

## Práctica 03 – Fila en un supermercado

Asignatura: Simulación • Unidad 1 • Equipo de 3 estudiantes

### 1. Datos del equipo:

Equipo N.º	Grupo I
Integrantes	César López, Dilan Chamba y Luis Armijos
Tema	Simulación de las filas de un supermercado
Fecha	25/10/2025

### 2. Propósito:

Determinar en qué caja debemos ingresar para salir más rápido, considerando tanto el número de personas en fila como la cantidad de artículos que tienen. Posteriormente, agregar una "Caja Express" que tiene un límite de 10 artículos por cliente, y validar si realmente es más eficiente o si el tiempo de interacción con el cliente la vuelve más lenta.

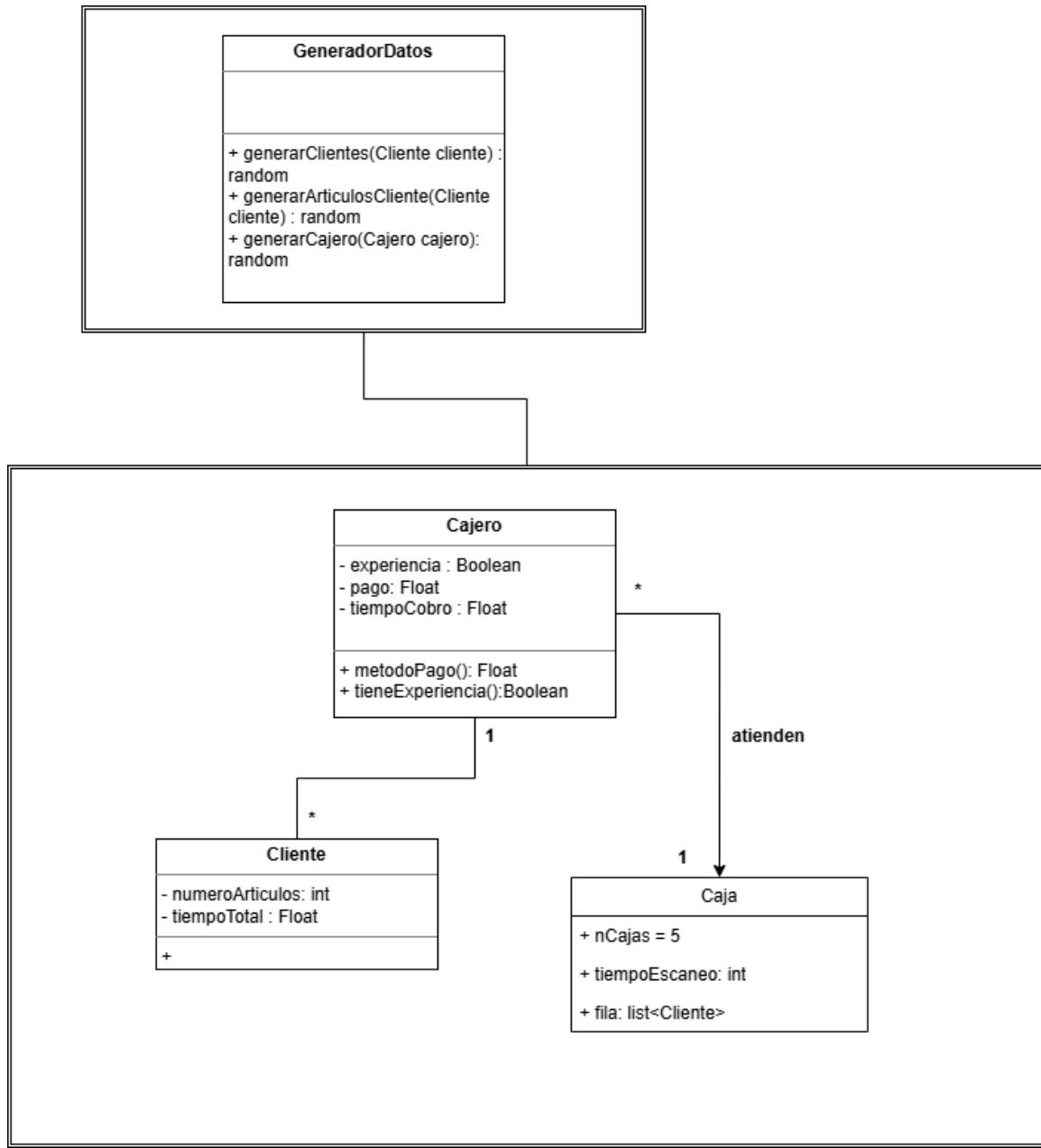
### 3. Instrucciones del ejercicio:

#### 3.1. Configurar la simulación: Usar el lenguaje de programación de su preferencia para modelar el siguiente problema, se recomienda el uso de Python, pero la decisión es del estudiante:

- Cantidad de personas en fila para cada caja. (Se puede ingresar por teclado para ejecutar diferentes casos de prueba controlados)
- Número de artículos por persona (aleatorio entre 1 y 50).
- Tiempo de escaneo por artículo, esto puede ser una constante por cada caja por ejemplo 5 segundos por artículo para un cajero normal, o 9 segundos para un cajero principiante etc.
- Tiempo de cobro (aleatorio entre 15 y 30 segundos, dependiendo del tipo de pago).

### 3.1.1. Planificación inicial.

Con todo el grupo creamos un diagrama UML, el cual lo tomamos como una guía, fue nuestra primera lluvia de ideas, dando el siguiente resultado:



Esta fue nuestra base, dividimos en dos carpetas la lógica de negocio inicial, implementamos y fuimos depurando.

### **3.1.2. Clases Identificadas:**

- Cliente: Representa a una persona con artículos para comprar
- Cajero: Empleado que atiende en una caja
- Caja: Estación de atención al cliente
- GeneradorDatos: Utilidad para crear datos aleatorios

### **3.1.3. Atributos y Métodos Planificados:**

- Cliente: numeroArticulos, tiempoTotal
- Cajero: experiencia, tiempoEscaneoPorArticulo, tiempoCobro
- Caja: idCaja, cajero, esExpress, filaClientes, tiempoAtencionTotal, LIMITE\_EXPRESS

### **3.1.4. Desarrollo de la Solución**

Estructura Final del Proyecto

supermercado/

```
|── main.py          # Archivo principal que ejecuta la simulación  
|── README.md       # Documentación del proyecto  
|── display/  
|   |── interfaz.py    # Interfaz gráfica con animación en tiempo real  
|── models/  
|   |── cliente.py     # Clase Cliente  
|   |── cajero.py      # Clase Cajero  
|   |── caja.py        # Clase Caja  
└── simulation/  
    └── generadorDatos.py # Generador de datos aleatorios
```

### 3.1.5. Implementación de Clases

Cliente (models/[cliente.py](#))

```
class Cliente:
    def __init__(self, numeroArticulos: int):
        self.numeroArticulos = numeroArticulos # 1-50 artículos
        self.tiempoTotal = 0.0 # Tiempo total en ser atendido, inicializado en 0.0 segundos
```

Cajero (models/[cajero.py](#))

```
class Cajero:
    def __init__(self, tieneExperiencia: bool):
        self.experiencia = tieneExperiencia

        if self.experiencia:
            self.tiempoEscaneoPorArticulo = 3 # segundos por artículo para cajero con experiencia
        else:
            self.tiempoEscaneoPorArticulo = 6 # segundos por artículo para cajero sin experiencia
        self.tiempoCobro = random.randint(15, 30) # tiempo de cobro aleatorio entre 15 y 30 segundos
```

Caja (models/caja.py)

```
class Caja:
    def __init__(self, idCaja: int, cajero: Cajero, esExpress : bool, filaInicial : List[Cliente] = None):
        self.idCaja = idCaja
        self.cajero = cajero
        self.esExpress = esExpress
        self.filaClientes = filaInicial if filaInicial is not None else [] # si no hay fila inicial, se crea una vacía
        self.tiempoAtencionTotal = 0.0 # tiempo total de atención en la caja
        self.LIMITE EXPRESS = 10 # Límite de artículos para cajas express
```

### 3.1.6. Lógica de Simulación

#### Configuración Inicial

# Parámetros configurables

num\_cajeros = 5 # 3-8 cajeros

num\_clientes = 25 # 10-50 clientes

posicion\_express = "primera" # "primera", "medio", "ultima", "aleatoria"

#### Asignación Inteligente de Clientes

La asignación sigue una lógica realista que simula el comportamiento de los clientes en un supermercado:

1. Clientes con pocos artículos ( $\leq 10$ ): Intentan automáticamente la caja express, ya que es más eficiente para compras pequeñas.
2. Clientes con muchos artículos ( $> 10$ ): No pueden usar la caja express y se asignan a cajas normales disponibles.
3. Selección de caja normal: Si hay múltiples cajas normales, se selecciona aleatoriamente entre las disponibles para distribuir la carga.

### 3.1.7. Código implementado:

```
for cliente in clientes:  
    asignado = False  
  
    # Paso 1: Clientes pequeños prefieren express  
  
    if cliente.numeroArticulos <= 10:  
        # Cliente puede usar cualquier caja (express o normal)  
        cajas_validas = cajas  
  
        else:  
            # Cliente solo puede usar cajas normales  
            cajas_validas = [caja for caja in cajas if not caja.esExpress]  
  
    # Paso 2: Clientes grandes van a cajas normales  
  
    while not asignado and cajas_validas:  
        caja_aleatoria = random.choice(cajas_validas)  
  
        if caja_aleatoria.agregarCliente(cliente):  
            asignado = True  
  
        else:  
            # Si no se pudo asignar, remover de la lista para evitar loop infinito  
            cajas_validas.remove(caja_aleatoria)
```

**3.2. Cálculo del tiempo total en cada caja:** Para cada persona en cada caja, deben calcular el tiempo total de atención (escaneo de artículos + tiempo de cobro). El tiempo total para cada caja será la suma de los tiempos de atención de todas las personas en fila.

```
def calcularTiempoAtencion(self) -> float:  
    self.tiempoAtencionTotal = 0.0  
  
    #obtenemos la velocidad y tiempo del cajero asignado a la caja  
    tiempoEscaneoPorArticulos = self.cajero.tiempoEscaneoPorArticulo  
    tiempoCobros = self.cajero.tiempoCobro  
  
    tiempoEsperaAcumulado = 0.0 # tiempo de espera acumulado para clientes posteriores  
  
    #recorremos la fila  
    for cliente in self.filaClientes:  
        tiempoEscaneoPorCliente = cliente.numeroArticulos * tiempoEscaneoPorArticulos  
        tiempoAtencionPorCliente = tiempoEscaneoPorCliente + tiempoCobros  
        cliente.tiempoTotal = tiempoAtencionPorCliente + tiempoEsperaAcumulado # tiempo total incluye espera  
        tiempoEsperaAcumulado += tiempoAtencionPorCliente # acumulamos el tiempo de atención para el siguiente cliente  
    self.tiempoAtencionTotal += tiempoAtencionPorCliente # acumulamos el tiempo total de atención  
return self.tiempoAtencionTotal
```

### 3.2.1. Variables Principales

- **tiempoEsperaAcumulado:** Acumula el tiempo de espera para clientes posteriores.
- **tiempoAtencionPorCliente:** Tiempo que toma atender a un cliente específico.
- **cliente.tiempoTotal:** Tiempo total que el cliente pasa en el sistema.
- **self.tiempoAtencionTotal:** Suma total de tiempos de atención de todos los clientes.

### 3.2.2. Resultados (En consola).

```
Caja 1 (Normal) atendida por Cajero sin experiencia: Tiempo de escaneo por articulo = 6s, Tiempo de cobro = 16s con 8 clientes en fila. Tiempo total de atención: 1040.00s  
Clientes en fila:  
    Cliente con 41 artículos., Tiempo total en ser atendido: 262.00s  
    Cliente con 14 artículos., Tiempo total en ser atendido: 362.00s  
    Cliente con 24 artículos., Tiempo total en ser atendido: 522.00s  
    Cliente con 11 artículos., Tiempo total en ser atendido: 604.00s  
    Cliente con 21 artículos., Tiempo total en ser atendido: 746.00s  
    Cliente con 11 artículos., Tiempo total en ser atendido: 828.00s  
    Cliente con 13 artículos., Tiempo total en ser atendido: 922.00s  
    Cliente con 17 artículos., Tiempo total en ser atendido: 1040.00s
```

**3.3. Comparar el tiempo en cada caja:** Determinar cuál es la mejor opción para un nuevo cliente que quiere salir más rápido.

### 3.3.1. Criterio de Evaluación

La comparación se basa en el **tiempoAtencionTotal** de cada caja, que representa el tiempo total que tomaría atender a todos los clientes actualmente en fila.

### 3.3.2. Proceso de Selección

1. Evaluación de todas las cajas: Se compara el **tiempoAtencionTotal** de cada caja activa
2. Selección del mínimo: Se elige la caja con el menor tiempo acumulado
3. Recomendación: Se presenta la caja más rápida al usuario

### 3.3.3. Resultado de Ejemplo

```
==== RECOMENDACIÓN PARA NUEVO CLIENTE ====
La caja más rápida para un nuevo cliente es la Caja 1 (Express)
Tiempo total de atención actual: 55.00s
Clientes en fila: 2
```

**3.4. Agregar una “Caja Express”:** Limitar el número de artículos a 10 por persona en la caja express, y repetir el proceso para verificar si realmente es más rápida o si, debido a la cantidad de personas (normalmente tiene más gente), resulta ser más lenta que una caja regular con menos personas con carritos más llenos.

Se implementó una caja especial con restricciones para evaluar si realmente mejora la eficiencia del sistema o si el mayor volumen de clientes la hace menos efectiva.

#### 3.4.1. Características de la Caja Express

Restricciones Implementadas

LIMITE\_EXPRESS = 10 # artículos máximo por cliente

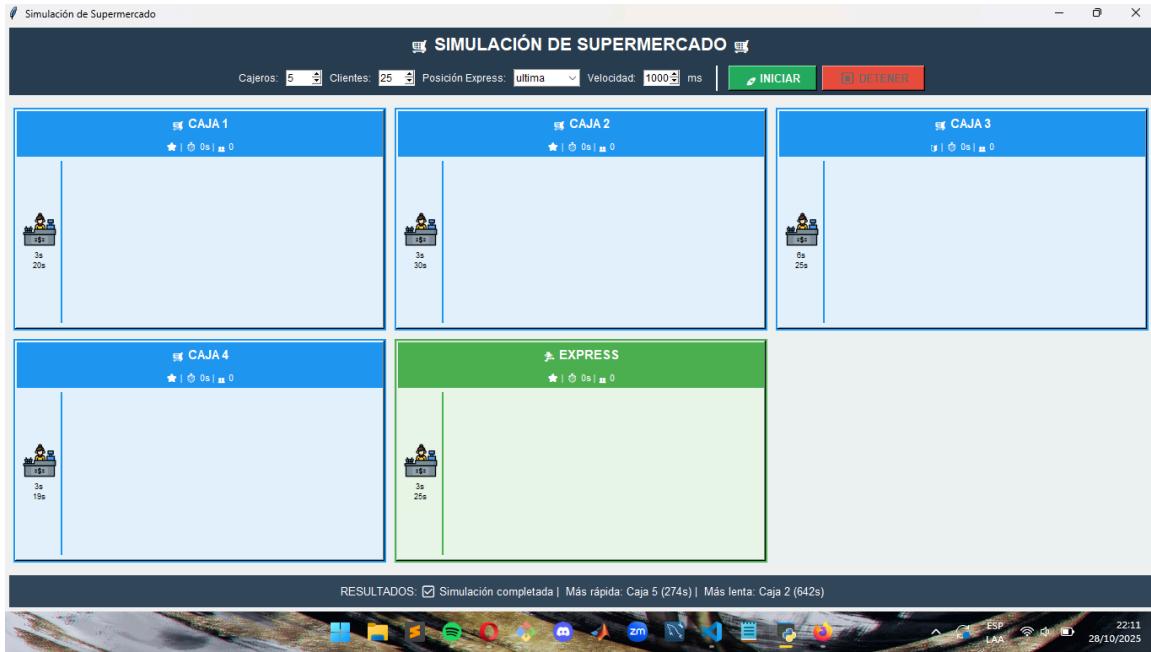
Método de Validación en models/[caja.py](#)

```
def agregarCliente(self, cliente: Cliente):
    if self.esExpress and cliente.numeroArticulos > self.LIMITE_EXPRESS:
        print(f"El cliente no puede usar la caja express {self.idCaja}.")
        return False
    else:
        self.filaClientes.append(cliente)
        return True
```

### 3.5. Representación visual de la simulación:

- Dibujar cada caja como un bloque o espacio en la pantalla.
- Mostrar el número de personas en fila en cada caja (como íconos, avatares o puntos).
- A medida que los clientes son atendidos, hacer que desaparezcan de la fila y mostrar una animación que represente su salida y/o el avance de todo el resto de la fila.
- Mostrar la caja express con un color distinto para facilitar su identificación.

Para la representación visual hemos decidido usar Tkinter, siendo esta la biblioteca estándar de Python para crear interfaces gráficas de usuario. Con esto en mente, en nuestra interfaz grafica propusimos que la caja express salga con el color verde para hacerla fácilmente identifiable y las cajas normales de color azul, ademas cabe destacar que para cada caja se mostrara una simbología las cuales son: símbolo de personas para representar el numero de clientes, reloj para el tiempo que demoran cobrando por cada usuario y la estrella o bandera para representar la experiencia que posee el cajero, siendo la estrella para aquellos que tienen experiencia y la bandera para aquellos que son novatos. Así pues, al ejecutar la simulación se presentará la siguiente ventana:



En donde la barra superior servirá para configurar el número de cajeros y clientes, la posición en la que deseemos que se encuentre la caja express y la velocidad de la simulación. Además que podremos inicializar una nueva simulación con el botón de iniciar y detenerla con el de pausa.



Al finalizar la simulación se mostrará un mensaje mostrando cuál fue la caja en despachar más rápido a los clientes y cuál fue la más lenta.

RESULTADOS:  Simulación completada | Más rápida: Caja 1 (156s) | Más lenta: Caja 2 (883s) | Tiempos finales: C1E:156s | C2N:883s | C3N:484s | C4N:660s | C5N:732s

## 4. Pruebas de simulación

### 4.1. Primeras pruebas de simulación

#### Caso de prueba 1

Clientes: 30

Cajeros: 4 (3 normales + 1 express)

Caja express: acepta solo clientes con  $\leq 10$  artículos

Artículos por cliente: aleatorios (1 a 50)

Prueba	Caja	Tipo	Experiencia	Total Artículos	Nº Clientes	Tiempo Total (S)
1	4	Express	sin	9	3	144
2	4	Express	con	11	3	99
3	4	Express	sin	10	1	85

4	1	Normal	con	211	11	886
5	2	Normal	con	298	13	1154
6	4	Express	con	6	1	42
7	4	Express	sin	26	4	224
8	4	Express	con	14	2	86
9	1	Normal	con	235	10	915
10	4	Express	con	8	2	70

### Conclusiones del caso:

- Caja Express es eficiente para pocos clientes y artículos: tiempos totales muy bajos comparados con cajas normales.
- La experiencia del cajero reduce el tiempo significativamente: prueba 1 (sin) 144 s vs prueba 2 (con) 99 s.
- Cuando hay más artículos que el límite de Express, el tiempo aumenta (prueba 7: 26 artículos, 224 s, sin experiencia).
- Cajetas normales manejan más clientes y artículos, por eso sus tiempos son mucho mayores.
- Para salir rápido, conviene elegir Express si el número de artículos  $\leq 10$  y preferentemente con un cajero con experiencia.

### Caso de prueba 2

Cajas: 3 (1 express, 2 normales)

Clientes: 30

Artículos por cliente: aleatorio 5–30

Cajeros: todos con experiencia

Prueba	Caja	Tipo	Experiencia	Nº Clientes	Total Artículos	Tiempo Total
1	3	Express	con	1	10	54
2	3	Express	con	1	6	37
3	3	Express	con	1	7	40
4	3	Express	con	1	7	42
5	3	Express	con	2	12	82
6	3	Express	con	4	33	183
7	3	Express	con	1	9	57
8	3	Express	sin	4	32	212
9	3	Express	con	2	15	87
10	3	Express	con	2	15	75

### Caso de prueba 3

Clientes: 30

Cajeros: 2 (1 normal + 1 express)

Artículos por cliente: aleatorios (1 a 50)

Prueba	Caja	Tipo	Experiencia	Total Artículos	Nº Clientes	Tiempo Total
1	2	Express	con	13	3	105
2	2	Express	con	21	3	123
3	2	Express	con	7	3	78
4	2	Express	con	12	2	78
5	2	Express	con	25	4	151
6	2	Express	con	18	3	102
7	2	Express	con	20	3	105
8	2	Express	con	41	5	243
9	2	Express	sin	10	5	243
10	2	Express	sin	10	2	112

#### Conclusiones del caso:

- Cajero con experiencia reduce significativamente el tiempo de atención.
- Caja express es útil solo si los clientes cumplen el límite de artículos.
- Cuando los clientes tienen muchos artículos, el tiempo de la express se dispara y puede ser igual o peor que una normal.
- Para salir rápido, conviene:
- Elegir express solo con  $\leq 10$  artículos.
- Priorizar cajeros con experiencia.

#### Caso de prueba 4

Cajas: 3 (1 express, 2 normales)

Clientes: 30

Artículos por cliente: aleatorio 10–30

Cajeros: todos sin experiencia

Objetivo: probar la eficiencia de la express cuando muchos clientes exceden el límite de 10 artículos.

Ejecución	Caja	Tipo	Experiencia	Total Artículos	Nº Clientes	Tiempo Total
1	1	Normal	sin	208	11	1512.00
1	2	Normal	sin	398	19	2749.00
1	3	Express	sin	0	0	0.00
2	1	Normal	sin	339	16	2402.00
2	2	Normal	sin	253	13	1726.00
2	3	Express	sin	10	1	85.00
3	1	Normal	sin	273	14	1904.00
3	2	Normal	sin	378	16	2652.00
3	3	Express	sin	0	0	0.00
4	1	Normal	sin	316	14	2316.00
4	2	Normal	sin	308	15	2088.00

4	3	Express	sin	10	1	75.00
5	1	Normal	sin	318	17	2180.00
5	2	Normal	sin	254	13	1745.00
5	3	Express	sin	0	0	0.00
6	1	Normal	sin	292	15	2172.00
6	2	Normal	sin	312	15	2202.00
6	3	Express	sin	0	0	0.00
7	1	Normal	sin	255	12	1854.00
7	2	Normal	sin	296	18	2082.00
7	3	Express	sin	0	0	0.00
8	1	Normal	sin	286	14	2010.00
8	2	Normal	sin	285	16	1982.00
8	3	Express	sin	0	0	0.00
9	1	Normal	sin	350	17	2389.00
9	2	Normal	sin	241	13	1823.00
9	3	Express	sin	0	0	0.00
10	1	Normal	sin	344	20	2384.00
10	2	Normal	sin	193	10	1438.00
10	3	Express	sin	0	0	0.00

#### Conclusiones del caso:

- Dado que la mayoría de los clientes tenía más de 10 artículos, la caja express fue ineficiente o inútil en estas pruebas.
- Los cajeros sin experiencia aumentaron los tiempos totales de atención, pero no afectaron el reparto (todas las cajas tuvieron rendimientos similares).

#### Caso de prueba 5

**Cajas:** 3 (1 express, 2 normales)

**Clientes:** 30

**Artículos por cliente:** aleatorio entre 1 y 15

**Cajeros:** todos con experiencia

Ejecución	Caja	Tipo	Experiencia	Total Artículos	Nº Clientes	Tiempo Total
1	1	Normal	con	117	13	585.00
1	2	Normal	con	116	11	656.00
1	3	Express	con	49	6	279.00
2	1	Normal	con	69	9	369.00

2	2	Normal	con	134	15	687.00
2	3	Express	con	43	6	249.00
3	1	Normal	con	98	11	514.00
3	2	Normal	con	117	14	757.00
3	3	Express	con	35	5	185.00
4	1	Normal	con	130	12	570.00
4	2	Normal	con	84	12	456.00
4	3	Express	con	22	6	240.00
5	1	Normal	con	100	14	552.00
5	2	Normal	con	140	12	696.00
5	3	Express	con	17	4	135.00
6	1	Normal	con	82	10	446.00
6	2	Normal	con	118	13	731.00
6	3	Express	con	38	7	219.00
7	1	Normal	con	72	10	486.00
7	2	Normal	con	100	12	552.00
7	3	Express	con	35	8	257.00
8	1	Normal	con	95	12	465.00
8	2	Normal	con	119	11	687.00
8	3	Express	con	39	7	320.00
9	1	Normal	con	98	10	474.00
9	2	Normal	con	156	14	678.00
9	3	Express	con	41	6	285.00
10	1	Normal	con	110	14	750.00

10	2	Normal	con	86	9	501.00
10	3	Express	con	35	7	273.00

#### Conclusiones del caso:

- La caja express fue consistentemente la más rápida en todas las simulaciones.
- A pesar de tener entre 4 y 8 clientes, su tiempo total nunca superó los 320 segundos, lo que confirma su efectividad con pocos artículos.
- Las dos cajas normales atendieron un número de clientes y artículos bastante equilibrado, mostrando que la asignación fue corrector y no hubo sobrecarga en una sola caja  
Gracias a la experiencia de los cajeros y el uso adecuado de la caja express, el tiempo total global se mantuvo bajo y estable entre todas las simulaciones.

#### 4.2. Segundas pruebas de simulación

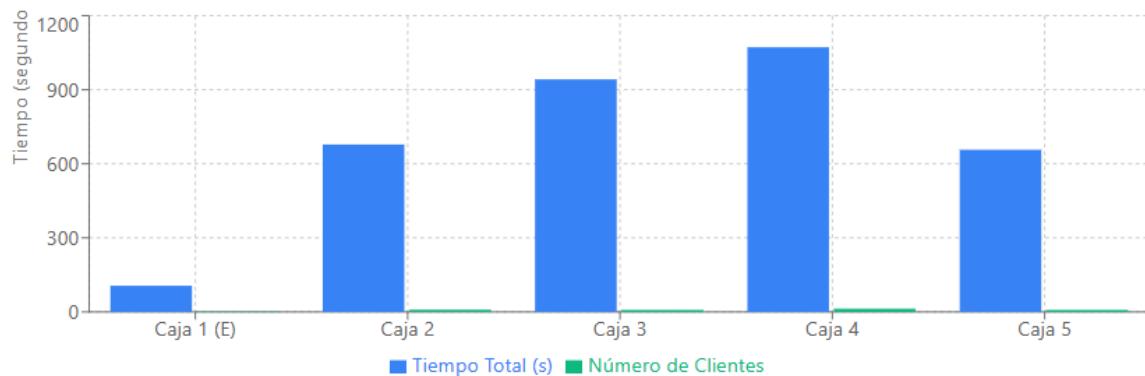
##### Caso de prueba 1 - Hora pico con alta demanda

Se evaluaron 5 cajas con 40 clientes.

Caja	Tipo	Cajero	Clientes	Artículos Totales	Tiempo Total (s)	Tiempo Promedio/Cliente (s)
Caja 1	Express ( $\leq 10$ )	Novato	2	11	106	53
		Escaneo: 6s   Cobro: 20s				
Caja 2	Normal	Experto	9	160	678	75
		Escaneo: 3s   Cobro: 22s				
Caja 3	Normal	Novato	8	133	942	118
		Escaneo: 6s   Cobro: 18s				
Caja 4	Normal	Experto	13	236	1072	82
		Escaneo: 3s   Cobro: 28s				
Caja 5	Normal	Experto	8	139	657	82

		Escaneo: 3s   Cobro: 30s				
--	--	-----------------------------	--	--	--	--

### Gráfica de resultados



### Conclusiones del caso

- Recomendación para nuevo cliente: La caja más rápida es la Caja 1 - Express con 106 segundos de tiempo acumulado.
- Eficiencia de caja Express: La caja express es 87% más eficiente que el promedio de cajas normales.
- Factor crítico: El tiempo de atención no solo depende del tipo de caja, sino también de: la experiencia del cajero, novato, el tiempo de cobro aleatorio, 20s, y la cantidad de clientes en fila, 2 clientes.
- Validación del límite de 10 artículos: La caja express atendió 2 clientes con un promedio de 6 artículos por cliente.
- Observación general: La estrategia óptima para un cliente es elegir la caja con el menor tiempo acumulado, considerando tanto la cantidad de personas como sus artículos, independientemente de si es express o normal.

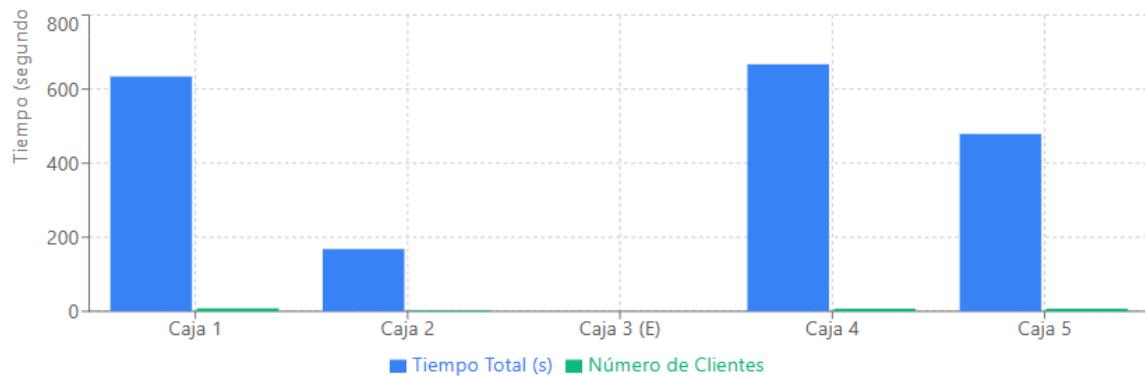
### Caso de prueba 2 - Hora normal con una demanda intermedia

Se evaluaron 5 cajas con 25 clientes.

Caja	Tipo	Cajero	Clientes	Artículos Totales	Tiempo Total (s)	Tiempo Promedio/Cliente (s)
Caja 1	Normal	Novato	8	83	634	79
		Escaneo: 6s   Cobro: 17s				
Caja 2	Normal	Experto	3	30	168	56

		Escaneo: 3s   Cobro: 26s				
Caja 3	Express (≤10)	Novato	0	0	0	0
		Escaneo: 6s   Cobro: 16s				
Caja 4	Normal	Experto	7	157	667	95
		Escaneo: 3s   Cobro: 28s				
Caja 5	Normal	Experto	7	99	479	68
		Escaneo: 3s   Cobro: 26s				

### Gráfica de resultados



### Conclusiones del caso

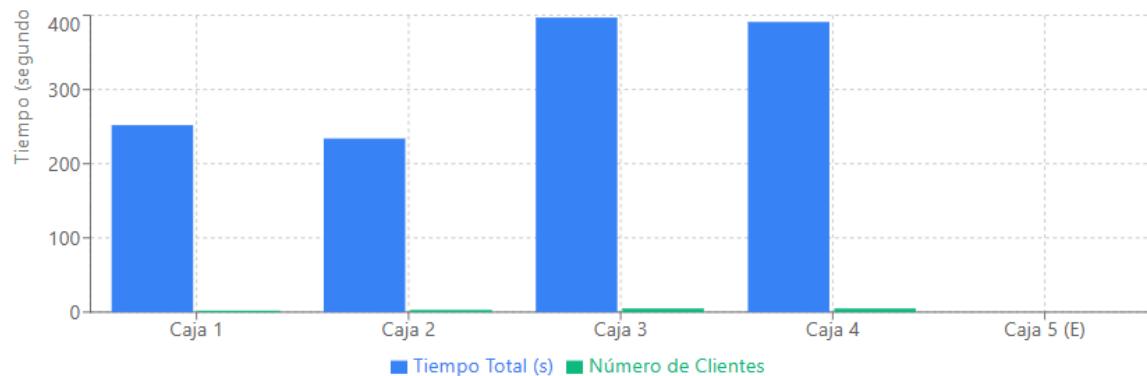
- Recomendación para nuevo cliente: La caja más rápida es la Caja 3 - Express con 0 segundos de tiempo acumulado.
- Eficiencia de caja Express: La caja express es 100% más eficiente que el promedio de cajas normales.
- Factor crítico: El tiempo de atención no solo depende del tipo de caja, sino también de: la experiencia del cajero, novato, el tiempo de cobro aleatorio, 16s, y la cantidad de clientes en fila, 0 clientes.
- Validación del límite de 10 artículos: La caja express no recibió clientes en esta simulación.
- Observación general: La estrategia óptima para un cliente es elegir la caja con el menor tiempo acumulado, considerando tanto la cantidad de personas como sus artículos, independientemente de si es express o normal.

### Caso de prueba 3 - Hora baja con poca demanda

Se evaluaron 5 cajas con 15 clientes.

Caja	Tipo	Cajero	Clientes	Artículos Totales	Tiempo Total (s)	Tiempo Promedio/Cliente (s)
Caja 1	Normal	Novato	2	34	252	126
		Escaneo: 6s   Cobro: 24s				
Caja 2	Normal	Experto	3	61	234	78
		Escaneo: 3s   Cobro: 17s				
Caja 3	Normal	Experto	5	99	397	79
		Escaneo: 3s   Cobro: 20s				
Caja 4	Normal	Experto	5	92	391	78
		Escaneo: 3s   Cobro: 23s				
Caja 5	Express ( $\leq 10$ )	Novato	0	0	0	0
		Escaneo: 6s   Cobro: 17s				

### Gráfica de resultados



### Conclusiones del caso

- Recomendación para nuevo cliente: La caja más rápida es la Caja 5 - Express con 0 segundos de tiempo acumulado.
- Eficiencia de caja Express: La caja express es 100% más eficiente que el promedio de cajas normales.
- Factor crítico: El tiempo de atención no solo depende del tipo de caja, sino también de: la experiencia del cajero, novato, el tiempo de cobro aleatorio, 17s, y la cantidad de clientes en fila, 0 clientes.
- Validación del límite de 10 artículos: La caja express no recibió clientes en esta simulación.
- Observación general: La estrategia óptima para un cliente es elegir la caja con el menor tiempo acumulado, considerando tanto la cantidad de personas como sus artículos, independientemente de si es express o normal.

#### Caso de prueba 4 - La caja express pierde

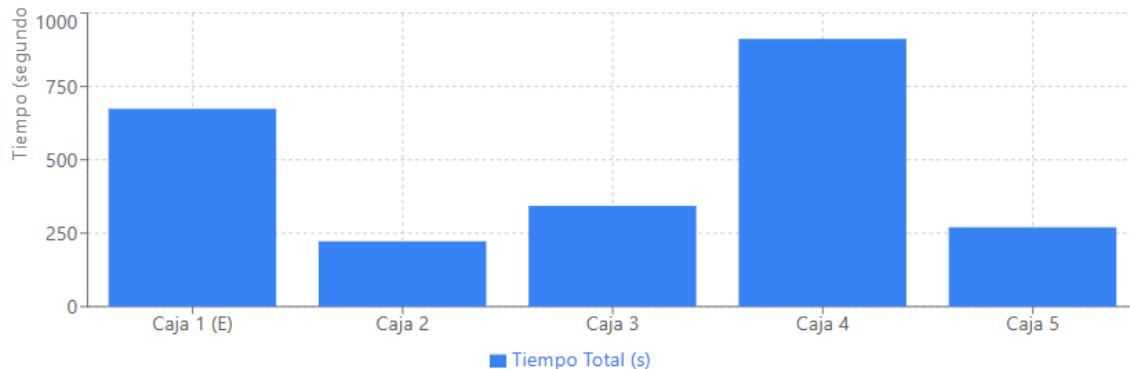
Se evaluaron 5 cajas con 26 clientes.

La caja express tiene muchos clientes, 8, con un cajero inexperto, 6s/art, y tiempo de cobro alto, 28s, mientras que una caja normal tiene solo 3 clientes con cajero experto, 3s/art, y cobro rápido, 16s.

Caja	Tipo	Cajero	Clientes	Artículos Totales	Artículos Promedio	Tiempo Total (s)
Caja 1	Express (≤10)	Novato	8	75	9.4	674s
		Escaneo: 6s/art				
		Cobro: 28s				
Caja 2	Normal	Experto	3	58	19.3	222s
		Escaneo: 3s/art				
		Cobro: 16s				
Caja 3	Normal	Experto	5	81	16.2	343s
		Escaneo: 3s/art				
		Cobro: 20s				
Caja 4	Normal	Novato	6	128	21.3	912s
		Escaneo: 6s/art				
		Cobro: 24s				
Caja 5	Normal	Experto	4	66	16.5	270s

		Escaneo: 3s/art				
		Cobro: 18s				

### Gráfica de resultados



### Conclusiones del caso

Recomendación para nuevo cliente: La caja más rápida es la Caja 2 – Normal, con 222 segundos de tiempo acumulado y 3 clientes en fila.

Rendimiento de la caja Express: La caja express perdió en este escenario con 674s, siendo 54% más lenta que el promedio de cajas normales, 437s. Esto se debe a que tiene un cajero novato con tiempo de cobro de 28s atendiendo 8 clientes.

Factores determinantes: El tiempo total no depende solo del tipo de caja, Express vs Normal, sino de:

- Experiencia del cajero; Experto: 3s/art, Novato: 6s/art.
- Tiempo de cobro aleatorio, 15-30 segundos.
- Cantidad de clientes en fila.
- Número de artículos por cliente.

Validación del límite de 10 artículos: La caja express atendió 8 clientes con un promedio de 9 artículos por cliente, todos cumpliendo el límite de 10 artículos. En este caso, a pesar de tener el límite, el cajero novato y el tiempo de cobro alto la hicieron ineficiente.

Conclusión general del caso 4 - express pierde: Este caso demuestra que la caja express no siempre es más rápida. Cuando tiene un cajero inexperto y tiempo de cobro alto, puede ser superada por cajas normales con menos clientes y cajeros eficientes. El tiempo de interacción, cobro, es crucial.

Estrategia óptima: Un cliente inteligente debe observar no solo el tipo de caja, sino el tiempo total acumulado considerando: número de personas en fila, artículos que llevan, y la velocidad aparente del cajero. La etiqueta "Express" no garantiza velocidad automáticamente.

## 5. Conclusiones

En este primer avance se logró definir una arquitectura clara del sistema mediante el uso de clases que representan los elementos principales del supermercado; Cliente, Cajero, Caja y Generador de Datos. Esta organización modular facilitó la comprensión del flujo de la simulación y permitió desarrollar una base sólida para futuras ampliaciones o mejoras del modelo.

Se consiguió representar de manera realista el comportamiento de los clientes al elegir cajas según el número de artículos, así como la influencia de la experiencia de los cajeros en los tiempos de atención. Los primeros resultados reflejan correctamente las diferencias entre cajas normales y la caja express, validando que la lógica implementada funciona conforme al planteamiento inicial.

La incorporación de una interfaz gráfica con Tkinter permitió visualizar la dinámica de las filas en tiempo real, mejorando la comprensión del proceso de atención y facilitando la detección de posibles mejoras. Los casos de prueba iniciales confirmaron que la caja express es eficiente sólo bajo ciertas condiciones, lo que motiva un análisis más profundo en los siguientes avances.