processes.
- Provide visual animations of how the processes are being entered, and executed with the help of queues.
- Providing comparative analysis between various scheduling algorithms based on theoretical time-space complexities and actual time taken.
- Provide flexible user interface by allowing user to manually add processes or upload csv file of various processes.
- Make application appealing by adding various colors and sound effects that makes the application engaging to use.

### **Methodology:**

- **Algorithm Implementation:** The 5 major CPU scheduling algorithms: FCFS, SJF, SRTF, Priority and Round Robin were implemented to be used along with queues.
- **Real-time development of visuals:** The gantt charts of the processes entered by the users were developed and rendered real time using CSS grid and flexbox.
- **Performance metrics calculation:** Algorithms were evaluated based on waiting time, total turn around time, CPU utilization and theoretical time complexities.
- **Adaptive scoring algorithm design:** For each set of processes, a score is calculated for each algorithm using 70% of the actual time used and 30% of the theoretical time complexity, thus providing a balanced assessment of the algorithm.
- **User interface and design:** All the features were combined with user interface to develop a fully functional application where user can add processes, select algorithms and can study various related evaluations and visualizations.

### **Results:**

- The application rendered visually appealing gantt charts for the set of process added/imported by the user
- Provided comparative analysis for various scheduling algorithms.
- Showed queue animation for the given set of processes displaying insertion, removal, and execution of the processes, along with operation logs.
- Overall the application provided educational information about application of Data Structures and algorithms in CPU scheduling with interactive visualizations.