



Universidad Nacional Autónoma de México

Facultad de Ingeniería

Sistemas Operativos

Proyecto 2

Grupo 6

Hernández Gallardo Alberto Javier

Vázquez Gómez Carlos Iván

Fecha: 06/11/2023

Situación:

Nuestra situación es un sistema de inventario donde varios trabajadores (hilos) se van a actualizar el inventario de un almacén. Para esto nosotros utilizamos threading.

Condition nos va a permitir que los trabajadores esperen a que haya suficiente stock para despachar órdenes y un mecanismo de apagado suave que permita reiniciar o detener los hilos de manera segura.

Se resolvió de la siguiente manera:

Con una interfaz de Usuario (Principal)

Una ventana principal master con un título y botones para agregar productos, modificar el stock y salir de la aplicación.

Una gestión de inventario en el diccionario global inventario que almacena los nombres de los productos y sus cantidades asociadas.

La función agregar\_producto pide al usuario que ingrese el nombre de un producto a través de un cuadro de dialogo y lo agrega al inventario con cantidad 0.

Modificar Stock al hacer clic se abre una nueva ventana (TopLvel) con campos para introducir el nombre del producto y agregar o despachar, la función agregar stock toma el nombre y la cantidad de los campos de entrada y si el producto existe incrementa su cantidad.

despachar stock también toma el nombre y la cantidad y si hay stock suficiente reduce la cantidad del producto. también se actualiza la interfaz mediante actualizar\_inventario\_label,.

La gestión de concurrencia mediante un objeto llamado condition se utilizará para gestionar el acceso al inventario cuando se realizan cambios, esto será importante si se extiende la aplicación para incluir hilos que puedan modificar el inventario concurrentemente.

Al finalizar la 3era versión del programa nos dimos cuenta que el código no tenía concurrencia en sí, estaba la base

para añadir la concurrencia con Condition, pero no la estábamos manejando.

Por esta razón, en la 4ta versión, decidimos añadir una función donde se actualizará el inventario automáticamente en paralelo a como lo estábamos

realizando mediante la interfaz. Dicha función realiza el actualizado del inventario en "segundo plano" con otro hilo diferente al principal para mantener a estabilidad del programa.

Al añadir esto es necesario que la interfaz también se actualice de forma constante para mantener una consistencia entre el inventario y lo que se muestra en pantalla. Por esto también se añadió una función que actualiza la interfaz gráfica cada cierto tiempo, elegimos 5 segundos de manera aleatoria (el programador puede modificar este tiempo para que se ajuste a la necesidad del cliente).

Sin embargo, nuestra función para actualizar de manera automática, al ser una manera de ejemplificar la concurrencia, utilizaba un tiempo de actualización aleatorio entre un rango determinado de segundos. Esto complicaba el funcionamiento de nuestro programa porque podía haber errores de actualización entre el segundo plano y la interfaz gráfica donde quisiéramos despachar cierta cantidad de productos y que estos no cuadraran con lo que se veía en la pantalla.

Es por esto que decidimos eliminar este rango aleatorio de tiempo y lo modificamos para que quedara en 30 segundos. De esta forma, la interfaz

gráfica se actualiza cada 5 segundos y de manera automática el inventario hace la operación cada 30 segundos dando un tiempo razonable para que el usuario manipule el programa (ya sea añadiendo productos, agregando stock o despachando). Al ser un tiempo establecido por nosotros los programadores también se puede modificar en caso de que se necesite más tiempo para teclear y manipular la información de manera que sea clara para ejemplificar la concurrencia.

Utilización deseada:

Usuario inicia aplicación

Utiliza el botón "Agregar Producto" para introducir el producto al inventario

Modifica el stock de los productos existentes a través de la ventana emergente "Modificar Stock"

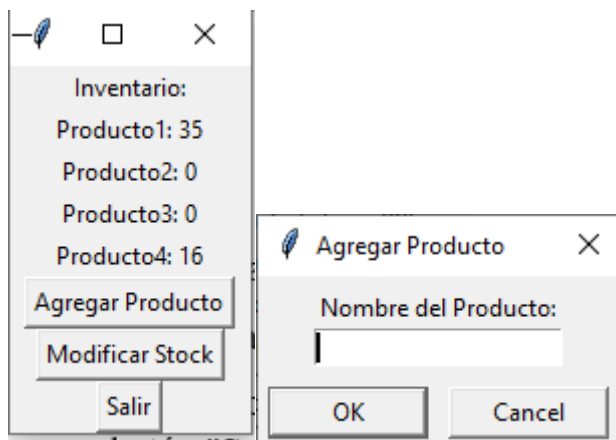
La interfaz reflejara los cambios en tiempo real

El usuario puede salir de la aplicación en cualquier momento utilizando el botón "Salir"

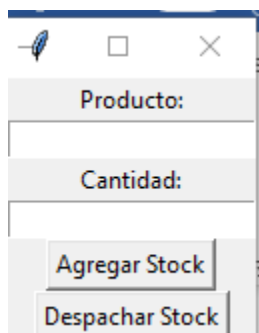
Se inicia el programa



Se agrega producto



Se modifica el stock



El stock de los productos se va actualizando cada 30 segundos

Fue diseñado en Python 7.19.0, es necesario tener instalada la biblioteca tkinter que nos proporciona la interfaz de usuario.

Por lo tanto, no se puede utilizar un python online. Además, se desarrolló en el sistema operativo Windows 10, solo se ha probado en este entorno.