import React, { useState } from "react";

import jsPDF from "jspdf";

import html2canvas from "html2canvas";

// Application de Paie Marocaine avec export PDF du bulletin

export default function MoroccoPayrollApp() {

const [employee, setEmployee] = useState({

nom: "Mohamed",

poste: "Employé",

salaireBrut: 6000,

primes: 0,

retenuesAutres: 0,

});

const [params, setParams] = useState({

cnss\_employee\_pct: 4.29,

cnss\_employer\_pct: 6.74,

ir\_flat\_pct: 0,

taxBrackets: [

{ upTo: 3333.3333, pct: 0 },

{ upTo: 5000, pct: 10 },

{ upTo: 6666.6667, pct: 20 },

{ upTo: 8333.3333, pct: 30 },

{ upTo: 15000, pct: 34 },

{ upTo: Infinity, pct: 37 },

],

cimr\_employee\_pct: 4,

otherEmployerCostsPct: 2,

});

function computeIR(taxable, brackets) {

if (params.ir\_flat\_pct > 0) {

return (taxable \* params.ir\_flat\_pct) / 100;

}

let remaining = taxable;

let tax = 0;

let lower = 0;

for (const b of brackets) {

const upper = b.upTo;

const base = Math.min(remaining, upper - lower);

if (base > 0) {

tax += (base \* b.pct) / 100;

remaining -= base;

}

lower = upper;

if (remaining <= 0) break;

}

return Math.max(0, tax);

}

const brut = Number(employee.salaireBrut) || 0;

const primes = Number(employee.primes) || 0;

const salaireAssiette = brut + primes;

const cnssEmployee = (salaireAssiette \* params.cnss\_employee\_pct) / 100;

const cimrEmployee = (salaireAssiette \* params.cimr\_employee\_pct) / 100;

const totalEmployeeContrib = cnssEmployee + cimrEmployee + Number(employee.retenuesAutres || 0);

const salaireImposable = Math.max(0, salaireAssiette - (cnssEmployee + cimrEmployee));

const ir = computeIR(salaireImposable, params.taxBrackets);

const netAPayer = Math.max(0, salaireAssiette - totalEmployeeContrib - ir);

const cnssEmployer = (salaireAssiette \* params.cnss\_employer\_pct) / 100;

const otherEmployerCosts = (salaireAssiette \* params.otherEmployerCostsPct) / 100;

const coutEmployeur = salaireAssiette + cnssEmployer + otherEmployerCosts;

// Export PDF avec jsPDF + html2canvas

const exportPDF = () => {

const input = document.getElementById("bulletin-paie");

html2canvas(input).then((canvas) => {

const imgData = canvas.toDataURL("image/png");

const pdf = new jsPDF("p", "mm", "a4");

const imgProps = pdf.getImageProperties(imgData);

const pdfWidth = pdf.internal.pageSize.getWidth();

const pdfHeight = (imgProps.height \* pdfWidth) / imgProps.width;

pdf.addImage(imgData, "PNG", 0, 0, pdfWidth, pdfHeight);

pdf.save(`Bulletin\_${employee.nom}.pdf`);

});

};

return (

<div className="p-6 max-w-4xl mx-auto">

<h1 className="text-2xl font-bold mb-4">Application de Paie - Maroc</h1>

<section className="mb-6 p-4 border rounded-lg">

<h2 className="font-semibold mb-2">Informations employé</h2>

<div className="grid grid-cols-2 gap-3">

<input className="p-2 border rounded" value={employee.nom} onChange={(e) => setEmployee({...employee, nom: e.target.value})} placeholder="Nom" />

<input className="p-2 border rounded" value={employee.poste} onChange={(e) => setEmployee({...employee, poste: e.target.value})} placeholder="Poste" />

<input type="number" className="p-2 border rounded" value={employee.salaireBrut} onChange={(e) => setEmployee({...employee, salaireBrut: e.target.value})} placeholder="Salaire brut" />

<input type="number" className="p-2 border rounded" value={employee.primes} onChange={(e) => setEmployee({...employee, primes: e.target.value})} placeholder="Primes / Avantages" />

<input type="number" className="p-2 border rounded" value={employee.retenuesAutres} onChange={(e) => setEmployee({...employee, retenuesAutres: e.target.value})} placeholder="Retenues autres" />

</div>

</section>

<div id="bulletin-paie" className="mb-6 p-4 border rounded-lg bg-white">

<h2 className="font-semibold mb-2">Bulletin de Paie</h2>

<p><strong>Employé :</strong> {employee.nom} ({employee.poste})</p>

<p><strong>Salaire brut :</strong> {brut.toFixed(2)} MAD</p>

<p><strong>Primes :</strong> {primes.toFixed(2)} MAD</p>

<p><strong>Assiette :</strong> {salaireAssiette.toFixed(2)} MAD</p>

<p><strong>CNSS Employé :</strong> {cnssEmployee.toFixed(2)} MAD</p>

<p><strong>CIMR Employé :</strong> {cimrEmployee.toFixed(2)} MAD</p>

<p><strong>IR :</strong> {ir.toFixed(2)} MAD</p>

<p><strong>Retenues autres :</strong> {employee.retenuesAutres} MAD</p>

<p className="text-lg font-bold mt-2">Net à payer : {netAPayer.toFixed(2)} MAD</p>

<hr className="my-2" />

<p><strong>CNSS Employeur :</strong> {cnssEmployer.toFixed(2)} MAD</p>

<p><strong>Autres coûts employeur :</strong> {otherEmployerCosts.toFixed(2)} MAD</p>

<p className="text-lg font-bold">Coût total employeur : {coutEmployeur.toFixed(2)} MAD</p>

</div>

<button onClick={exportPDF} className="px-4 py-2 rounded bg-green-600 text-white">Exporter en PDF</button>

</div>

);

}