

■ Documentación Técnica — Bioclima 3D App

Esta documentación describe el **código** y el **flujo completo** de la aplicación **Bioclima 3D App** construida con **Streamlit**.

1) Arquitectura general

Tecnologías principales

- Streamlit → interfaz web reactiva.
- folium + streamlit-folium → mapa interactivo.
- matplotlib → render del modelo 3D y gráficas.
- requests + Retry → cliente HTTP con tolerancia a fallos (Open-Meteo).
- pandas + numpy → manejo de series y agregación.

Archivo principal

- `app.py` contiene **toda la aplicación**.

Bloques lógicos

1. Configuración inicial y constantes (`MESES`).
2. Estado global (`busy`) para bloquear UI durante cálculos.
3. Funciones auxiliares:
 - `series_dummy`
 - `alturas_conceptuales`
 - `plot_modelo_3d`
 - `line_chart`
 - `requests_retry_session`
 - `fetch_open_meteo_monthly`
4. Interfaz (sidebar, mapa, botones).
5. Flujo principal con loader/spinner.

2) Flujo funcional

1. **Mapa** → usuario selecciona ubicación (click) o edita lat/lon manual.
2. **Fuente de datos**:
 - CSV (si existe).
 - Open-Meteo (si está activo y no hay CSV).
 - Serie dummy (fallback).
3. **Acción** → botón “Diseño Bioclimático”:
 - Se activa loader (spinner + bloqueo UI).
 - Se cargan datos (CSV/OpenMeteo/Dummy).
 - Se calculan alturas normalizadas.
 - Se genera modelo 3D y gráficas.
 - Se habilita descarga PNG.
 - UI se desbloquea.
4. **Reintento** → botón fuerza nueva descarga con mismos parámetros.

3) Funciones principales

`series_dummy(n=12, lat=0.0)`

Genera series de prueba (senoidales) ajustadas levemente por latitud.

`alturas_conceptuales(tmax, viento, radiacion=None)`

- Fórmula: $h = tmax + 0.5 * viento + 0.1 * radiacion$

- Normalización: $[0-100]$

`plot_modelo_3d(...)`

- Crea torres (base y tapa) con Poly3DCollection.

- Parámetros:

- `paso`: separación vertical.

- `escala`: factor altura.

- `torre_xy`: tamaño base.

`line_chart(values, title, ylabel)`

- Gráfica 2D simple para series mensuales.

`requests_retry_session(...)`

- Cliente HTTP con reintentos automáticos.

`fetch_open_meteo_monthly(lat, lon, year)`

- Descarga diario ERA5 → agrega mensual (`resample("MS")`).

- Devuelve arrays de 12 valores para `tmax`, `tmin`, `viento`, `rad`.

4) Interfaz de usuario

Sidebar

- Entrada CSV.

- Sliders elev/azim cámara.

- Fuente de datos (checkbox + año).

- Parámetros 3D (paso, escala, base).

Mapa

- Folium con popup de coordenadas.

- Inputs numéricos de lat/lon.

Acciones

- Botones: “Diseño Bioclimático”, “Reintentar descarga”.

Salidas

- Modelo 3D.

- Gráficas de tmax, viento, radiación, alturas.

- Botón descarga PNG.

5) Estado y loader

- Variable `st.session_state["busy"]` bloquea la UI.
- Mientras está activo:
- Inputs/mapa deshabilitados.
- Spinner visible.
- Al finalizar: éxito/error + desbloqueo.

6) Validaciones

- CSV inválido → error y stop.
- Fallo Open-Meteo → advertencia y dummy.
- Timeouts → reintentos automáticos.

7) Extensiones posibles

- Soporte a NASA POWER / Meteostat.
- Guardar datos procesados en CSV.
- Exportar modelo a DXF/SVG.
- Añadir capas urbanas (OSMnx).
- Cierre de caras laterales 3D.

8) Checklist

- Python 3.9+
- Instalar dependencias (`requirements.txt`).
- Ejecutar: `streamlit run app.py`
- Probar con CSV de ejemplo y Open-Meteo.