

MySchedule System Design

Table of Contents

● CRC Cards	2-4
○ App.js	2
○ App.test.js	2
○ Calender.js	3
○ indes.js	3
○ reportWebVitals.js	4
○ setupTests.js	4
● Software Architecture Design	5

Class Name: App.js	
Parent Class: n/a Subclass: n/a	
Responsibilities: <ul style="list-style-type: none"> • Manage task state (add, edit, delete, toggle completion) • Handle sorting and filtering of tasks. • Request notification permission and trigger notifications for due tasks. • Render the task list and calendar view. 	Collaborators: <ul style="list-style-type: none"> • MyCalender (to display tasks in the calendar) (./Calendar) • Browser Notification API (axios) • React (react) • App.css (./App.css) (responsible for HTML display)

Class Name: App.test.js	
Parent Class: n/a Subclass: n/a	
Responsibilities: <ul style="list-style-type: none"> • Render the App component for testing • Verify that a specific element (text: "learn react") exists in the rendered output 	Collaborators: <ul style="list-style-type: none"> • App (renders and tests UI elements) • @testing-library/react (provides rendering and querying utilities)

Class Name: Calendar.js	
Parent Class: n/a Subclass: n/a	
Responsibilities: <ul style="list-style-type: none"> • Display tasks on a calendar. • Convert task due dates into event objects for rendering. • Allow users to navigate between dates and views (month, week, day). • Handle user interactions (e.g., clicking on an event shows an alert). 	Collaborators: <ul style="list-style-type: none"> • App (receives tasks as props). • react-big-calendar (used to display the calendar). • moment (handles date formatting and conversions).

Class Name: index.js	
Parent Class: n/a Subclass: n/a	
Responsibilities: <ul style="list-style-type: none"> • Initialize the React application. • Render the App component inside the root DOM element. • Wrap App in <React.StrictMode> is used to highlight potential problems in development. • Call reportWebVitals() to measure app performance (if configured). 	Collaborators: <ul style="list-style-type: none"> • App (renders the main UI). • ReactDOM (handles rendering). • reportWebVitals (optional performance monitoring).

Class Name: reportWebVitals.js	
Parent Class: n/a Subclass: n/a	
Responsibilities: <ul style="list-style-type: none"> • Measure and report web performance metrics. • Import the web-vitals library dynamically. • Call functions to measure Core Web Vitals (CLS, FID, FCP, LCP, TTFB). • Execute a callback function (onPerfEntry) with the performance data. 	Collaborators: <ul style="list-style-type: none"> • web-vitals (library for measuring performance). • index.js (calls this function to report performance metrics).

Class Name: setupTest.js	
Parent Class: n/a Subclass: n/a	
Responsibilities: <ul style="list-style-type: none"> • Extend Jest with custom matches to test DOM elements. • Enable assertions like .toHaveTextContent(). • Improve test readability and maintainability. 	Collaborators: <ul style="list-style-type: none"> • @testing-library/jest-dom (provides additional matches). • @testing-library/react (used for rendering components in tests).

Software Architecture Design Diagram

