Inventory Management System

 Inventory.jsx

import React, { useState } from 'react';

const Inventory = ({ products, onAddProduct, onUpdateStock }) => {

const [newProduct, setNewProduct] = useState({ name: '', quantity: 0 });

const handleAddProduct = (e) => {

e.preventDefault();

if (newProduct.name && newProduct.quantity > 0) {

onAddProduct({ ...newProduct });

setNewProduct({ name: '', quantity: 0 });

}

};

const handleUpdateStock = (index, increment) => {

onUpdateStock(index, increment);

};

return (

<div>

<form onSubmit={handleAddProduct}>

<input

type="text"

placeholder="Product Name"

value={newProduct.name}

onChange={(e) => setNewProduct({ ...newProduct, name: e.target.value })}

required

/>

<input

type="number"

placeholder="Quantity"

value={newProduct.quantity}

onChange={(e) => setNewProduct({ ...newProduct, quantity: parseInt(e.target.value) })}

required

/>

<button type="submit">Add Product</button>

</form>

<h2>Current Inventory</h2>

<ul>

{products.map((product, index) => (

<li key={index}>

{product.name}: {product.quantity}

<button onClick={() => handleUpdateStock(index, 1)}>Increase</button>

<button onClick={() => handleUpdateStock(index, -1)}>Decrease</button>

</li>

))}

</ul>

</div>

);

};

export default Inventory;

app.jsx

import React, { useState } from 'react';

import Inventory from './Inventory';

const App = () => {

const [products, setProducts] = useState([]);

const addProduct = (product) => {

setProducts((prevProducts) => [...prevProducts, product]);

};

const updateStock = (index, quantity) => {

const updatedProducts = [...products];

updatedProducts[index].quantity += quantity;

setProducts(updatedProducts);

};

return (

<div className="App">

<h1>Inventory Management System</h1>

<Inventory

products={products}

onAddProduct={addProduct}

onUpdateStock={updateStock}

/>

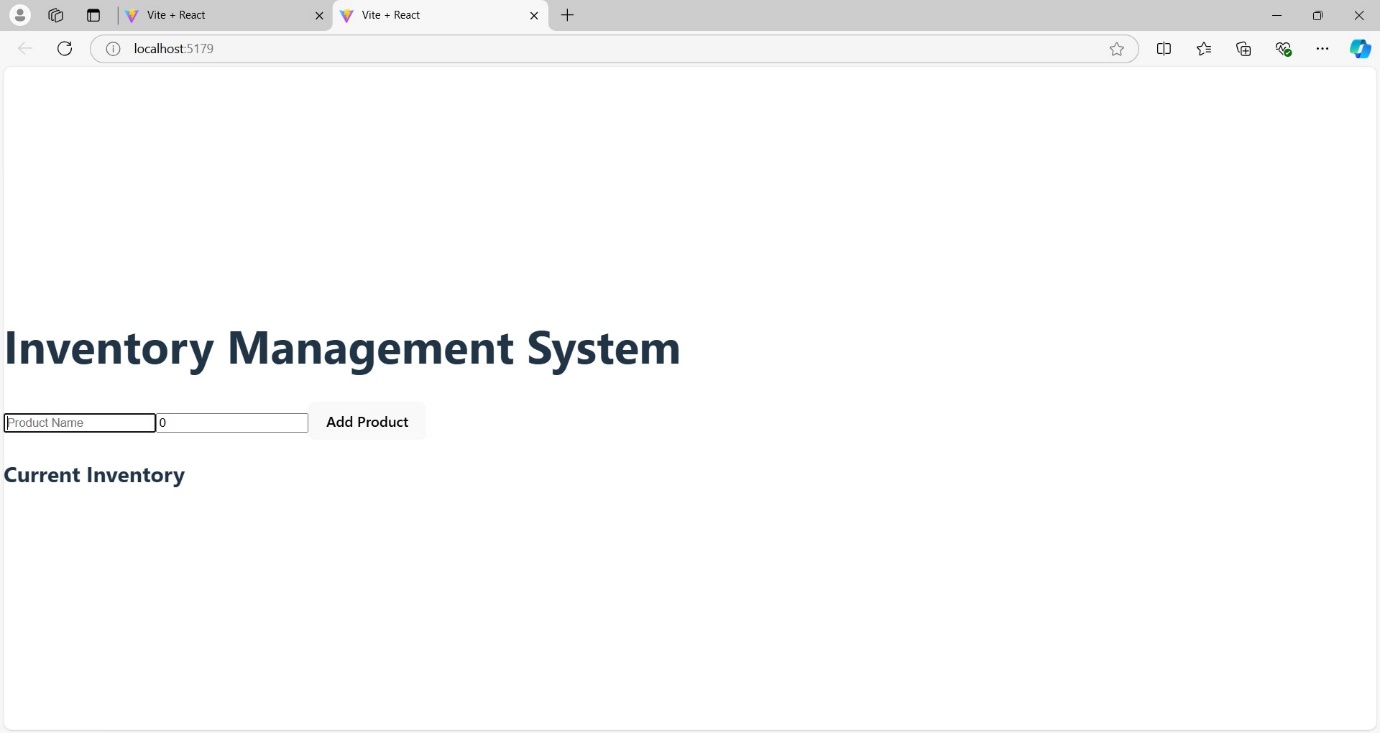
</div>

);

};

export default App;

home screen



After Adding new products with a specified quantity, you can View the current inventory of products.

And also Increase or decrease the stock level for each product.