HTML

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Persistent Grocery List with Search</title>
    <link rel="stylesheet" href="Grocery List Maker.css">
</head>
<body>
    <div class="container">
        <h1>Grocery List Maker</h1>

        <!-- Search Panel -->
        <div class="search-area">
            <input type="text" id="search" placeholder="Search items..."
oninput="searchItems()">
        </div>

        <!-- Input Form -->
        <div class="input-area">
            <input type="text" id="item-name" placeholder="Item name...">
            <input type="number" id="item-sale-price" placeholder="Sale Price..." min="0"
step="0.01">
            <input type="number" id="item-sell-price" placeholder="Sell Price..." min="0"
step="0.01">
            <input type="number" id="item-stock" placeholder="Stock available..."
min="0">
            <input type="text" id="item-weight" placeholder="Weight (e.g., 1kg,
500g)...">
            <button onclick="addItem()">Add</button>
        </div>

        <ul id="grocery-list"></ul>
    </div>
</body>
</html>
```

CSS

```css
body {
    font-family: 'Arial', sans-serif;
    margin: 0;
    padding: 0;
    background: linear-gradient(to left,  #6771fd,#1c7e1f);
}
.container {
    max-width: 600px;
    margin: 50px auto;
    padding: 20px;
    background: #fff;
    border-radius: 10px;
    box-shadow: 0 4px 10px rgba(0, 0, 0, 0.1);
}
h1 {
    text-align: center;
    background: linear-gradient(to left, #1c7e1f, #6771fd);
    background-clip: text;
    color: transparent;
}
.input-area,
.search-area {
    display: flex;
    flex-direction: column;
    gap: 10px;
```

```css
        margin-bottom: 20px;
}
.input-area input,
.search-area input,
.input-area button {
    padding: 10px;
    border-radius: 5px;
    border: 1px solid #ddd;
}
.input-area button {
    background-color: #0284c7;
    color: white;
    cursor: pointer;
}
.input-area button:hover {
    background-color: #15803d;
}
ul {
    list-style: none;
    padding: 0;
}
ul li {
    display: flex;
    justify-content: space-between;
    align-items: center;
    margin-bottom: 10px;
    padding: 10px;
    background: #f9f9f9;
    border-radius: 5px;
    border: 1px solid #ddd;
}
.delete-button, .save-button {
    background-color: #0284c7;
    border: none;
    color: white;
    padding: 5px 10px;
    cursor: pointer;
    border-radius: 5px;
    margin-left: 5px;
}
.delete-button:hover {
    background-color: #b91c1c;
}
.save-button {
    background-color: #0284c7;
}
.save-button:hover {
    background-color: #15803d;
}
```

Script

```html
    <script>
        // Load items from localStorage when the page loads
        document.addEventListener('DOMContentLoaded', () => {
            loadItems();
        });

        function addItem() {
            const itemName = document.getElementById('item-name').value.trim();
            const itemSalePrice = parseFloat(document.getElementById('item-sale-price').value.trim());
            const itemSellPrice = parseFloat(document.getElementById('item-sell-price').value.trim());
            const itemStock = parseInt(document.getElementById('item-stock').value.trim());
```

```javascript
            const itemWeight = document.getElementById('item-weight').value.trim();

            if (!itemName || isNaN(itemSalePrice) || isNaN(itemSellPrice) ||
isNaN(itemStock) || !itemWeight) {
                alert('Please fill out all fields correctly.');
                return;
            }

            const newItem = {
                name: itemName,
                salePrice: itemSalePrice,
                sellPrice: itemSellPrice,
                stock: itemStock,
                weight: itemWeight
            };

            addToDOM(newItem);
            clearInputs();
        }

        function clearInputs() {
            document.getElementById('item-name').value = '';
            document.getElementById('item-sale-price').value = '';
            document.getElementById('item-sell-price').value = '';
            document.getElementById('item-stock').value = '';
            document.getElementById('item-weight').value = '';
        }

        function addToDOM(item, isSaved = false) {
            const groceryList = document.getElementById('grocery-list');

            const listItem = document.createElement('li');

            const details = document.createElement('div');
            details.innerHTML = `
                <strong>${item.name}</strong><br>
                Sale Price: $${item.salePrice.toFixed(2)}, Sell Price:
$${item.sellPrice.toFixed(2)}<br>
                Stock: ${item.stock}, Weight: ${item.weight}
            `;

            const saveButton = document.createElement('button');
            saveButton.textContent = 'Save';
            saveButton.className = 'save-button';
            saveButton.onclick = function () {
                saveItem(item);
                saveButton.style.display = 'none'; // Hide save button for saved items
            };

            const deleteButton = document.createElement('button');
            deleteButton.textContent = 'Delete';
            deleteButton.className = 'delete-button';
            deleteButton.onclick = function () {
                deleteItem(item);
                listItem.remove();
            };

            listItem.appendChild(details);

            if (!isSaved) {
                listItem.appendChild(saveButton); // Only show save button for new
(unsaved) items
            }
            listItem.appendChild(deleteButton);
```

```javascript
            listItem.dataset.name = item.name.toLowerCase(); // Add a searchable dataset
attribute
            groceryList.appendChild(listItem);
        }

        function saveItem(item) {
            const savedItems = JSON.parse(localStorage.getItem('groceryItems')) || [];
            savedItems.push(item);
            localStorage.setItem('groceryItems', JSON.stringify(savedItems));
            alert('Item saved!');
        }

        function deleteItem(item) {
            let savedItems = JSON.parse(localStorage.getItem('groceryItems')) || [];
            savedItems = savedItems.filter(savedItem => savedItem.name !== item.name);
            localStorage.setItem('groceryItems', JSON.stringify(savedItems));
        }

        function loadItems() {
            const savedItems = JSON.parse(localStorage.getItem('groceryItems')) || [];
            savedItems.forEach(item => {
                addToDOM(item, true);
            });
        }

        function searchItems() {
            const query = document.getElementById('search').value.toLowerCase();
            const listItems = document.querySelectorAll('#grocery-list li');

            listItems.forEach(item => {
                const itemName = item.dataset.name;
                if (itemName.includes(query)) {
                    item.style.display = 'flex';
                } else {
                    item.style.display = 'none';
                }
            });
        }
    </script>
```