**MAIN**

import React from "react";

import { Toaster } from "@/components/ui/toaster";

import { Toaster as Sonner } from "@/components/ui/sonner";

import { TooltipProvider } from "@/components/ui/tooltip";

import { QueryClient, QueryClientProvider } from "@tanstack/react-query";

import { BrowserRouter, Routes, Route } from "react-router-dom";

import { ThemeProvider } from "@/components/ThemeProvider";

import { AccessibilityProvider } from "@/components/AccessibilityContext";

import Index from "@/pages/Index";

import UserLogin from "@/pages/UserLogin";

import AdminLogin from "@/pages/AdminLogin";

import UserDashboard from "@/pages/UserDashboard";

import AdminDashboard from "@/pages/AdminDashboard";

import NotFound from "@/pages/NotFound";

import AccessibleUserDashboard from "@/pages/AccessibleUserDashboard";

import UserManualPage from "@/pages/UserManualPage";

const queryClient = new QueryClient();

const App = () => (

  <QueryClientProvider client={queryClient}>

    <ThemeProvider defaultTheme="light" storageKey="apexwolf-theme">

      <AccessibilityProvider>

        <TooltipProvider>

          <Toaster />

          <Sonner />

          <BrowserRouter>

            <Routes>

              <Route path="/" element={<Index />} />

              <Route path="/login/user" element={<UserLogin />} />

              <Route path="/login/admin" element={<AdminLogin />} />

              <Route path="/user/dashboard" element={<UserDashboard />} />

              <Route path="/user/accessible" element={<AccessibleUserDashboard />} />

              <Route path="/admin/dashboard" element={<AdminDashboard />} />

              <Route path="/user/user-manual" element={<UserManualPage />} />

              <Route path="\*" element={<NotFound />} />

            </Routes>

          </BrowserRouter>

        </TooltipProvider>

      </AccessibilityProvider>

    </ThemeProvider>

  </QueryClientProvider>

);

export default App;

**ACCESSIBLE MODE CONTEXT**

// src/components/AccessibilityContext.tsx

import React, { createContext, useContext, useState } from "react";

// Define the shape of the accessibility settings context

type AccessibilitySettings = {

  largeText: boolean;

  setLargeText: (v: boolean) => void;

  dyslexiaFont: boolean;

  setDyslexiaFont: (v: boolean) => void;

  highContrast: boolean;

  setHighContrast: (v: boolean) => void;

  colorBlindSafe: boolean;

  setColorBlindSafe: (v: boolean) => void;

  sunlightMode: boolean;

  setSunlightMode: (v: boolean) => void;

};

// Create the context, defaulting to undefined

export const AccessibilityContext = createContext<AccessibilitySettings | undefined>(undefined);

// Hook for consuming the accessibility context with safety check

export const useAccessibility = () => {

  const ctx = useContext(AccessibilityContext);

  if (!ctx) {

    throw new Error("useAccessibility must be used within an AccessibilityProvider");

  }

  return ctx;

};

// AccessibilityProvider component to wrap your app

export const AccessibilityProvider: React.FC<{ children: React.ReactNode }> = ({ children }) => {

  // States for various accessibility options

  const [largeText, setLargeText] = useState(true);

  const [dyslexiaFont, setDyslexiaFont] = useState(false);

  // Only one of these three modes can be enabled at once

  const [highContrast, setHighContrastState] = useState(false);

  const [colorBlindSafe, setColorBlindSafeState] = useState(false);

  const [sunlightMode, setSunlightModeState] = useState(false);

  // Mutually exclusive setters for modes

  const setHighContrast = (v: boolean) => {

    setHighContrastState(v);

    if (v) {

      setColorBlindSafeState(false);

      setSunlightModeState(false);

    }

  };

  const setColorBlindSafe = (v: boolean) => {

    setColorBlindSafeState(v);

    if (v) {

      setHighContrastState(false);

      setSunlightModeState(false);

    }

  };

  const setSunlightMode = (v: boolean) => {

    setSunlightModeState(v);

    if (v) {

      setHighContrastState(false);

      setColorBlindSafeState(false);

    }

  };

  return (

    <AccessibilityContext.Provider

      value={{

        largeText,

        setLargeText,

        dyslexiaFont,

        setDyslexiaFont,

        highContrast,

        setHighContrast,

        colorBlindSafe,

        setColorBlindSafe,

        sunlightMode,

        setSunlightMode,

      }}

    >

      {children}

    </AccessibilityContext.Provider>

  );

};

**ACCESSIBLE DASHBOARD**

import React, { useRef, useState } from "react";

import { useNavigate } from "react-router-dom";

import {

  Brain, User, FileText, BarChart3, MessageSquare, TrendingUp, Calculator,

  Trophy, Building2, Bell, Receipt, Shield, HelpCircle, Globe, Smartphone,

  LogOut, UploadCloud, Volume2

} from "lucide-react";

import { Button } from "@/components/ui/button";

import { useAccessibility } from "@/components/AccessibilityContext";

// Import all your Accessible components here...

import AccessibleLoanSimulator from "@/components/AccessibleLoanSimulator";

import AccessibleLoanPrediction from "@/components/AccessibleLoanPrediction";

import AccessibleBiometricForm from "@/components/AccessibleBiometric";

import AccessibleChatbot from "@/components/AccessibleChatbot";

import AccessibleSmartInvestment from "@/components/AccessibleSmartInvestment";

import AccessibleExpenseTracker from "@/components/AccessibleExpenseTracker";

import AccessibleFeedback from "@/components/AccessibleFeedback";

import AccessibleMarketTrends from "@/components/AccessibleMarketTrends";

import AccessibleNegotiation from "@/components/AccessibleNegotiation";

import AccessibleNotification from "@/components/AccessibleNotification";

import AccessibleRewardsHub from "@/components/AccessibleRewardsHub";

import AccessibleSettings from "@/components/AccessibleSettings";

import AccessibleCustomerSupport from "@/components/AccessibleCustomerSupport";

// --- ACCESSIBLE PALETTE + LOGO CONTRAST ---

function getPalette({ sunlightMode, highContrast, colorBlindSafe }) {

  if (sunlightMode) {

    return {

      background: "#FFF",

      foreground: "#111",

      border: "#FFC107",

      accent: "#FFC107",

      headerBg: "#FFF",

      headerText: "#111",

      sidebar: "#FFF",

      sidebarText: "#111",

      sidebarActive: "#FFC107",

      sidebarActiveText: "#111",

      cardBg: "#FFFDE7",

      buttonBg: "#FFC107",

      buttonText: "#111",

      buttonActiveBg: "#111",

      buttonActiveText: "#FFC107",

      logoBg: "#FFC107",

      logoText: "#111"

    };

  }

  if (highContrast) {

    return {

      background: "#FFF",

      foreground: "#003366",

      border: "#003366",

      accent: "#FFD600",

      headerBg: "#003366",

      headerText: "#FFD600",

      sidebar: "#003366",

      sidebarText: "#FFD600",

      sidebarActive: "#FFD600",

      sidebarActiveText: "#003366",

      cardBg: "#FFF",

      buttonBg: "#FFD600",

      buttonText: "#003366",

      buttonActiveBg: "#003366",

      buttonActiveText: "#FFD600",

      logoBg: "#003366",

      logoText: "#FFD600"

    };

  }

  if (colorBlindSafe) {

    return {

      background: "#F3F7FA",

      foreground: "#1B4F72",

      border: "#009688",

      accent: "#FFC107",

      headerBg: "#1B4F72",

      headerText: "#FFF",

      sidebar: "#E1EAF3",

      sidebarText: "#1B4F72",

      sidebarActive: "#FFC107",

      sidebarActiveText: "#1B4F72",

      cardBg: "#FFF",

      buttonBg: "#009688",

      buttonText: "#FFF",

      buttonActiveBg: "#FFC107",

      buttonActiveText: "#1B4F72",

      logoBg: "#1B4F72",

      logoText: "#FFF"

    };

  }

  // Default palette

  return {

    background: "#F8F9FA",

    foreground: "#212121",

    border: "#E1EAF3",

    accent: "#FFC107",

    headerBg: "#F3F7FA",

    headerText: "#1B4F72",

    sidebar: "#F3F7FA",

    sidebarText: "#1B4F72",

    sidebarActive: "#FFC107",

    sidebarActiveText: "#1B4F72",

    cardBg: "#FFF",

    buttonBg: "#FFC107",

    buttonText: "#1B4F72",

    buttonActiveBg: "#1B4F72",

    buttonActiveText: "#FFC107",

    logoBg: "#FFC107",

    logoText: "#1B4F72"

  };

}

const AccessibleUserDashboard = () => {

  // useAccessibility context (keeps all settings in sync across tabs/components)

  const {

    largeText, setLargeText,

    highContrast, setHighContrast,

    dyslexiaFont, setDyslexiaFont,

    colorBlindSafe, setColorBlindSafe,

    sunlightMode, setSunlightMode,

  } = useAccessibility();

  // Local

  const [textSpacing, setTextSpacing] = useState(false);

  const [readAloudEnabled] = useState(true); // Always on!

  const [activeTab, setActiveTab] = useState("overview");

  const [isReading, setIsReading] = useState(false);

  const navigate = useNavigate();

  const mainContentRef = useRef<HTMLDivElement>(null);

  // --- Style settings

  const fontSize = largeText ? "22px" : "16px";

  const fontFamily = dyslexiaFont

    ? "'OpenDyslexic', 'Comic Sans MS', 'Segoe UI', Arial, sans-serif"

    : "'Segoe UI', 'Open Sans', Arial, sans-serif";

  const lineHeight = textSpacing ? 2.0 : 1.7;

  const palette = getPalette({ sunlightMode, highContrast, colorBlindSafe });

  // --- Web Speech API: Read aloud

  const handleReadAloud = () => {

    if (!window.speechSynthesis) return;

    setIsReading(true);

    let text = "";

    try {

      text = mainContentRef.current?.innerText ?? "";

    } catch {

      text = "Reading content.";

    }

    const utterance = new window.SpeechSynthesisUtterance(text);

    utterance.rate = 0.95;

    utterance.onend = () => setIsReading(false);

    utterance.onerror = () => setIsReading(false);

    window.speechSynthesis.speak(utterance);

  };

  const handleStopReading = () => {

    window.speechSynthesis?.cancel();

    setIsReading(false);

  };

  const handleLogout = () => navigate("/");

  const handleRegularDashboard = () => navigate("/user/dashboard");

  // Tabs (all the original dashboard features)

  const tabs = [

    { id: "overview", name: "Dashboard Overview", icon: BarChart3 },

    { id: "negotiation", name: "Interest Rate Optimization", icon: UploadCloud },

    { id: "smartinvestment", name: "Smart Investment", icon: TrendingUp },

    { id: "loansimulator", name: "Loan Simulator", icon: Calculator },

    { id: "loanprediction", name: "Loan Eligibility", icon: Brain },

    { id: "chatbot", name: "AI Chatbot", icon: MessageSquare },

    { id: "biometric", name: "Biometric Profile", icon: Smartphone },

    { id: "market", name: "Market Trends", icon: Building2 },

    { id: "gamification", name: "Rewards & Badges", icon: Trophy },

    { id: "notifications", name: "Notifications", icon: Bell },

    { id: "expense", name: "Expense Tracker", icon: Receipt },

    { id: "privacy", name: "Privacy Settings", icon: Shield },

    { id: "feedback", name: "Feedback", icon: MessageSquare },

    { id: "support", name: "Customer Support", icon: HelpCircle },

  ];

  // --- Accessibility Panel ---

  const AccessibilityPanel = (

    <section

      style={{

        background: palette.accent,

        borderRadius: 16,

        padding: "18px 22px",

        display: "flex",

        flexWrap: "nowrap",

        alignItems: "center",

        justifyContent: "flex-start",

        gap: "40px",

        marginBottom: 28,

        marginTop: 16,

        width: "100%",

        minWidth: 0,

        boxSizing: "border-box"

      }}

      aria-label="Accessibility Settings"

    >

      <label style={{ display: "flex", alignItems: "center", fontSize, fontFamily, fontWeight: 700, color: palette.foreground, flexShrink: 0 }}>

        <input type="checkbox" checked={largeText} onChange={() => setLargeText(v => !v)} style={{ marginRight: 8, width: 20, height: 20 }} />

        Large Text

      </label>

      <label style={{ display: "flex", alignItems: "center", fontSize, fontFamily, fontWeight: 700, color: palette.foreground, flexShrink: 0 }}>

        <input type="checkbox" checked={highContrast}

          onChange={() => {

            setHighContrast(v => {

              if (!v) { setColorBlindSafe(false); setSunlightMode(false); }

              return !v;

            });

          }}

          style={{ marginRight: 8, width: 20, height: 20 }} />

        High Contrast

      </label>

      <label style={{ display: "flex", alignItems: "center", fontSize, fontFamily, fontWeight: 700, color: palette.foreground, flexShrink: 0 }}>

        <input type="checkbox" checked={dyslexiaFont} onChange={() => setDyslexiaFont(v => !v)} style={{ marginRight: 8, width: 20, height: 20 }} />

        Lexie Readable Font

      </label>

      <label style={{ display: "flex", alignItems: "center", fontSize, fontFamily, fontWeight: 700, color: palette.foreground, flexShrink: 0 }}>

        <input type="checkbox" checked={colorBlindSafe}

          onChange={() => {

            setColorBlindSafe(v => {

              if (!v) { setHighContrast(false); setSunlightMode(false); }

              return !v;

            });

          }}

          style={{ marginRight: 8, width: 20, height: 20 }} />

        Color-blind Safe

      </label>

      <label style={{ display: "flex", alignItems: "center", fontSize, fontFamily, fontWeight: 700, color: palette.foreground, flexShrink: 0 }}>

        <input type="checkbox" checked={sunlightMode}

          onChange={() => {

            setSunlightMode(v => {

              if (!v) { setHighContrast(false); setColorBlindSafe(false); }

              return !v;

            });

          }}

          style={{ marginRight: 8, width: 20, height: 20 }} />

        Sunlight Mode

      </label>

      <label style={{ display: "flex", alignItems: "center", fontSize, fontFamily, fontWeight: 700, color: palette.foreground, flexShrink: 0 }}>

        <input type="checkbox" checked={textSpacing} onChange={() => setTextSpacing(v => !v)} style={{ marginRight: 8, width: 20, height: 20 }} />

        Text Spacing

      </label>

    </section>

  );

  // --- Render the main dashboard tab content ---

  const renderTabContent = () => {

    switch (activeTab) {

      case "overview":

        return (

          <div style={{ fontSize, fontFamily, lineHeight, color: palette.foreground }}>

            {/\* 6-Grid Summary Cards \*/}

            <section style={{

              display: "grid",

              gridTemplateColumns: "repeat(auto-fit, minmax(260px, 1fr))",

              gap: 32,

              marginBottom: 38

            }}>

              <div style={{

                borderRadius: 18, boxShadow: "0 3px 12px #0001",

                padding: 24, background: palette.cardBg,

                border: `3px solid ${palette.border}`,

                color: palette.foreground, fontSize, fontFamily

              }}>

                <div style={{ display: "flex", alignItems: "center", justifyContent: "space-between" }}>

                  <div>

                    <p style={{ fontWeight: 700, fontSize: largeText ? 24 : 18 }}>Current Interest Rate</p>

                    <p style={{ fontWeight: 900, fontSize: largeText ? 42 : 32 }}>8.5%</p>

                  </div>

                  <TrendingUp size={44} color={palette.accent} aria-label="Current Interest Rate" />

                </div>

              </div>

              <div style={{

                borderRadius: 18, boxShadow: "0 3px 12px #0001",

                padding: 24, background: palette.cardBg,

                border: `3px solid ${palette.border}`,

                color: palette.foreground, fontSize, fontFamily

              }}>

                <div style={{ display: "flex", alignItems: "center", justifyContent: "space-between" }}>

                  <div>

                    <p style={{ fontWeight: 700, fontSize: largeText ? 24 : 18 }}>Risk Score</p>

                    <p style={{ fontWeight: 900, fontSize: largeText ? 42 : 32 }}>Low</p>

                  </div>

                  <Shield size={44} color={palette.accent} aria-label="Risk Score" />

                </div>

              </div>

              <div style={{

                borderRadius: 18, boxShadow: "0 3px 12px #0001",

                padding: 24, background: palette.cardBg,

                border: `3px solid ${palette.border}`,

                color: palette.foreground, fontSize, fontFamily

              }}>

                <div style={{ display: "flex", alignItems: "center", justifyContent: "space-between" }}>

                  <div>

                    <p style={{ fontWeight: 700, fontSize: largeText ? 24 : 18 }}>Loan Tenure</p>

                    <p style={{ fontWeight: 900, fontSize: largeText ? 42 : 32 }}>15 Years</p>

                  </div>

                  <Calculator size={44} color={palette.accent} aria-label="Loan Tenure" />

                </div>

              </div>

              <div style={{

                borderRadius: 18, boxShadow: "0 3px 12px #0001",

                padding: 24, background: palette.cardBg,

                border: `3px solid ${palette.border}`,

                color: palette.foreground, fontSize, fontFamily

              }}>

                <div style={{ display: "flex", alignItems: "center", justifyContent: "space-between" }}>

                  <div>

                    <p style={{ fontWeight: 700, fontSize: largeText ? 24 : 18 }}>Savings Potential</p>

                    <p style={{ fontWeight: 900, fontSize: largeText ? 42 : 32 }}>₹2.5L</p>

                  </div>

                  <Trophy size={44} color={palette.accent} aria-label="Savings Potential" />

                </div>

              </div>

              <div style={{

                borderRadius: 18, boxShadow: "0 3px 12px #0001",

                padding: 24, background: palette.cardBg,

                border: `3px solid ${palette.border}`,

                color: palette.foreground, fontSize, fontFamily

              }}>

                <div style={{ display: "flex", alignItems: "center", justifyContent: "space-between" }}>

                  <div>

                    <p style={{ fontWeight: 700, fontSize: largeText ? 24 : 18 }}>Debt Balance</p>

                    <p style={{ fontWeight: 900, fontSize: largeText ? 42 : 32 }}>₹1.2L</p>

                  </div>

                  <BarChart3 size={44} color={palette.accent} aria-label="Debt Balance" />

                </div>

              </div>

              <div style={{

                borderRadius: 18, boxShadow: "0 3px 12px #0001",

                padding: 24, background: palette.cardBg,

                border: `3px solid ${palette.border}`,

                color: palette.foreground, fontSize, fontFamily

              }}>

                <div style={{ display: "flex", alignItems: "center", justifyContent: "space-between" }}>

                  <div>

                    <p style={{ fontWeight: 700, fontSize: largeText ? 24 : 18 }}>EMI Balance</p>

                    <p style={{ fontWeight: 900, fontSize: largeText ? 42 : 32 }}>₹32,000</p>

                  </div>

                  <Receipt size={44} color={palette.accent} aria-label="EMI Balance" />

                </div>

              </div>

            </section>

            {/\* Welcome Section \*/}

            <section style={{

              borderRadius: 18, boxShadow: "0 3px 12px #0001",

              padding: 32, background: palette.cardBg,

              border: `3px solid ${palette.border}`,

              color: palette.foreground, marginTop: 0,

              fontSize, fontFamily

            }}>

              <h2 style={{ fontSize: largeText ? 32 : 28, fontWeight: 700, marginBottom: 16 }}>

                Welcome to ApexWolf Accessible Dashboard

              </h2>

              <p style={{ fontSize: largeText ? 24 : 20 }}>

                This dashboard is made to be easy, clear, and usable for everyone.

                Explore all the same features, now in a more accessible way!

              </p>

            </section>

          </div>

        );

      case "negotiation":

        return <AccessibleNegotiation />;

      case "smartinvestment":

        return <AccessibleSmartInvestment />;

      case "loansimulator":

        return <AccessibleLoanSimulator />;

      case "loanprediction":

        return <AccessibleLoanPrediction />;

      case "market":

        return <AccessibleMarketTrends />;

      case "expense":

        return <AccessibleExpenseTracker />;

      case "notifications":

        return <AccessibleNotification />;

      case "gamification":

        return <AccessibleRewardsHub />;

      case "privacy":

        return <AccessibleSettings />;

      case "feedback":

        return <AccessibleFeedback />;

      case "chatbot":

        return <AccessibleChatbot onTabChange={setActiveTab} />;

      case "biometric":

        return <AccessibleBiometricForm />;

      case "profile":

      case "explanation":

      case "support":

  return <AccessibleCustomerSupport />;

        return (

          <div style={{

            borderRadius: 18, boxShadow: "0 3px 12px #0001",

            padding: 36, background: palette.cardBg,

            border: `3px solid ${palette.border}`,

            color: palette.foreground, textAlign: "center",

            fontSize, fontFamily

          }}>

            <div style={{

              width: 96, height: 96, background: "linear-gradient(135deg, #1a237e 60%, #4fc3f7 100%)",

              borderRadius: "100%", display: "flex", alignItems: "center",

              justifyContent: "center", margin: "0 auto 24px auto"

            }}>

              <Brain size={44} color="#fff" />

            </div>

            <h3 style={{ fontSize: largeText ? 32 : 28, fontWeight: 600, marginBottom: 14 }}>Feature Coming Soon</h3>

            <p style={{ fontSize: largeText ? 24 : 20 }}>This feature is under accessible design review and will be available soon.</p>

          </div>

        );

      default:

        return null;

    }

  };

  return (

    <div

      style={{

        background: palette.background,

        minHeight: "100vh",

        fontFamily,

        fontSize,

        lineHeight,

        color: palette.foreground

      }}

    >

      {/\* Header \*/}

      <header

        style={{

          background: palette.headerBg,

          color: palette.headerText,

          fontFamily,

          fontSize,

          boxShadow: "0 2px 10px #0001",

          borderBottom: `6px solid ${palette.accent}`,

          padding: "18px 0 12px 0"

        }}

      >

        <div style={{

          display: "flex", alignItems: "center", justifyContent: "space-between", flexWrap: "wrap", padding: "0 32px"

        }}>

          <div style={{ display: "flex", alignItems: "center", gap: 14 }}>

            <div style={{

              width: 56,

              height: 56,

              background: palette.logoBg,

              color: palette.logoText,

              borderRadius: 18,

              display: "flex",

              alignItems: "center",

              justifyContent: "center",

              fontSize: largeText ? 40 : 32,

              fontWeight: 900,

              fontFamily

            }} aria-label="ApexWolf Logo">

              AW

            </div>

            <h1 style={{

              fontSize: largeText ? 40 : 32,

              fontWeight: 700,

              letterSpacing: 1,

              color: palette.headerText,

              fontFamily

            }}>

              ApexWolf Accessible Dashboard

            </h1>

          </div>

          <div style={{ display: "flex", alignItems: "center", gap: 16 }}>

            <Button

              onClick={handleRegularDashboard}

              style={{

                background: palette.buttonBg,

                color: palette.buttonText,

                fontWeight: 700,

                borderRadius: 12,

                border: "none",

                fontFamily,

                minWidth: 44, minHeight: 44, fontSize, padding: "10px 20px"

              }}

              aria-label="Back to regular dashboard"

            >

              <Globe size={28} /> Regular View

            </Button>

            <Button

              onClick={handleLogout}

              style={{

                background: "#f44336",

                color: "#fff",

                fontWeight: 700,

                borderRadius: 12,

                border: "none",

                fontFamily,

                minWidth: 44, minHeight: 44, fontSize, padding: "10px 20px"

              }}

              aria-label="Logout"

            >

              <LogOut size={28} /> Logout

            </Button>

          </div>

        </div>

      </header>

      {/\* Accessibility Settings \*/}

      <div style={{ padding: "0 36px", marginTop: 0, marginBottom: 0, fontFamily, fontSize }} aria-label="Accessibility controls">

        {AccessibilityPanel}

      </div>

      {/\* Layout \*/}

      <div style={{ display: "flex" }}>

        {/\* Sidebar \*/}

        <nav

          style={{

            width: 320,

            minHeight: "100vh",

            borderRight: `4px solid ${palette.border}`,

            padding: "38px 18px 24px 18px",

            background: palette.sidebar,

            color: palette.sidebarText,

            fontFamily,

            fontSize

          }}

          aria-label="Sidebar Navigation"

        >

          <div style={{

            fontWeight: 700,

            fontSize: largeText ? 26 : 22,

            marginBottom: 28,

            color: palette.sidebarText,

            fontFamily

          }}>

            All Features

          </div>

          <ul style={{ display: "flex", flexDirection: "column", gap: 14 }}>

            {tabs.map((tab) => {

              const Icon = tab.icon;

              const active = activeTab === tab.id;

              return (

                <li key={tab.id}>

                  <button

                    onClick={() => setActiveTab(tab.id)}

                    style={{

                      width: "100%",

                      display: "flex",

                      alignItems: "center",

                      gap: 16,

                      padding: "18px 22px",

                      fontSize: largeText ? 22 : 18,

                      borderRadius: 14,

                      background: active ? palette.sidebarActive : "#FFF",

                      color: active ? palette.sidebarActiveText : palette.sidebarText,

                      fontWeight: 600,

                      boxShadow: active ? "0 2px 8px #FFC10733" : "none",

                      border: "none",

                      outline: "none",

                      cursor: "pointer",

                      fontFamily,

                      minWidth: 44, minHeight: 44

                    }}

                    tabIndex={0}

                    aria-current={active ? "page" : undefined}

                  >

                    <Icon size={28} aria-hidden="true" />

                    <span>{tab.name}</span>

                  </button>

                </li>

              );

            })}

          </ul>

        </nav>

        {/\* Main content \*/}

        <main

          style={{ flex: 1, padding: "40px 48px", fontFamily, fontSize, color: palette.foreground, background: palette.background }}

          id="main-content"

          tabIndex={-1}

          ref={mainContentRef}

        >

          <div style={{ display: "flex", alignItems: "center", gap: 22, marginBottom: 24 }}>

            <h2 style={{ fontSize: largeText ? 36 : 32, fontWeight: 700, color: palette.foreground, fontFamily }}>

              {tabs.find(t => t.id === activeTab)?.name}

            </h2>

            {/\* Read aloud always visible \*/}

            {!isReading ? (

              <Button

                onClick={handleReadAloud}

                style={{

                  background: palette.buttonBg,

                  color: palette.buttonText,

                  borderRadius: 12,

                  border: "none",

                  fontFamily,

                  minWidth: 44, minHeight: 44, fontSize, padding: "10px 20px"

                }}

                aria-label="Read this section aloud"

              >

                <Volume2 size={24} /> Read Aloud

              </Button>

            ) : (

              <Button

                onClick={handleStopReading}

                style={{

                  background: palette.buttonActiveBg,

                  color: palette.buttonActiveText,

                  borderRadius: 12,

                  border: "none",

                  fontFamily,

                  minWidth: 44, minHeight: 44, fontSize, padding: "10px 20px"

                }}

                aria-label="Stop reading aloud"

              >

                <Volume2 size={24} /> Stop

              </Button>

            )}

          </div>

          {renderTabContent()}

        </main>

      </div>

      {/\* Styles for accessibility features \*/}

      <style>

        {dyslexiaFont

          ? `

          @import url('https://cdn.jsdelivr.net/npm/open-dyslexic-font@0.0.3/open-dyslexic.min.css');

          body, .dyslexia { font-family: 'OpenDyslexic', 'Comic Sans MS', 'Segoe UI', Arial, sans-serif !important; }

        `

          : ''}

        {textSpacing ? `

          body, #root, .dyslexia, .accessible-dashboard {

            letter-spacing: 0.07em !important;

            word-spacing: 0.16em !important;

            line-height: 2.0 !important;

          }

        ` : ""}

      </style>

    </div>

  );

};

export default AccessibleUserDashboard;

**USER DASHBOARD**

import React, { useState } from 'react';

import { useNavigate } from 'react-router-dom';

import { Banknote, AlarmClock } from 'lucide-react';

import {

  Brain, BookOpen, User, FileText, BarChart3, MessageSquare,

  TrendingUp, Calculator, Trophy, Building2, Bell,

  Receipt, Shield, HelpCircle, Globe, Smartphone,

  LogOut, Menu, X, UploadCloud, Megaphone, Wallet2, Mic

} from 'lucide-react';

import { Button } from '@/components/ui/Button';

import { Card, CardContent, CardHeader, CardTitle } from '@/components/ui/Card';

import SmartInvestment from '@/components/SmartInvestment';

import Negotiation from '@/components/Negotiation';

import LoanSimulator from '@/components/LoanSimulator';

import LoanPrediction from '@/components/LoanPrediction';

import MarketTrends from '@/components/MarketTrends';

import ExpenseTracker from '@/components/ExpenseTracker';

import Notification from '@/components/Notification';

import RewardsHub from '@/components/RewardsHub';

import SettingsComponent from '@/components/Settings';

import Feedback from '@/components/Feedback';

import Chatbot from '@/components/Chatbot';

import BiometricForm from '@/components/Biometric';

import LoanProcessing from '@/components/LoanProcessing';

import EmergencyLoan from '@/components/EmergencyLoan';

import CustomerSupport from '@/components/CustomerSupport';

import RoleBased from '@/components/RoleBased';

import RepoRate from '@/components/RepoRate';

import UserManual from "@/components/UserManual";

import { startVoiceRecognition } from '@/Voicecontrol';

import Announcement from '@/components/Announcement';

import RepaymentManagement from '@/components/Repayment';

const UserDashboard = () => {

  const [activeTab, setActiveTab] = useState('overview');

  const [sidebarOpen, setSidebarOpen] = useState(true);

  const navigate = useNavigate();

  const [showManual, setShowManual] = useState(false);

  // Handles logout

  const handleLogout = () => {

    navigate('/');

  };

  // Handles navigation to accessible dashboard

  const handleAccessibleDashboard = () => {

    navigate('/user/accessible');

  };

  // Handles voice commands

  const handleVoiceCommand = (command: string) => {

    const c = command.toLowerCase();

    if (c.includes('logout')) {

      handleLogout();

    } else if (c.includes('accessible')) {

      handleAccessibleDashboard();

    } else if (c.includes('dashboard') || c.includes('overview')) {

      setActiveTab('overview');

    } else if (c.includes('prediction') || c.includes('role based')) {

      setActiveTab('profile');

    } else if (c.includes('loan eligibility')) {

      setActiveTab('loanprediction');

    } else if (c.includes('interest') && c.includes('optimization')) {

      setActiveTab('negotiation');

    } else if (c.includes('loan simulator') || c.includes('simulate')) {

      setActiveTab('loansimulator');

    } else if (c.includes('loan processing') || c.includes('process loan')) {

      setActiveTab('processing');

    } else if (c.includes('emergency loan')) {

      setActiveTab('emergency');

    } else if (c.includes('smart investment')) {

      setActiveTab('smartinvestment');

    } else if (c.includes('repo rate') || c.includes('interest rate policy')) {

      setActiveTab('repo');

    } else if (c.includes('market trends') || c.includes('market data')) {

      setActiveTab('market');

    } else if (c.includes('expense tracker') || c.includes('expenses')) {

      setActiveTab('expense');

    } else if (c.includes('rewards') || c.includes('badges') || c.includes('gamification')) {

      setActiveTab('gamification');

    } else if (c.includes('notifications') || c.includes('alerts')) {

      setActiveTab('notifications');

    } else if (c.includes('chatbot') || c.includes('help bot')) {

      setActiveTab('chatbot');

    } else if (c.includes('privacy') || c.includes('data settings')) {

      setActiveTab('privacy');

    } else if (c.includes('biometric') || c.includes('kyc')) {

      setActiveTab('biometric');

    } else if (c.includes('feedback')) {

      setActiveTab('feedback');

    } else if (c.includes('support') || c.includes('customer support')) {

      setActiveTab('support');

    } else {

      alert(`Command not recognized: "${command}"`);

    }

  };

  const handleStartVoice = () => {

    startVoiceRecognition(handleVoiceCommand);

  };

  const dashboardTabs = [

  { id: 'overview', name: 'Dashboard Overview', icon: BarChart3 },

  { id: 'profile', name: 'Role-Based Prediction', icon: User },

  { id: 'loanprediction', name: 'Loan Eligibility', icon: Brain },

  { id: 'negotiation', name: 'Interest Rate Optimization', icon: UploadCloud },

        { id: 'smartinvestment', name: 'Smart Investment', icon: TrendingUp },

  { id: 'loansimulator', name: 'Loan Simulator', icon: Calculator },

        { id: 'repayment', name: 'Repayment Management', icon: Wallet2 },

  { id: 'processing', name: 'Loan Processing', icon: Banknote },

  { id: 'emergency', name: 'Emergency Loan', icon: AlarmClock },

  { id: 'repo', name: 'Repo Rate', icon: FileText },

  { id: 'market', name: 'Market Trends', icon: Building2 },

  { id: 'expense', name: 'Expense Tracker', icon: Receipt },

  { id: 'gamification', name: 'Rewards & Badges', icon: Trophy },

        { id: 'announcements', name: 'Announcements Banner', icon: Megaphone },

  { id: 'notifications', name: 'Notifications', icon: Bell },

  { id: 'chatbot', name: 'AI Chatbot', icon: MessageSquare },

  { id: 'privacy', name: 'Privacy Settings', icon: Shield },

  { id: 'biometric', name: 'Biometric Profile', icon: Smartphone },

  { id: 'feedback', name: 'Feedback', icon: MessageSquare },

  { id: 'support', name: 'Customer Support', icon: HelpCircle },

  ];

  const renderTabContent = () => {

    switch (activeTab) {

      case 'overview':

        return (

          <div className="space-y-6">

            <div className="grid grid-cols-1 md:grid-cols-2 lg:grid-cols-4 gap-6">

              <Card className="bg-gradient-to-br from-emerald-50 to-emerald-100 border-emerald-200">

                <CardContent className="p-6">

                  <div className="flex items-center justify-between">

                    <div>

                      <p className="text-emerald-600 text-sm font-medium">Current Interest Rate</p>

                      <p className="text-2xl font-bold text-emerald-800">8.5%</p>

                    </div>

                    <TrendingUp className="w-8 h-8 text-emerald-600" />

                  </div>

                </CardContent>

              </Card>

              <Card className="bg-gradient-to-br from-blue-50 to-blue-100 border-blue-200">

                <CardContent className="p-6">

                  <div className="flex items-center justify-between">

                    <div>

                      <p className="text-blue-600 text-sm font-medium">Risk Score</p>

                      <p className="text-2xl font-bold text-blue-800">Low</p>

                    </div>

                    <Shield className="w-8 h-8 text-blue-600" />

                  </div>

                </CardContent>

              </Card>

              <Card className="bg-gradient-to-br from-amber-50 to-amber-100 border-amber-200">

                <CardContent className="p-6">

                  <div className="flex items-center justify-between">

                    <div>

                      <p className="text-amber-600 text-sm font-medium">Loan Tenure</p>

                      <p className="text-2xl font-bold text-amber-800">15 Years</p>

                    </div>

                    <Calculator className="w-8 h-8 text-amber-600" />

                  </div>

                </CardContent>

              </Card>

              <Card className="bg-gradient-to-br from-rose-50 to-rose-100 border-rose-200">

                <CardContent className="p-6">

                  <div className="flex items-center justify-between">

                    <div>

                      <p className="text-rose-600 text-sm font-medium">Savings Potential</p>

                      <p className="text-2xl font-bold text-rose-800">₹2.5L</p>

                    </div>

                    <Trophy className="w-8 h-8 text-rose-600" />

                  </div>

                </CardContent>

              </Card>

            </div>

            <Card className="bg-gradient-to-br from-violet-50 to-violet-100 border-violet-200">

              <CardHeader>

                <CardTitle className="text-violet-800">Welcome to ApexWolf Dashboard</CardTitle>

              </CardHeader>

              <CardContent>

                <p className="text-violet-700">

                  Your personalized AI-powered financial dashboard is ready. Explore the various tabs to get predictions,

                  analyze your financial profile, and make informed decisions.

                </p>

              </CardContent>

            </Card>

          </div>

        );

      case 'negotiation':

        return <Negotiation />;

      case 'smartinvestment':

        return <SmartInvestment />;

      case "loansimulator":

        return <LoanSimulator />;

      case 'loanprediction':

        return <LoanPrediction />;

      case 'market':

        return <MarketTrends />;

      case 'expense':

        return <ExpenseTracker />;

      case 'notifications':

        return <Notification />;

      case 'gamification':

        return <RewardsHub />;

      case 'privacy':

        return <SettingsComponent />;

      case 'feedback':

        return <Feedback />;

      case 'chatbot':

        return <Chatbot onTabChange={setActiveTab} />;

      case 'biometric':

        return <BiometricForm />;

      case 'processing':

  return <LoanProcessing />;

      case 'emergency':

  return <EmergencyLoan />;

      case 'support':

  return <CustomerSupport />;

      case 'profile':

  return <RoleBased />;

      case 'repo':

  return <RepoRate />;

      case 'announcements':

  return <Announcement />;

      case 'repayment':

  return <RepaymentManagement />;

      default:

        return (

          <Card className="bg-gradient-to-br from-slate-50 to-slate-100 border-slate-200">

            <CardContent className="p-8 text-center">

              <div className="w-16 h-16 bg-gradient-to-r from-indigo-400 to-purple-400 rounded-full flex items-center justify-center mx-auto mb-4">

                <Brain className="w-8 h-8 text-white" />

              </div>

              <h3 className="text-xl font-semibold text-slate-800 mb-2">Feature Under Development</h3>

              <p className="text-slate-600">This advanced feature is being developed with cutting-edge AI technology.</p>

            </CardContent>

          </Card>

        );

    }

  };

  return (

    <div className="min-h-screen bg-gradient-to-br from-slate-50 via-blue-50 to-indigo-50">

      {/\* Header \*/}

      <header className="bg-white shadow-lg border-b border-indigo-100">

            <div className="flex items-center justify-between px-6 py-4">

                  {/\* Left section: toggle + logo \*/}

                  <div className="flex items-center gap-4">

                        <Button

                              variant="ghost"

                              size="sm"

                              onClick={() => setSidebarOpen(!sidebarOpen)}

                              className="lg:hidden text-indigo-600 hover:bg-indigo-50"

                        >

                              {sidebarOpen ? <X className="w-5 h-5" /> : <Menu className="w-5 h-5" />}

                        </Button>

                        <div className="flex items-center gap-3">

                              <div className="w-10 h-10 bg-gradient-to-r from-indigo-500 to-purple-500 rounded-lg flex items-center justify-center">

                                    <Brain className="w-6 h-6 text-white" />

                              </div>

                              <div>

                                    <h1 className="text-xl font-bold bg-gradient-to-r from-indigo-600 to-purple-600 bg-clip-text text-transparent">

                                          ApexWolf

                                    </h1>

                                    <p className="text-sm text-slate-600">AI Powered Pricing</p>

                              </div>

                        </div>

                  </div>

                  {/\* Right section: action buttons \*/}

                  <div className="flex flex-wrap justify-end items-center gap-2 sm:gap-3 md:gap-4">

                        {/\* User Manual \*/}

                        <Button

                              variant="ghost"

                              size="sm"

                              onClick={() => navigate("/user/user-manual", { state: { from: "/user/dashboard" } })}

                              className="bg-gradient-to-r from-indigo-500 to-purple-500 text-white hover:opacity-90 px-3 py-2 rounded-md flex items-center gap-2"

                              aria-label="Open User Manual"

                              title="User Manual"

                        >

                              <BookOpen className="w-5 h-5" />

                              <span className="hidden sm:inline">User Manual</span>

                        </Button>

                        {/\* Accessibility \*/}

                        <Button

                              variant="ghost"

                              size="sm"

                              onClick={handleAccessibleDashboard}

                              className="bg-gradient-to-r from-indigo-500 to-purple-500 text-white hover:opacity-90 px-3 py-2 rounded-md flex items-center gap-2"

                              aria-label="Switch to accessible dashboard"

                              title="Accessible Mode"

                        >

                              <Globe className="w-5 h-5" />

                              <span className="hidden sm:inline">Accessible Mode</span>

                        </Button>

                        {/\* Voice Control \*/}

                        <Button

                              variant="ghost"

                              size="sm"

                              onClick={handleStartVoice}

                              className="bg-gradient-to-r from-indigo-500 to-purple-500 text-white hover:opacity-90 px-3 py-2 rounded-md flex items-center gap-2"

                              title="Activate Voice Control"

                        >

                              <Mic classname="w-5 h-5" />

                              <span className="hidden sm:inline">Voice Access</span>

                        </Button>

                        {/\* Logout \*/}

                        <Button

                              variant="ghost"

                              size="sm"

                              onClick={handleLogout}

                              className="bg-gradient-to-r from-indigo-500 to-purple-500 text-white hover:opacity-90 px-3 py-2 rounded-md flex items-center gap-2"

                        >

                              <LogOut className="w-5 h-5" />

                              <span className="hidden sm:inline">Logout</span>

                        </Button>

                  </div>

            </div>

      </header>

      <div className="flex">

        {/\* Sidebar \*/}

        <aside className={`${sidebarOpen ? 'translate-x-0' : '-translate-x-full'} fixed lg:relative lg:translate-x-0 z-30 w-72 h-screen bg-white shadow-xl border-r border-indigo-100 transition-transform duration-300 ease-in-out`}>

          <div className="p-6 border-b border-indigo-100">

            <div className="flex items-center space-x-3">

              <div className="w-12 h-12 bg-gradient-to-r from-emerald-400 to-cyan-400 rounded-full flex items-center justify-center">

                <User className="w-6 h-6 text-white" />

              </div>

              <div>

                <h3 className="font-semibold text-slate-800">Welcome User!</h3>

              </div>

            </div>

          </div>

          <nav className="p-4 space-y-2 max-h-[calc(100vh-140px)] overflow-y-auto">

            {dashboardTabs.map((tab) => {

              const Icon = tab.icon;

              return (

                <button

                  key={tab.id}

                  onClick={() => setActiveTab(tab.id)}

                  className={`w-full flex items-center space-x-3 px-4 py-3 rounded-lg text-left transition-all duration-200 ${

                    activeTab === tab.id

                      ? 'bg-gradient-to-r from-indigo-500 to-purple-500 text-white shadow-md'

                      : 'text-slate-600 hover:bg-gradient-to-r hover:from-indigo-50 hover:to-purple-50 hover:text-indigo-700'

                  }`}

                >

                  <Icon className="w-5 h-5" />

                  <span className="text-sm font-medium">{tab.name}</span>

                </button>

              );

            })}

          </nav>

        </aside>

        {/\* Main Content \*/}

        <main className="flex-1 p-6 lg:ml-0">

          <div className="mb-6">

            <h2 className="text-2xl font-bold text-slate-800 mb-2">

              {dashboardTabs.find(tab => tab.id === activeTab)?.name}

            </h2>

            <p className="text-slate-600">Manage your financial profile and get AI-powered insights</p>

          </div>

          {renderTabContent()}

        </main>

      </div>

      {/\* Mobile Overlay \*/}

      {sidebarOpen && (

        <div className="fixed inset-0 bg-black bg-opacity-50 z-20 lg:hidden" onClick={() => setSidebarOpen(false)} />

      )}

    </div>

  );

};

export default UserDashboard;

**ADMIN DASHBOARD**

import React, { useState } from 'react';

import { useNavigate } from 'react-router-dom';

import { Banknote, AlarmClock } from 'lucide-react';

import {

  Shield,

  Users,

  FileText,

  BarChart3,

  BarChart,

  Settings,

  Database,

  TrendingUp,

  Brain,

  AlertTriangle,

  Download,

  Upload,

  Monitor,

  MessageSquare,

  Globe,

  Key,

  Activity,

  Layers,

  Zap,

  Archive,

  LogOut,

  Menu,

  X,

  UploadCloud

} from 'lucide-react';

import { Button } from '@/components/ui/button';

import { Card, CardContent, CardHeader, CardTitle } from '@/components/ui/card';

import SettingsAdmin from '@/components/SettingsAdmin';

import InvestmentAdmin from '@/components/InvestmentAdmin';

import NegotiationAdmin from '@/components/NegotiationAdmin';

import LoanSimulatorAdmin from '@/components/LoanSimulatorAdmin';

import PredictionAdmin from '@/components/PredictionAdmin';

import MarketTrendsAdmin from '@/components/MarketTrendsAdmin';

import ExpenseTrackerAdmin from '@/components/ExpenseTrackerAdmin';

import NotificationAdmin from '@/components/NotificationAdmin';

import RewardsHubAdmin from '@/components/RewardsHubAdmin';

import FeedbackAdmin from '@/components/FeedbackAdmin';

import LoanEligibilityAdmin from '@/components/LoanEligibilityAdmin';

import SystemLogs from '@/components/SystemLogs';

import LoanProcessingAdmin from '@/components/LoanProcessingAdmin';

import EmergencyLoanAdmin from '@/components/EmergencyLoanAdmin';

import RoleBasedAdmin from '@/components/RoleBasedAdmin';

const AdminDashboard: React.FC = () => {

  const [activeTab, setActiveTab] = useState<string>('overview');

  const [sidebarOpen, setSidebarOpen] = useState<boolean>(true);

  const navigate = useNavigate();

  const handleLogout = () => {

    navigate('/');

  };

  const adminTabs = [

  { id: 'overview',            name: 'Admin Overview',         icon: BarChart3 },              // Dashboard Summary

  { id: 'users',               name: 'Role Based Analytics',   icon: Users },                  // Role Segmentation & Insights

  { id: 'loanpredictionadmin', name: 'Eligibility Weights',    icon: Brain },                  // Model Configs

  { id: 'loaneligibility',     name: 'Loan Eligibility Logs',  icon: FileText },               // Prediction Logs

  { id: 'loanadmin',           name: 'Loan Simulator',         icon: Settings },               // EMI Configs

  { id: 'processing',          name: 'Loan Processing',        icon: Banknote },

  { id: 'negotiation',         name: 'Rate Optimization',      icon: UploadCloud },

  { id: 'emergency',           name: 'Emergency Loan',         icon: AlarmClock },

  { id: 'navadmin',            name: 'Investment Update',      icon: Database },

  { id: 'markettrends',        name: 'Market Trends Monitor',  icon: BarChart },

  { id: 'economic',            name: 'Economic Tracker',       icon: TrendingUp },

  { id: 'gamification',        name: 'Gamification Config',    icon: Layers },

  { id: 'broadcast',           name: 'Notification Broadcast', icon: Zap },

  { id: 'feedback',            name: 'Feedback Review',        icon: MessageSquare },

  { id: 'logs',                name: 'System Logs',            icon: Monitor },

  { id: 'settings',            name: 'Settings',               icon: Key }

  ];

const renderTabContent = () => {

    switch (activeTab) {

      case 'overview':

        return (

          <div className="space-y-6">

            <div className="grid grid-cols-1 md:grid-cols-2 lg:grid-cols-4 gap-6">

              <Card className="bg-gradient-to-br from-emerald-50 to-emerald-100 border-emerald-200">

                <CardContent className="p-6">

                  <div className="flex items-center justify-between">

                    <div>

                      <p className="text-emerald-600 text-sm font-medium">Total Users</p>

                      <p className="text-2xl font-bold text-emerald-800">12,847</p>

                    </div>

                    <Users className="w-8 h-8 text-emerald-600" />

                  </div>

                </CardContent>

              </Card>

              <Card className="bg-gradient-to-br from-blue-50 to-blue-100 border-blue-200">

                <CardContent className="p-6">

                  <div className="flex items-center justify-between">

                    <div>

                      <p className="text-blue-600 text-sm font-medium">Model Accuracy</p>

                      <p className="text-2xl font-bold text-blue-800">99.98%</p>

                    </div>

                    <Brain className="w-8 h-8 text-blue-600" />

                  </div>

                </CardContent>

              </Card>

              <Card className="bg-gradient-to-br from-amber-50 to-amber-100 border-amber-200">

                <CardContent className="p-6">

                  <div className="flex items-center justify-between">

                    <div>

                      <p className="text-amber-600 text-sm font-medium">Active Models</p>

                      <p className="text-2xl font-bold text-amber-800">32/32</p>

                    </div>

                    <Database className="w-8 h-8 text-amber-600" />

                  </div>

                </CardContent>

              </Card>

              <Card className="bg-gradient-to-br from-rose-50 to-rose-100 border-rose-200">

                <CardContent className="p-6">

                  <div className="flex items-center justify-between">

                    <div>

                      <p className="text-rose-600 text-sm font-medium">Daily Predictions</p>

                      <p className="text-2xl font-bold text-rose-800">1,234</p>

                    </div>

                    <TrendingUp className="w-8 h-8 text-rose-600" />

                  </div>

                </CardContent>

              </Card>

            </div>

            <Card className="bg-gradient-to-br from-violet-50 to-violet-100 border-violet-200">

              <CardHeader>

                <CardTitle className="text-violet-800">Administrative Control Center</CardTitle>

              </CardHeader>

              <CardContent>

                <p className="text-violet-700">

                  Welcome to the ApexWolf Admin Dashboard. Monitor system performance, manage users,

                  and configure AI models from this comprehensive control panel.

                </p>

              </CardContent>

            </Card>

          </div>

        );

      case 'negotiation':

        return <NegotiationAdmin />;

      case 'navadmin':

        return <InvestmentAdmin />;

      case 'loanadmin':

        return <LoanSimulatorAdmin />;

      case 'loanpredictionadmin':

        return <PredictionAdmin />;

      case 'economic':

        return <ExpenseTrackerAdmin />;

      case 'markettrends':

        return <MarketTrendsAdmin />;

      case 'broadcast':

  return <NotificationAdmin />;

      case 'gamification':

  return <RewardsHubAdmin />;

      case 'settings':

  return <SettingsAdmin />;

      case 'feedback':

  return <FeedbackAdmin />;

      case 'loaneligibility':

  return <LoanEligibilityAdmin />;

      case 'logs':

  return <SystemLogs />;

      case 'emergency':

  return <EmergencyLoanAdmin />;

      case 'processing':

  return <LoanProcessingAdmin />;

      case 'users':

  return <RoleBasedAdmin />;

      default:

        return (

          <Card className="bg-gradient-to-br from-slate-50 to-slate-100 border-slate-200">

            <CardContent className="p-8 text-center">

              <div className="w-16 h-16 bg-gradient-to-r from-red-400 to-orange-400 rounded-full flex items-center justify-center mx-auto mb-4">

                <Shield className="w-8 h-8 text-white" />

              </div>

              <h3 className="text-xl font-semibold text-slate-800 mb-2">Admin Feature</h3>

              <p className="text-slate-600">

                This administrative feature provides advanced system control and monitoring capabilities.

              </p>

            </CardContent>

          </Card>

        );

    }

  };

  return (

    <div className="min-h-screen bg-gradient-to-br from-slate-50 via-red-50 to-orange-50">

      {/\* Header \*/}

      <header className="bg-white shadow-lg border-b border-red-100">

        <div className="flex items-center justify-between px-6 py-4">

          <div className="flex items-center space-x-4">

            <Button

              variant="ghost"

              size="sm"

              onClick={() => setSidebarOpen(!sidebarOpen)}

              className="lg:hidden text-red-600 hover:bg-red-50"

            >

              {sidebarOpen ? <X className="w-5 h-5" /> : <Menu className="w-5 h-5" />}

            </Button>

            <div className="flex items-center space-x-3">

              <div className="w-10 h-10 bg-gradient-to-r from-red-500 to-orange-500 rounded-lg flex items-center justify-center">

                <Shield className="w-6 h-6 text-white" />

              </div>

              <div>

                <h1 className="text-xl font-bold bg-gradient-to-r from-red-600 to-orange-600 bg-clip-text text-transparent">

                  ApexWolf Admin

                </h1>

                <p className="text-sm text-slate-600">Administrative Portal</p>

              </div>

            </div>

          </div>

          <div className="flex items-center space-x-4">

            <Button variant="ghost" size="sm" className="text-slate-600 hover:bg-slate-100">

              <AlertTriangle className="w-5 h-5" />

            </Button>

            <Button

              variant="ghost"

              size="sm"

              onClick={handleLogout}

              className="text-red-500 hover:bg-red-50"

            >

              <LogOut className="w-5 h-5" />

            </Button>

          </div>

        </div>

      </header>

      <div className="flex">

        {/\* Sidebar \*/}

        <aside className={`${sidebarOpen ? 'translate-x-0' : '-translate-x-full'} fixed lg:relative lg:translate-x-0 z-30 w-72 h-screen bg-white shadow-xl border-r border-red-100 transition-transform duration-300 ease-in-out`}>

          <div className="p-6 border-b border-red-100">

            <div className="flex items-center space-x-3">

              <div className="w-12 h-12 bg-gradient-to-r from-red-400 to-orange-400 rounded-full flex items-center justify-center">

                <Shield className="w-6 h-6 text-white" />

              </div>

              <div>

                <h3 className="font-semibold text-slate-800">System Admin</h3>

                <p className="text-sm text-slate-500">Full Access Control</p>

              </div>

            </div>

          </div>

          <nav className="p-4 space-y-2 max-h-[calc(100vh-140px)] overflow-y-auto">

            {adminTabs.map((tab) => {

              const Icon = tab.icon;

              return (

                <button

                  key={tab.id}

                  onClick={() => setActiveTab(tab.id)}

                  className={`w-full flex items-center space-x-3 px-4 py-3 rounded-lg text-left transition-all duration-200 ${

                    activeTab === tab.id

                      ? 'bg-gradient-to-r from-red-500 to-orange-500 text-white shadow-md'

                      : 'text-slate-600 hover:bg-gradient-to-r hover:from-red-50 hover:to-orange-50 hover:text-red-700'

                  }`}

                >

                  <Icon className="w-5 h-5" />

                  <span className="text-sm font-medium">{tab.name}</span>

                </button>

              );

            })}

          </nav>

        </aside>

        {/\* Main Content \*/}

        <main className="flex-1 p-6 lg:ml-0">

          <div className="mb-6">

            <h2 className="text-2xl font-bold text-slate-800 mb-2">

              {adminTabs.find(tab => tab.id === activeTab)?.name}

            </h2>

            <p className="text-slate-600">Advanced administrative controls and system monitoring</p>

          </div>

          {renderTabContent()}

        </main>

      </div>

      {/\* Mobile Overlay \*/}

      {sidebarOpen && (

        <div className="fixed inset-0 bg-black bg-opacity-50 z-20 lg:hidden" onClick={() => setSidebarOpen(false)} />

      )}

    </div>

  );

};

export default AdminDashboard;