CODE

```
HTML (index.html)
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Personal Bucket List</title>
  <link rel="stylesheet" href="styles.css">
</head>
<body>
  <header>
    <nav>
     ul>
       <a href="#">Home</a>
       <a href="#">Login</a>
       <a href="#">Register</a>
     </nav>
  </header>
  <main>
    <h1>My Bucket List</h1>
    <!-- List items will be generated dynamically -->
    <form id="add-item-form">
```

```
<input type="text" id="item-name" placeholder="Add new item">
      <button id="add-item-btn">Add</button>
    </form>
  </main>
  <script src="script.js"></script>
</body>
</html>
CSS (styles.css)
body {
  font-family: Arial, sans-serif;
  margin: 0;
  padding: 0;
}
header {
  background-color: #333;
  color: #fff;
  padding: 1em;
  text-align: center;
}
nav ul {
  list-style: none;
  margin: 0;
  padding: 0;
  display: flex;
```

```
justify-content: space-between;
}
nav li {
  margin-right: 20px;
}
nav a {
  color: #fff;
  text-decoration: none;
}
main {
  display: flex;
  flex-direction: column;
  align-items: center;
  padding: 2em;
}
#bucket-list {
  list-style: none;
  margin: 0;
  padding: 0;
}
#bucket-list li {
  margin-bottom: 10px;
}
```

```
#add-item-form {
  margin-top: 20px;
}
#add-item-btn {
  background-color: #333;
  color: #fff;
  border: none;
  padding: 10px 20px;
  font-size: 16px;
  cursor: pointer;
}
#add-item-btn:hover {
  background-color: #555;
}
JavaScript (script.js)
// Get the bucket list element
const bucketListElement = document.getElementById('bucket-list');
// Get the add item form element
const addFormItemElement = document.getElementById('add-item-form');
// Get the add item button element
const add ItemBtnElement = document.getElementById('add-item-btn');
```

```
// Define a function to render the bucket list
function renderBucketList(bucketList) {
  // Clear the existing list items
  bucketListElement.innerHTML = ";
  // Loop through the bucket list items and create list items
  bucketList.forEach((item) => {
    const listItemElement = document.createElement('LI');
    listItemElement.textContent = item.name;
    bucketListElement.appendChild(listItemElement);
  });
}
// Define a function to handle adding a new item
function handleAddItem(event) {
  // Prevent the default form submission behavior
  event.preventDefault();
  // Get the item name from the input field
  const itemName = document.getElementById('item-name').value;
  // Create a new item object
  const newItem = { name: itemName };
  // Add the new item to the bucket list
  bucketList.push(newItem);
  // Render the updated bucket list
  renderBucketList(bucketList);
```

```
// Clear the input field
        document.getElementById('item-name').value = ";
}
// Add an event listener to the add item button
add ItemBtnElement.addEventListener('click', handleAddItem);
// Initialize the bucket list
const bucketList = [];
// Render the initial bucket list
renderBucketList(bucketList);
Back-end (Server-side)
Node.js (server.js)
const express = require('express');
const app = express();
const mongoose = require('mongoose');
// Connect to the MongoDB database
mongoose.connect ('mongodb://localhost/bucket-list', \{ use New Url Parser: true, use Unified Topology: use Unified Topology: true, use Unified Topology: true, use Unifi
true });
// Define a schema for the bucket list items
const itemSchema = new mongoose.Schema({
        name: String
```

```
});

// Create a model for the bucket list items

const Item = mongoose.model('Item', itemSchema);

// Define a route to get all bucket list items

app.get('/api/items', async (req, res) => {
    const items = await Item.find();
    res.json(items);

});

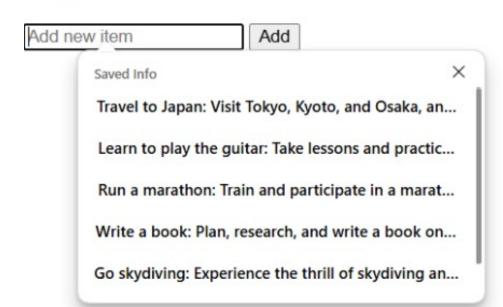
// Define a route to add a new bucket list item

App
```

OUTPUT

- Home
- Login
- Register

My Bucket List



- Home
- Login
- Register

My Bucket List

