

## Caso de Estudio N°3

Este caso de estudio será desarrollado como un **proyecto aplicado**, cuyo entregable principal será una **aplicación interactiva construida en Python utilizando Streamlit**, orientada al **Análisis Exploratorio de Datos (EDA)** del dataset **InsuranceCompany.csv**.

El objetivo **NO es construir modelos predictivos**, sino **aplicar de manera integrada los conceptos vistos a lo largo del curso**, desarrollando una herramienta funcional, clara y bien estructurada, similar a un producto analítico real.

**Objetivos específicos del proyecto**

Desarrollar una aplicación app.py ejecutable con **Streamlit**.

Aplicar conceptos fundamentales de:

- Variables y tipos de datos
- Funciones
- f-strings
- Programación Orientada a Objetos (POO)
- NumPy y Pandas
- Visualización con Matplotlib y Seaborn
- Estadística descriptiva

Diseñar una **interfaz ordenada, intuitiva y profesional**, utilizando sidebar, tabs, columns y widgets interactivos.

Publicar el proyecto como **primer proyecto de portafolio profesional**.

**Estructura obligatoria de la aplicación (Streamlit)**

La aplicación deberá estar organizada en **módulos navegables**, usando **sidebar** como menú principal.

**Módulo 1: Home (Presentación del proyecto)**

Este módulo deberá contener:

- Título del proyecto
- Breve descripción del objetivo del análisis
- Datos del autor:
  - Nombre completo
  - Curso / Especialización
  - Año
- Breve explicación del dataset
- Tecnologías utilizadas (Python, Pandas, Streamlit, etc.)

Este módulo no realiza análisis, solo contextualiza y presenta el proyecto.

## Módulo 2: Carga del dataset

Antes de ejecutar cualquier análisis, la aplicación **DEBE**:

- Utilizar `st.file_uploader()` para cargar el archivo `.csv`
- Validar que el archivo fue cargado correctamente
- Mostrar una vista previa del dataset (`head`)
- Mostrar dimensiones del dataset (filas y columnas)

**Ningún análisis debe ejecutarse si el archivo no ha sido cargado**

## Análisis Exploratorio de Datos (EDA)

Este módulo será el **núcleo del proyecto** y deberá estar organizado usando **tabs y columns**, mostrando una herramienta de análisis bien elaborada.

**Se deberán desarrollar 10 ítems de análisis, como mínimo**

Cada ítem debe estar **claramente titulado, explicado y visualizado**.

### Ítem 1: Información general del dataset

- `.info()`
- Tipos de datos
- Conteo de valores nulos

### Ítem 2: Clasificación de variables

- Identificación de variables:
  - Numéricas
  - Categóricas
- Uso de una **función personalizada**
- Mostrar resultados con conteo

### Ítem 3: Estadísticas descriptivas

- Uso de `.describe()`
- Interpretación básica de medias, medianas y dispersión

### Ítem 4: Análisis de valores faltantes

- Conteo
- Visualización simple (si aplica)
- Discusión breve

### Ítem 5: Distribución de variables numéricas

- Histogramas
- Uso de Matplotlib o Seaborn
- Interpretación visual

#### Ítem 6: Análisis de variables categóricas

- Conteos
- Gráficos de barras
- Proporciones

#### Ítem 7: Análisis bivariado (numérico vs categórico)

Ejemplos:

- income vs renewal
- agent\_effort\_hours vs renewal

#### Ítem 8: Análisis bivariado (categórico vs categórico)

Ejemplos:

- residence\_area\_type vs renewal
- sourcing\_channel vs renewal

#### Ítem 9: Análisis basado en parámetros seleccionados

- Uso de selectbox, multiselect
- Análisis dinámico según columnas elegidas por el usuario

#### Ítem 10: Hallazgos clave

- Visualización resumen
- Insights principales derivados del EDA

#### Uso obligatorio de widgets Streamlit

A lo largo del proyecto se deberá evidenciar el uso de:

- st.sidebar
- st.tabs
- st.columns
- st.selectbox
- st.multiselect
- st.slider
- st.checkbox

#### Uso de Programación Orientada a Objetos (POO)

Se deberá implementar al menos **una clase**, por ejemplo:

**DataAnalyzer / DataProcessor**, que encapsule:

- Estadísticas descriptivas
- Clasificación de variables
- Funciones de visualización

### Conceptos estadísticos mínimos a aplicar

- Media
- Mediana
- Moda (cuando aplique)
- Distribución
- Comparación de grupos

### Conclusiones finales

El proyecto deberá incluir una sección con:

- **5 conclusiones claras**, basadas en los análisis realizados
- Redacción propia, técnica pero comprensible
- Enfoque en **toma de decisiones**, no en predicción

### Entregables obligatorios

#### Repositorio GitHub

El repositorio deberá contener:

- app.py
- requirements.txt
- README.md **personalizado**
- Dataset

El README debe incluir:

- Descripción del proyecto
- Capturas de la app
- Instrucciones de ejecución
- Links relevantes

#### Aplicación desplegada

- Publicada en **Streamlit Cloud**
- Dominio funcional y accesible

### Documento final (PDF)

El estudiante deberá entregar un **PDF** que incluya:

- **Link al repositorio GitHub**
- **Link a la aplicación desplegada**
- Reflexión final sobre el aprendizaje

### Criterios de evaluación

- Correcta aplicación de conceptos del curso
- Orden, estructura y presentación
- Calidad del análisis
- Uso adecuado de Streamlit
- Personalización del proyecto
- Claridad del README
- Funcionamiento correcto del aplicativo

### Contexto del dataset

Variable	Descripción
<b>id</b>	Identificador único del cliente o póliza
<b>perc_premium_paid_by_cash_credit</b>	Porcentaje de la prima pagada mediante efectivo o crédito
<b>age_in_days</b>	Edad del cliente expresada en días
<b>Income</b>	Ingreso mensual del cliente
<b>Count_3-6_months_late</b>	Número de pagos demorados entre 3 y 6 meses
<b>Count_6-12_months_late</b>	Número de pagos demorados entre 6 y 12 meses
<b>Count_more_than_12_months_late</b>	Número de pagos demorados más de 12 meses
<b>application_underwriting_score</b>	Puntuación de evaluación del cliente (nivel de riesgo o confiabilidad)
<b>no_of_premiums_paid</b>	Número total de primas pagadas
<b>sourcing_channel</b>	Canal de captación del cliente (por ejemplo, A, B, C, D)
<b>residence_area_type</b>	Tipo de área de residencia (urbana o rural)
<b>premium</b>	Valor monetario de la prima del seguro
<b>renewal</b>	Indicador si el cliente renovó la póliza ("Yes"/"No")