

Name:Ranjitha

Employee id:12103

Q2) Bug Management System

Required Properties are : Bug Id, Bug Name, Bug Description, Bug identifiedby, BugresolvedBy.

Note:

- Define a web application for the given scenario using React as a front end.
- Configure JSON server and consider JSON file as your database.
- Perform all the CRUD operations using React HttpClient Service.
- Perform the validations for all the fields.
- Application must be the Responsive Page (Including Mobile View) and Single Page Application (SPA).
- UI must be a Rich UI and look & Feel.
- Maintain the Code Quality .

Source Code

Index.js:

```
import React from 'react';

import ReactDOM from 'react-dom/client';
import './index.css';
// import App from './App'
import reportWebVitals from './reportWebVitals';
import Bugs from './Bugs';
import './style.css';

const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(
  <React.StrictMode>
    { /* <App /> */ }
    <Bugs/>
  </React.StrictMode>
);

reportWebVitals();
```

Bugs.js:

```
import React, { useState, useEffect } from "react";
import axios from "axios";
import BugsUpdate from "./BugsUpdate";
import "bootstrap/dist/css/bootstrap.min.css";
import "bootstrap/dist/js/bootstrap.bundle.min";

const Bugs = () => {
  const [Bugs, setBugs] = useState([]);
  const [newBug, setNewBug] = useState({
    Bugname: "",
    position: "",
    phone: "",
  });
  const [updatedBug, setUpdatedBug] = useState(null);

  useEffect(() => {
    fetchBugs();
  }, []);

  const fetchBugs = () => {
    axios
      .get("http://localhost:4000/employee")
      .then((response) => setBugs(response.data))
      .catch((error) => console.error("Error fetching data:", error));
  };

  const addBug = () => {
    axios
      .post("http://localhost:4000/employee", Bugs)
      .then(() => {
        fetchBugs();
        setNewBug({ BugName: "", position: "", phone: "" });
      })
      .catch((error) => console.error("Error adding bugs:", error));
  };

  const updateBug = (id, updatedData) => {
    axios
      .put(`http://localhost:4000/employee/${id}`, updatedData)
      .then(() => fetchBugs())
      .catch((error) => console.error("Error updating bugs:", error));
  };
};
```

```

const deleteBug = (id) => {
  axios
    .delete(`http://localhost:4000/employee/${id}`)
    .then(() => fetchBugs())
    .catch((error) => console.error("Error deleting bugs:", error));
};

const handleUpdateClick = (Bug) => {
  setUpdatedBug(Bug);
};

const handleUpdateSubmit = (updatedBug) => {
  updatedBug(updatedBug.id, updatedBug);
  setUpdatedBug(null);
};

return (
  <div className="container mt-4">
    <div className="container d-flex flex-column justify-content-center ">
      <h2>Bug Management System</h2>

      <div className="mb-3 justify-content-center ">
        <label className="col g-8 "> Bug Name:</label>
        <input
          className="form-control "
          type="text"
          value={newBug.BugName}
          onChange={(e) =>
            setNewBug({ ...newBug, BugName: e.target.value })
          }
        />
      </div>

      <div className="mb-3">
        <label>Position:</label>
        <input
          className="form-control "
          type="text"
          value={newBug.position}
          onChange={(e) =>
            setNewBug({ ...newBug, position: e.target.value })
          }
        />
      </div>
    </div>
  </div>

```

```

<div className="mb-3 ">
  <label>Phone:</label>
  <input
    className="form-control "
    type="text"
    value={newBug.phone}
    onChange={(e) =>
      setNewBug({ ...newBug, phone: e.target.value })
    }
  />
</div>

<button className="btn btn-primary mr-2" onClick={addBug}>
  Add Bugs
</button>
</div>

```

```

<table className="table mt-4">
  <thead>
    <tr>
      <th>ID</th>
      <th>Name</th>
      <th>Position</th>
      <th>phone</th>
      <th>Actions</th>
    </tr>
  </thead>
  <tbody>
    {Bugs.map((Bug) => (
      <tr key={Bug.id}>
        <td>{Bug.id}</td>
        <td>{Bug.name}</td>
        <td>{Bug.position}</td>
        <td>{Bug.phone}</td>

        <td>
          <button
            className="btn btn-warning "
            onClick={() => handleUpdateClick(Bug)}
          >
            Update
          </button>
          <button
            className="btn btn-danger ms-3"
            onClick={() => deleteBug(Bug.id)}

```

```

        >
        Delete
      </button>
    </td>
  </tr>
)}
</tbody>
</table>

    {updatedBug && (
      <BugsUpdate Bug={updateBug} onUpdate={handleUpdateSubmit} />
    )}
  </div>
);
};
export default Bugs;

```

BugsUpdate.js:

```

import React, { useState } from 'react'

const BugUpdate = ({ Bug, onUpdate }) => {
  const [updatedBug, setUpdatedBug] = useState(Bug)

  const handleInputChange = (e) => {
    const { name, value } = e.target
    setUpdatedBug((prevState) => ({ ...prevState, [name]: value }))
  }

  const handleUpdate = () => {
    onUpdate(updatedBug)
  }

  return (
    <div>
      <input
        type="text"
        name="name"
        value={updatedBug.name}
        onChange={handleInputChange}
      />
      <input
        type="text"
        name="position"
        value={updatedBug.position}

```

```
        onChange={handleInputChange}
      />

    <input
      type="text"
      name="phone"
      value={updatedBugphone}
      onChange={handleInputChange}
    />
    <button className="btn btn-success" onClick={handleUpdate}>
      submit
    </button>
  </>
)
}

export default BugsUpdate
```

Style.css:

```
body{
  background-color:lightgreen;

}
```

Output:

1) Bug management system

Bugs Management System

Bug Name:

Bug Description:

Bug IdentifiedBy&ResolvedBy :

Add Bugs

ID	Bug Name	Bug Description	Bug IdentifiedBy&ResolvedBy	Actions
6085	logic	Disrupts the intended workflow of software and causes it to behave incorrectly	Ranjani	<div>Update</div> <div>Delete</div>

2)Server page

```
1 [
2   {
3     "name": "logic",
4     "position": "Disrupts the intended workflow of software and causes it to behave incorrectly",
5     "phone": "Ran",
6     "id": "6085",
7     "phone": "Ranjani"
8   }
9 ]
```

3)Adding Datas

Bugs Management System

Bug Name:

Error

Bug Description:

Functional error

Bug IdentifiedBy&ResolvedBy :

Ram

Add Bugs

ID	Bug Name	Bug Description	Bug IdentifiedBy&ResolvedBy	Actions
6085	logic	Disrupts the intended workflow of software and causes it to behave incorrectly	Ranjani	<div>Update</div> <div>Delete</div>

4)After adding the Data

Bugs Management System

Bug Name:

Bug Description:

Bug IdentifiedBy&ResolvedBy :

Add Bugs

ID	Bug Name	Bug Description	Bug IdentifiedBy&ResolvedBy	Actions
6085	logic	Disrupts the intended workflow of software and causes it to behave incorrectly	Ranjani	<div>Update</div> <div>Delete</div>
0697	Error	Functional error	Ram	<div>Update</div> <div>Delete</div>

5)Server Data

```
1 [
2   {
3     "name": "logic",
4     "position": "Disrupts the intended workflow of software and causes it to behave incorrectly",
5     "phone": "Ram",
6     "id": "6085",
7     "phone": "Ranjani"
8   },
9   {
10    "id": "0697",
11    "name": "Error",
12    "position": "Functional error",
13    "phone": "Ram"
14  }
15 ]
```

6)Updation

Bugs Management System

Bug Name:

Bug Description:

Bug IdentifiedBy&ResolvedBy :

Add Bugs

Bug ID	Bug Name	Bug Description	Bug IdentifiedBy&ResolvedBy	Actions
6085	logic	Disrupts the intended workflow of software and causes it to behave incorrectly	Ranjani	<div>Update</div> <div>Delete</div>
0697	Error	Functional error	Ram	<div>Update</div> <div>Delete</div>

Error

Functional error

Ram

submit

7)After updation

Bugs Management System

Bug Name:

Bug Description:

Bug IdentifiedBy&ResolvedBy :

Add Bugs

ID	Bug Name	Bug Description	Bug IdentifiedBy&ResolvedBy	Actions
6085	logic	Disrupts the intended workflow of software and causes it to behave incorrectly	Ranjani	<div>Update</div> <div>Delete</div>
0697	Error	Functional error	Raj	<div>Update</div> <div>Delete</div>

8)Before delete the data:

Bug Name:

Bug Description:

Bug IdentifiedBy&ResolvedBy :

Add Bugs

ID	Bug Name	Bug Description	Bug IdentifiedBy&ResolvedBy	Actions
6085	logic	Disrupts the intended workflow of software and causes it to behave incorrectly	Ranjani	<div>Update</div> <div>Delete</div>
0697	Error	Functional error	Raj	<div>Update</div> <div>Delete</div>
be11	Control Flow error	System dependability during component interactions	Vinolisha	<div>Update</div> <div>Delete</div>

9)After delete the Data

Bugs Management System

Bug Name:

Bug Description:

Bug IdentifiedBy&ResolvedBy :

Add Bugs

ID	Bug Name	Bug Description	Bug IdentifiedBy&ResolvedBy	Actions
6085	logic	Disrupts the intended workflow of software and causes it to behave incorrectly	Ranjani	<button>Update</button> <button>Delete</button>
be11	Control Flow error	System dependability during component interactions	Vinolisha	<button>Update</button> <button>Delete</button>

10)Server page:

```
1 [{
2   {
3     "name": "logic",
4     "position": "Disrupts the intended workflow of software and causes it to behave incorrectly",
5     "phone": "Ran",
6     "id": "6085",
7     "phone": "Ranjani"
8   },
9   {
10    "id": "be11",
11    "name": "Control Flow error",
12    "position": "System dependability during component interactions",
13    "phone": "Vinolisha"
14  }
15 ]
```

