

# Appointment Management System Documentation

## Overview

The **Appointment Management System** is a React application that allows users to create, edit, delete, and view appointments. It consists of three main components:

1. `AppointmentManagement` (Main Component)
  2. `AppointmentForm` (Form for adding appointments)
  3. `AppointmentList` (List of appointments with editing and deleting options)
- 

## 1. AppointmentManagement Component

File: `App.js`

### Description:

This is the main component that manages the state of appointments. It includes functions to add, delete, edit, and clear appointments.

### State Variables:

- `appointments` (Array) - Stores appointment objects with `name` and `date`.

### Functions:

- `addAppointment(appointment)`: Adds a new appointment to the list.
- `deleteAppointment(index)`: Removes an appointment from the list based on the index.
- `editAppointment(index, editedName, editedDate)`: Updates an existing appointment.
- `clearAppointments()`: Clears all stored appointments.

### Rendering:

- Displays the title "Appointment Management System".
  - Renders `AppointmentForm` to input new appointments.
  - Renders `AppointmentList` to display, edit, or delete appointments.
- 

## 2. AppointmentForm Component

File: `AppointmentForm.js`

### Description:

Handles the input form to add new appointments.

### State Variables:

- `name` (String) - Stores the appointment holder's name.
- `date` (String) - Stores the appointment date.

### Functions:

- `handleSubmit(e)`: Prevents default form submission and calls `addAppointment()` with name and date.

### Rendering:

- Displays input fields for name and date.
  - A submit button to add the appointment.
- 

## 3. AppointmentList Component

File: `AppointmentList.js`

### Description:

Displays the list of appointments with options to edit or delete each entry.

### State Variables:

- `editedIndex` (Number | Null) - Tracks the index of the appointment being edited.

- `editedName` (String) - Stores the edited name.
- `editedDate` (String) - Stores the edited date.

### Functions:

- `handleEdit(index)`: Sets the selected appointment for editing.
- `handleSaveEdit(index)`: Saves changes made to the appointment.
- `handleCancelEdit()`: Cancels the edit action.

### Rendering:

- Displays a table with ID, Name, Date, and Actions.
  - Provides Edit and Delete buttons for each entry.
  - Allows inline editing of name and date fields.
  - Optional button to clear all appointments (commented out in the code).
- 

## Technologies Used

- **React.js** for component-based UI development.
  - **CSS** for styling.
- 

## Future Improvements

- Add local storage to persist appointments.
  - Implement validation for input fields.
  - Improve UI/UX with better styling and alerts.
  - Enhance accessibility and mobile responsiveness.
- 

## Conclusion

The Appointment Management System provides a simple interface for handling appointments. With state management using React hooks, users can add, modify, and delete appointments efficiently.

**Git:** <https://github.com/brahmaputraS/react-project.git>

# Screenshot:

**Pro Research Solutions Appointment Management System**

Full Name:

Appointment Date:

### Appointment List

ID	Name	Date	Action
1	Jhon	2025-02-20	<input type="button" value="Edit"/> <input type="button" value="Delete"/>

