# app/win\_home.py

import tkinter as tk

from tkinter import ttk, messagebox

import datetime

import requests

# Ventanas de tu app

from app.win\_canvas import open\_win\_canvas

from app.win\_list import CoachApp

from app.win\_consejos\_ai import open\_win\_consejos # Consejo inteligente (hora+clima)

from app.win\_frase\_motivacional import open\_win\_frase # Frase motivacional (IA)

# Si tu panel de pasos está en app/win\_steps.py con un main()

try:

from app.win\_steps import main as open\_win\_steps

except Exception:

open\_win\_steps = None

# Música (búsqueda y reproducción)

try:

from app.win\_musica import open\_win\_musica

except Exception:

open\_win\_musica = None

# Mini-player (play/pausa/siguiente/anterior/estado)

try:

from app.win\_player import open\_win\_player

except Exception:

open\_win\_player = None

# ====== Paleta AbueFit ======

COL\_BG\_APP = "#F9E8C6" # fondo general crema

COL\_SIDEBAR = "#F07C00" # naranja principal

COL\_SIDELITE = "#FFAA40" # naranja claro para nav

COL\_BTN\_BG = "#FFE7C2" # botones crema

COL\_BTN\_FG = "#2B2B2B" # texto botón

COL\_PANEL\_BG = "#FFF3DD" # panel derecho

COL\_TEXT\_ORG = "#F07C00" # texto destacado

NOMBRE\_USUARIO = "Mauricio" # <- personaliza

# ====== Utilidades fecha/clima ======

\_DIAS = ["Lun", "Mar", "Mié", "Jue", "Vie", "Sáb", "Dom"]

\_MESES = ["ene", "feb", "mar", "abr", "may", "jun", "jul", "ago", "sep", "oct", "nov", "dic"]

def fecha\_compacta(dt: datetime.datetime) -> str:

return f"{\_DIAS[dt.weekday()]} {dt.day} {\_MESES[dt.month-1]}"

def saludo\_por\_hora(nombre: str) -> str:

h = datetime.datetime.now().hour

if h < 12: return f"Buenos días, {nombre}"

if h < 19: return f"Buenas tardes, {nombre}"

return f"Buenas noches, {nombre}"

def clima\_leon\_texto() -> str:

"""Ej. '☀️ Despejado · 24°C' (fallback si falla)."""

try:

url = "https://api.open-meteo.com/v1/forecast"

params = {"latitude": 21.12, "longitude": -101.68, "current\_weather": True}

r = requests.get(url, params=params, timeout=4)

cw = r.json()["current\_weather"]

temp = int(round(cw["temperature"]))

code = int(cw["weathercode"])

if code in (61,63,65,80,81,82):

icono, desc = "🌧️", "Lluvia"

elif code in (71,73,75):

icono, desc = "❄️", "Nieve"

elif code in (0,1):

icono, desc = "☀️", "Despejado"

elif code in (2,3):

icono, desc = "⛅", "Parcial nublado"

elif code in (45,48):

icono, desc = "🌫️", "Niebla"

else:

icono, desc = "🌥️", "Variable"

return f"{icono} {desc} · {temp}°C"

except Exception:

return "ℹ️ Clima no disponible"

# ====== Ventana del Coach (chat) ======

def open\_win\_coach(parent: tk.Tk):

win = tk.Toplevel(parent)

win.title("Asistente de Bienestar — AbueFit")

win.configure(bg="#0A0F1E")

app = CoachApp(win)

app.pack(fill="both", expand=True)

win.bind("<Escape>", lambda e: win.destroy())

# ====== Home ======

def open\_win\_home(parent: tk.Tk):

win = tk.Toplevel(parent)

win.title("AbueFit — Inicio")

win.geometry("360x640+60+60")

win.resizable(False, False)

win.configure(bg=COL\_BG\_APP)

# Base

base = tk.Frame(win, bg=COL\_BG\_APP)

base.pack(fill="both", expand=True)

# Sidebar

sidebar = tk.Frame(base, bg=COL\_SIDEBAR, width=200)

sidebar.pack(side="left", fill="y")

sidebar.pack\_propagate(False)

# Content

content = tk.Frame(base, bg=COL\_PANEL\_BG)

content.pack(side="left", fill="both", expand=True)

# Header sidebar

header = tk.Frame(sidebar, bg=COL\_SIDEBAR, height=150)

header.pack(fill="x")

header.pack\_propagate(False)

tk.Label(

header, text="❤️ AbueFit", bg=COL\_SIDEBAR, fg="white",

font=("Segoe UI", 18, "bold"), anchor="w", padx=1

).pack(fill="both", expand=True)

# Breadcrumb

crumb = tk.Frame(sidebar, bg=COL\_SIDELITE, height=36)

crumb.pack(fill="x")

crumb.pack\_propagate(False)

tk.Label(

crumb, text="🏠 Inicio", bg=COL\_SIDELITE, fg="white",

font=("Segoe UI", 10, "bold")

).pack(side="left", padx=(12,0))

tk.Label(

crumb, text="›", bg=COL\_SIDELITE, fg="white",

font=("Segoe UI", 12, "bold")

).pack(side="right", padx=12)

# Botones sidebar

def info(title, msg): messagebox.showinfo(title, msg, parent=win)

btn\_opts = dict(

master=sidebar, bd=1, relief="ridge", cursor="hand2",

bg=COL\_BTN\_BG, fg=COL\_BTN\_FG, activebackground="#FFE0B0",

font=("Segoe UI", 10), height=2

)

# Mis pasos

if open\_win\_steps:

tk.Button(\*\*btn\_opts, text="Mis pasos",

command=lambda: open\_win\_steps()).pack(fill="x", padx=12, pady=(10,8))

else:

tk.Button(\*\*btn\_opts, text="Mis pasos",

command=lambda: info("Mis pasos", "Módulo no disponible en este entorno.")

).pack(fill="x", padx=12, pady=(10,8))

# Resto de opciones

tk.Button(\*\*btn\_opts, text="BMI",

command=lambda: open\_win\_canvas(parent)).pack(fill="x", padx=12, pady=8)

tk.Button(\*\*btn\_opts, text="Frase motivacional",

command=lambda: open\_win\_frase(parent)).pack(fill="x", padx=12, pady=8)

tk.Button(\*\*btn\_opts, text="Coach (chat)",

command=lambda: open\_win\_coach(parent)).pack(fill="x", padx=12, pady=8)

tk.Button(\*\*btn\_opts, text="Consejos",

command=lambda: open\_win\_consejos(parent)).pack(fill="x", padx=12, pady=8)

# Música (búsqueda y reproducción directa)

if open\_win\_musica:

tk.Button(\*\*btn\_opts, text="🎵 Música",

command=lambda: open\_win\_musica(parent)).pack(fill="x", padx=12, pady=8)

else:

tk.Button(\*\*btn\_opts, text="🎵 Música",

command=lambda: info("Música", "Instala/crea app/win\_musica.py para activar esta sección.")

).pack(fill="x", padx=12, pady=8)

# Mini-player (play/pausa/siguiente/anterior/estado)

if open\_win\_player:

tk.Button(\*\*btn\_opts, text="🎶 Control de música",

command=lambda: open\_win\_player(parent)).pack(fill="x", padx=12, pady=8)

else:

tk.Button(\*\*btn\_opts, text="🎶 Control de música",

command=lambda: info("Control de música", "Instala/crea app/win\_player.py para activar esta sección.")

).pack(fill="x", padx=12, pady=8)

# ===== Panel derecho: card de bienvenida =====

right\_pad = tk.Frame(content, bg=COL\_PANEL\_BG)

right\_pad.pack(fill="both", expand=True)

card = tk.Frame(right\_pad, bg="#FFF6E6", bd=0, highlightthickness=0)

card.pack(fill="x", padx=12, pady=(16, 10))

saludo\_lbl = tk.Label(

card, text="", bg="#FFF6E6", fg=COL\_TEXT\_ORG,

font=("Segoe UI", 12, "bold"), anchor="w", justify="left"

)

fecha\_lbl = tk.Label(

card, text="", bg="#FFF6E6", fg="#7C4A00",

font=("Segoe UI", 10), anchor="w", justify="left"

)

clima\_lbl = tk.Label(

card, text="", bg="#FFF6E6", fg="#7C4A00",

font=("Segoe UI", 10, "bold"), anchor="w", justify="left"

)

# Colocación vertical

saludo\_lbl.pack(anchor="w", padx=12, pady=(10, 0))

fecha\_lbl.pack( anchor="w", padx=12, pady=(2, 0))

clima\_lbl.pack( anchor="w", padx=12, pady=(4, 10))

def \_rellenar():

saludo\_lbl.config(text=f"👋 {saludo\_por\_hora(NOMBRE\_USUARIO)}")

fecha\_lbl.config(text=fecha\_compacta(datetime.datetime.now()))

clima\_lbl.config(text=clima\_leon\_texto())

def \_ajustar\_wrap(\_evt=None):

w = max(card.winfo\_width() - 24, 100)

saludo\_lbl.config(wraplength=w)

fecha\_lbl.config(wraplength=w)

clima\_lbl.config(wraplength=w)

card.bind("<Configure>", \_ajustar\_wrap)

win.after(60, lambda: (\_rellenar(), \_ajustar\_wrap()))

# Pie decorativo

homebar = tk.Frame(content, bg=COL\_BG\_APP, height=8)

homebar.pack(side="bottom", pady=12)

tk.Frame(homebar, bg="#C7A883", height=4, width=120).pack()

# Escape para cerrar

win.bind("<Escape>", lambda e: win.destroy())