

Manual de usuario

Introducción

El Dashboard de Vacunación de Fiebre Amarilla es una herramienta interactiva diseñada para la Secretaría de Salud del Tolima. Permite visualizar y analizar datos de vacunación contra la fiebre amarilla en todo el departamento.

Navegación principal

El dashboard se divide en cinco pestañas principales:

1. **Visión General:** Resumen de las principales métricas de vacunación
2. **Distribución Geográfica:** Análisis por municipios del departamento
3. **Perfil Demográfico:** Distribución por edades, sexo y grupos étnicos
4. **Aseguramiento:** Análisis por régimen de afiliación y aseguradoras
5. **Tendencias:** Evolución temporal de la vacunación

Uso de filtros

Los filtros se encuentran en la barra lateral izquierda:

1. **Fuente de datos:** Seleccione entre datos de población DANE o SISBEN
2. **Filtros de datos:** Puede filtrar por:
 - Municipio
 - Grupo de edad
 - Sexo
 - Grupo étnico
 - Régimen de afiliación
 - Aseguradora

Para aplicar un filtro, seleccione el valor deseado en el menú desplegable correspondiente. Para quitar todos los filtros, use el botón "Restablecer Filtros".

Interpretación de gráficos

Los principales tipos de gráficos son:

- **Gráficos de barras:** Muestran comparativas entre diferentes categorías
- **Gráficos circulares:** Representan la distribución porcentual
- **Gráficos de dispersión:** Muestran la relación entre dos variables
- **Gráficos de líneas:** Visualizan tendencias temporales

Al pasar el cursor sobre los gráficos, aparecerá información detallada.

Documentación técnica

Arquitectura del sistema

El dashboard está desarrollado en Python utilizando Streamlit como framework principal. La arquitectura se divide en:

1. **Capa de datos:** Módulos para carga, procesamiento y filtrado de datos
2. **Capa de visualización:** Módulos para creación de gráficos y mapas
3. **Capa de interfaz:** Vistas organizadas por secciones temáticas

Estructura de archivos

```
dashboard-vacunacion-fa-tolima/
├── app.py                # Punto de entrada principal
├── data/                 # Directorio para archivos de datos
│   ├── POBLACION.xlsx   # Datos de población por municipio
│   └── vacunacion_fa.csv # Datos de vacunación
├── assets/              # Recursos estáticos
│   ├── images/          # Imágenes (logo, iconos)
│   └── styles/           # Archivos CSS
├── src/                 # Código fuente
│   ├── data/            # Módulos de procesamiento de datos
│   │   ├── loader.py    # Carga de datos
│   │   ├── normalize.py # Normalización de datos
│   │   └── preprocessor.py # Preprocesamiento y filtrado
│   ├── utils/           # Utilidades generales
│   │   └── helpers.py   # Funciones auxiliares
│   └── visualization/   # Módulos de visualización
│       ├── charts.py    # Gráficos estándar
│       └── maps.py      # Visualizaciones geográficas
└── vistas/              # Módulos de interfaz de usuario
    ├── overview.py      # Vista general
    ├── geographic.py    # Vista geográfica
    ├── demographic.py   # Vista demográfica
    ├── insurance.py     # Vista de aseguramiento
    └── trends.py        # Vista de tendencias
```

Flujo de datos

1. **Carga inicial:** `load_datasets()` en `loader.py` carga los archivos desde el sistema local o Google Drive.
2. **Normalización:** Se normalizan nombres de columnas y se convierten tipos de datos.

3. **Filtrado:** Cuando el usuario aplica filtros, `apply_filters()` en `preprocessor.py` procesa los datos.
4. **Visualización:** Cada vista utiliza funciones de `charts.py` para generar gráficos.

Guía de mantenimiento y solución de problemas

Mantenimiento regular

1. **Actualización de datos:**
 - Los archivos pueden actualizarse simplemente reemplazando los existentes
 - Para actualizar vía Google Drive, actualice los archivos con los mismos IDs
2. **Actualización de dependencias:**
 - Revise periódicamente `requirements.txt` para actualizaciones

Solución de problemas comunes

1. **Error de carga de datos:**
 - Verifique que los archivos estén en el formato correcto
 - Compruebe las rutas y los permisos de los archivos
 - Revise los IDs de Google Drive si utiliza esa opción
2. **Errores de tipo de datos:**
 - Los errores más comunes ocurren cuando hay columnas categóricas
 - Verifique y modifique `normalize_dataframe()` según sea necesario
3. **Problemas de rendimiento:**
 - Verifique el tamaño de los archivos, considere utilizar muestras más pequeñas para pruebas
 - Ajuste los parámetros de caché en las funciones decoradas con `@st.cache_data`
4. **Errores de visualización:**
 - Compruebe que los datos filtrados no estén vacíos
 - Verifique la consistencia de nombres de columnas

Recomendaciones para futuras versiones

1. **Mejoras de rendimiento:**
 - Implementar pre-agregaciones para análisis comunes
 - Optimizar la estructura de datos para filtrados más rápidos
 - Añadir más caching estratégico
2. **Mejoras de visualización:**
 - Añadir mapas interactivos más detallados
 - Implementar paneles de control personalizables
 - Añadir opciones de exportación de datos y gráficos
3. **Funcionalidades adicionales:**
 - Sistema de alertas para coberturas bajas
 - Predicciones de tendencias futuras
 - Comparativas con años anteriores

- Integraciones con otros sistemas de salud