Tworzenie aplikacji do zarządznia fakturami z czatem AI

GPT

let invoices = [];

function loadInvoices() {

    const invoicesData = localStorage.getItem('invoices');

    if (invoicesData) {

        invoices = JSON.parse(invoicesData);

    }

}

function saveInvoices() {

    localStorage.setItem('invoices', JSON.stringify(invoices));

}

function displayInvoices() {

    const invoicesList = document.getElementById('invoicesList');

    invoicesList.innerHTML = '';

    invoices.forEach(invoice => {

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

        const recurring = invoice.recurring ? 'Tak' : 'Nie';

        listItem.textContent = `${invoice.recipient} - Kwota: ${invoice.amount} zł - Termin: ${invoice.dueDate} - Cykliczna: ${recurring}`;

        const dueDate = new Date(invoice.dueDate);

        const reminderDate = new Date(dueDate.getTime() - 7 \* 24 \* 60 \* 60 \* 1000);

        const reminderMessage = `Przypomnienie: Płatność faktury ${invoice.recipient} (${invoice.amount} zł) jest za tydzień!`;

        const reminder = {

            date: reminderDate.toISOString().slice(0, 10),

            message: reminderMessage

        };

        if (!invoice.reminders) {

            invoice.reminders = [];

        }

        invoice.reminders.push(reminder);

        const deleteButton = document.createElement('button');

        deleteButton.textContent = 'Usuń';

        deleteButton.addEventListener('click', () => deleteInvoice(invoice.id));

        listItem.appendChild(deleteButton);

        invoicesList.appendChild(listItem);

    });

    saveInvoices();

}

function deleteInvoice(invoiceId) {

    invoices = invoices.filter(invoice => invoice.id !== invoiceId);

    saveInvoices();

    displayInvoices();

}

function addInvoice() {

    const recipient = document.getElementById('recipientInput').value;

    const amount = parseFloat(document.getElementById('amountInput').value);

    const dueDate = document.getElementById('dueDateInput').value;

    const recurring = document.getElementById('recurringInput').checked;

    const newId = invoices.length > 0 ? invoices[invoices.length - 1].id + 1 : 1;

    const newInvoice = {

        id: newId,

        recipient: recipient,

        amount: amount,

        dueDate: dueDate,

        recurring: recurring,

        reminders: []

    };

    invoices.push(newInvoice);

    displayInvoices();

    document.getElementById('recipientInput').value = '';

    document.getElementById('amountInput').value = '';

    document.getElementById('dueDateInput').value = '';

    document.getElementById('recurringInput').checked = false;

}

function checkReminders() {

    invoices.forEach(invoice => {

        if (invoice.reminders) {

            invoice.reminders.forEach(reminder => {

                const today = new Date().toISOString().slice(0, 10);

                if (reminder.date === today) {

                    alert(reminder.message);

                }

            });

        }

    });

}

document.addEventListener('DOMContentLoaded', () => {

    loadInvoices();

    displayInvoices();

    checkReminders();

    const addButton = document.getElementById('addButton');

    addButton.addEventListener('click', addInvoice);

});

<!DOCTYPE html>

<html lang="pl">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Zarządzanie fakturami</title>

    <style>

        body {

            font-family: Arial, sans-serif;

            padding: 20px;

            max-width: 600px;

            margin: 0 auto;

        }

        h1 {

            text-align: center;

            color: #333;

        }

        label {

            display: block;

            margin-bottom: 5px;

            font-weight: bold;

        }

        input[type="text"],

        input[type="number"],

        input[type="date"],

        input[type="checkbox"],

        button {

            width: 100%;

            padding: 10px;

            margin-bottom: 15px;

            border: 1px solid #ccc;

            border-radius: 4px;

            box-sizing: border-box;

            font-size: 16px;

        }

        button {

            background-color: #007bff;

            color: #fff;

            cursor: pointer;

            transition: background-color 0.3s ease;

        }

        button:hover {

            background-color: #0056b3;

        }

        ul {

            list-style-type: none;

            padding: 0;

        }

        li {

            border: 1px solid #ddd;

            border-radius: 4px;

            padding: 10px;

            margin-bottom: 10px;

        }

        li button {

            background-color: #dc3545;

            color: #fff;

            border: none;

            padding: 8px 12px;

            border-radius: 4px;

            cursor: pointer;

            float: right;

        }

        li button:hover {

            background-color: #c82333;

        }

    </style>

</head>

<body>

    <h1>Zarządzanie fakturami</h1>

    <div>

        <label for="recipientInput">Odbiorca:</label>

        <input type="text" id="recipientInput">

    </div>

    <div>

        <label for="amountInput">Kwota (zł):</label>

        <input type="number" id="amountInput">

    </div>

    <div>

        <label for="dueDateInput">Termin płatności:</label>

        <input type="date" id="dueDateInput">

    </div>

    <div>

        <label for="recurringInput">Cykliczna:</label>

        <input type="checkbox" id="recurringInput">

    </div>

    <button id="addButton">Dodaj fakturę</button>

    <ul id="invoicesList"></ul>

    <script src="app.js">

    </script>

</body>

</html>

Gpt poradził sobie dobrze dodał styl w razie potrzeby zmieniał i dodawał rzeczy

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Gemini odmówił posluszeństwa

A screen shot of a computer

Description automatically generated