

# Informe Técnico de Proyecto Final:

## La Inflación en el Plato

**Autor:** Alejandro Manuel de la Torre Almarales

**Asignaturas:** Introducción a la Ciencia de Datos  
e Introducción a la Programación

10 de diciembre de 2025

## Índice

<b>1. Introducción y Problemática</b>	<b>2</b>
<b>2. Adquisición de Datos</b>	<b>2</b>
2.1. Datos In-Situ (Trabajo de Campo) . . . . .	2
2.2. Datos Oficiales (ONEI) . . . . .	2
2.3. Datos Digitales (Redes Sociales) . . . . .	2
2.4. Tasa Cambiaria (El Toque) . . . . .	3
<b>3. Estructura de Datos</b>	<b>3</b>
3.1. Estructura de Datos Online (db_online.json) . . . . .	3
3.2. Estructura de Datos In-Situ (db_in_situ.json) . . . . .	4
<b>4. Procesamiento y Normalización</b>	<b>4</b>
4.1. Normalización de Unidades . . . . .	4
4.2. Fusión de Fuentes (Merge) . . . . .	5
4.3. Limpieza de Fechas . . . . .	5
<b>5. Herramientas Utilizadas</b>	<b>5</b>

# 1. Introducción y Problemática

La economía cubana vive momentos de mucha inestabilidad. La inflación se siente a diario y el dinero rinde cada vez menos, pero a veces los números oficiales tardan en reflejar lo que pasa realmente en la calle, especialmente en los mercados privados e informales.

Este proyecto nace de una pregunta simple pero difícil de responder: **¿Cuánto cuesta realmente vivir en La Habana hoy?**

El objetivo fue rastrear el costo de la vida desde Enero de 2024 hasta Noviembre de 2025, no solo para ver cuánto han subido los precios, sino para entender su relación con el dólar informal y ver qué tanto golpea esto al salario de un trabajador estatal. Básicamente, ponerle datos duros a la experiencia cotidiana de ir al mercado.

## 2. Adquisición de Datos

Para que el análisis fuera serio, no podía quedarme con una sola fuente. Tuve que triangular información de tres lugares distintos: lo que dice el gobierno, lo que dice internet y lo que dicen los precios físicos en la tarima.

### 2.1. Datos In-Situ (Trabajo de Campo)

Esta fue la parte más activa del proyecto. Visité personalmente varias MIPYMES y puntos de venta privados en La Habana.

- **El proceso:** En cada lugar, pedí permiso a los dependientes para tomar fotos de los productos y sus precios.
- **Validación:** No anoté precios de memoria; todo está respaldado por evidencia fotográfica que incluye la fecha, la hora exacta y la ubicación del lugar para garantizar que los datos sean reales.

### 2.2. Datos Oficiales (ONEI)

Utilicé los informes públicos de la Oficina Nacional de Estadística e Información (ONEI) para tener un punto de comparación. De aquí digitalicé las tablas de salarios medios por sectores (2024-2025) y los rangos de precios oficiales (mínimos y máximos) de los productos básicos.

### 2.3. Datos Digitales (Redes Sociales)

Hice un monitoreo constante de grupos de compra-venta en Facebook, que es donde se mueve gran parte del comercio informal.

- **Metodología:** Tomé capturas de pantalla de publicaciones de venta de alimentos y aseo.

- **Organización:** Para no perder el rastro temporal, codifiqué la fecha de cada oferta directamente en el nombre del archivo de la imagen antes de procesarla.

## 2.4. Tasa Cambiaria (El Toque)

Necesitaba el histórico del dólar informal, pero copiarlo a mano era inviable. Desarrollé una solución automatizada:

- **Ingeniería Inversa:** Inspeccioné el tráfico de red del navegador mientras visitaba la web de El Toque para encontrar su API interna.
- **Scraping:** Con el token de acceso y un script en Python (usando `requests`), descargué un JSON con el histórico diario de dos años. Luego limpié manualmente los datos de 2023 que no eran relevantes para este estudio.

## 3. Estructura de Datos

Elegí trabajar con archivos **JSON** porque son mucho más flexibles que una tabla de Excel tradicional. Me permitieron guardar estructuras complejas (como listas de productos dentro de una fecha específica) y se integran nativamente con los diccionarios de Python.

### 3.1. Estructura de Datos Online (`db_online.json`)

Aquí agrupé todo por fecha. Usé listas paralelas para productos, precios y unidades. Esto ahorra espacio y hace muy fácil recorrer los datos del día.

```
{
    "2025-11-21": {
        "products": [ "aceite", "paquete muslo de pollo", "picadillo mdm" ],
        "prices": [ "1300", "3800", "9000" ],
        "unit": [ "1L", "10lb", "15kg" ]
    },
    "2025-11-22": { ... }
}
```

### 3.2. Estructura de Datos In-Situ (db\_in\_situ.json)

Los datos de campo tienen más detalles. La clave principal es el nombre del negocio, y dentro guarda la geolocalización, la hora y el municipio, además de la lista de precios.

```
{  
    "Carales Supermercado": {  
        "date": "2025-10-30",  
        "time": "15:40",  
        "geolocation": "23.1508751,  
                      -82.3062920",  
        "township": "Habana del Este"  
    },  
    "products_info": {  
        "Products": [ "  
                      galletas  
                      rellenitas", "  
                      cerveza presidente  
                      " ],  
        "Prices": [ "120", "  
                   250" ],  
        "Units": [ "paquete",  
                   "lata" ]  
    },  
    "exchange_rate": "470"  
},  
    "Mercado Toledo": { ... }  
}
```

## 4. Procesamiento y Normalización

Como los datos venían de fuentes tan distintas, tuve que crear un módulo de funciones propias (Functions.py) para poner orden. Los mayores retos fueron:

### 4.1. Normalización de Unidades

Este fue el problema más común: unos venden en libras, otros en kilos, otros por "paquete". Para poder comparar peras con peras, estandaricé todo a una unidad base.

- **La lógica:** Calculé el precio unitario real dividiendo el precio total entre la cantidad.

$$P_{norm} = \frac{P_{bruto}}{Cantidad_{unidad}}$$

- Por ejemplo: Si una botella de 500ml cuesta 600 CUP, mi código la convierte a 1200 CUP/Litro para poder graficarla junto a las demás.

## **4.2. Fusión de Fuentes (Merge)**

Creé funciones para mezclar los datos de la ONEI, los de internet y los de la calle en una sola "lista maestra". Así, a la hora de hacer los gráficos, no importa de dónde vino el dato, todos tienen el mismo formato.

## **4.3. Limpieza de Fechas**

Estandaricé todas las fechas al formato ISO (YYYY-MM-DD). Para los análisis de tendencias a largo plazo, agrupé los datos por mes, eliminando el ruido diario que dificultaba ver la tendencia general.

# **5. Herramientas Utilizadas**

Para construir todo esto utilicé:

- **Lenguaje:** Python 3.10+.
- **Entorno:** Visual Studio Code.
- **Control de Versiones:** Git y GitHub.
- **Librerías Clave:**
  - `collections`: Para estructuras de datos eficientes.
  - `plotly`: Para crear los gráficos interactivos.
  - `requests`: Para conectarme a la API de El Toque.