
PROYECTO 1 “IPCMARKET”

202210483 – Angely Lucrecia García Martínez

Resumen

En la era digital, la implementación de sistemas electrónicos para la gestión de mercados se ha convertido en una necesidad fundamental. Este ensayo presenta el desarrollo de IPCmarket, una aplicación de escritorio diseñada para administrar un mercado electrónico.

El objetivo principal fue crear una interfaz gráfica de usuario (GUI) utilizando Python y la biblioteca Tkinter que facilitara la interacción tanto para administradores como para clientes, garantizando una experiencia de usuario intuitiva y eficiente.

La aplicación permite a los administradores gestionar de manera efectiva mediante archivos XML los productos, empleados y actividades de la empresa, así como la autenticación de usuarios. A su vez, los clientes pueden navegar por los productos, agregar artículos a su carrito de compras y realizar transacciones de manera segura teniendo siempre la oportunidad de visualizar los productos mediante diagramas ya que se ha implementado la biblioteca Graphviz para llevar a cabo la generación de reportes para llevar una mejor administración de la información que se ha cargado en el sistema.

Palabras clave

XML, usuarios, productos, empleados, actividades, Graphviz, reportes, compras.

Abstract

In the digital era, the implementation of electronic systems for market management has become a fundamental necessity. This essay presents the development of IPCmarket, a desktop application designed to manage an electronic market.

The main objective was to create a graphical user interface (GUI) using Python and the Tkinter library that would facilitate interaction for both administrators and customers, ensuring an intuitive and efficient user experience.

The application allows administrators to effectively manage products, employees, and company activities using XML files, as well as authenticate users. Meanwhile, customers can browse products, add items to their shopping cart, and complete transactions securely, always having the opportunity to visualize products through diagrams. This is made possible by the implementation of the Graphviz library to generate reports for better management of the information loaded into the system.

Keywords

XML, users, products, employees, activities, Graphviz, reports, purchases.

Introducción

IPCmarket, una aplicación de escritorio desarrollada para administrar un mercado electrónico, responde a esta necesidad, ofreciendo una solución tecnológica avanzada que optimiza tanto la gestión interna como la experiencia del cliente. Se presentará un panorama general de la aplicación, explicando sus bases teóricas y la perspectiva adoptada en su diseño y desarrollo. Además, se discutirán los aspectos relevantes que sustentan la argumentación, tales como la elección de Python y la biblioteca Tkinter para la creación de una interfaz gráfica de usuario intuitiva, el uso de archivos XML para la gestión de productos, empleados y actividades de la empresa y la implementación de Graphviz para la generación de reportes mediante diagramas en formato PNG.

Desarrollo del tema

IPCmarket fue desarrollado utilizando Python y la biblioteca Tkinter para la creación de la GUI. Además, se implementó la biblioteca Graphviz para la generación de diagramas y reportes visuales. Estas herramientas fueron seleccionadas por su capacidad de integrar funcionalidades avanzadas con una interfaz amigable para el usuario. Para tener acceso al sistema se tienen dos tipos de usuarios: el administrador encargado de la gestión de recursos y compras y el usuario quien tiene la oportunidad de ver los productos disponibles y realizar sus compras. A continuación, se detallan las funcionalidades de la aplicación.

1. Funcionalidades como Administrador

- a. Gestión de Productos: Los administradores pueden agregar al sistema productos mediante la carga de

archivos XML. Esta funcionalidad asegura que la información del inventario esté siempre actualizada.

- b. Gestión de Empleados y Actividades: La aplicación permite la administración de los empleados, incluyendo la asignación de roles y la gestión de sus actividades.
 - c. Autenticación de Usuarios: Se implementa un sistema de autenticación que permite que los únicos usuarios cargados al sistema tengan acceso a la aplicación.
- ### 2. Funcionalidades como Usuario
- a. Navegación de Productos: Los clientes pueden explorar los productos disponibles en el mercado, así como los detalles de los mismos: nombre, descripción, precio, categoría y la imagen del producto.
 - b. Visualización de productos: Mediante la implementación de Graphviz, los clientes pueden visualizar diagramas que representan las características de los productos que fueron agregados a su carrito de compras.
 - c. Realizar compras: Cuando el cliente ya ha agregado los productos a su compra, puede enviar su solicitud de compra al administrador quien la aceptará o rechazará.
- ### 3. Generación de reportes
- Una de las características distintivas de IPCmarket es la capacidad de generar reportes gráficos. Estos reportes proporcionan una visión clara y detallada de la información del mercado como los reportes de los usuarios autenticados, productos del inventario, los empleados y un reporte de las actividades que

tienen asignadas, solicitudes de compra y las compras realizadas.. La integración de Graphviz permite crear diagramas visualmente atractivos y fáciles de interpretar.

Para manejar y almacenar la información se utilizan las siguientes estructuras:

- 1. Lista doblemente enlazada: Empleada para el registro de los usuarios en el sistema junto a su información personal y sus credenciales.



Figura 1. Representación gráfica de la lista doblemente enlazada de usuarios.

Fuente: Elaboración propia, 2024

- 2. Lista circular doblemente enlazada: Empleada para la gestión de productos, facilitando un acceso continuo y cíclico a los mismos conteniendo los detalles de los productos.

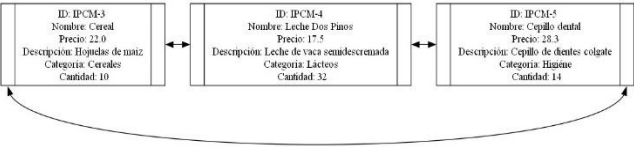


Figura 2. Representación gráfica de la lista circular doblemente enlazada de productos.

Fuente: Elaboración propia, 2024

- 3. Lista circular simplemente enlazada: Implementada para la administración de empleados, ofreciendo una estructura ligera y eficiente con el código, nombre y puesto del empleado en la empresa.

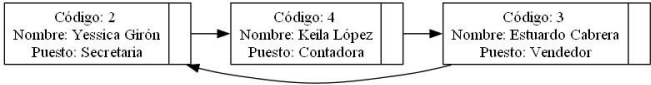


Figura 3. Representación gráfica de la lista circular simplemente enlazada de empleados.

Fuente: Elaboración propia, 2024

- 4. Matriz dispersa: Utilizada para la gestión de actividades de la empresa, optimizando la organización y planificación de tareas por hacer.

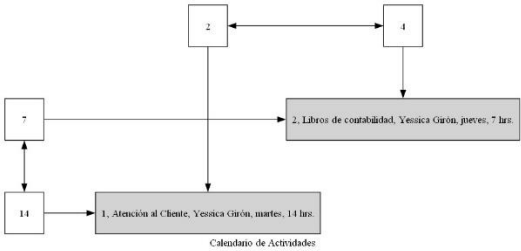


Figura 4. Representación gráfica de la matriz dispersa de actividades a realizar.

Fuente: Elaboración propia, 2024

- 5. Pila: Utilizada para el carrito de compras que contiene los productos que el usuario ha seleccionado, permitiendo un manejo LIFO (Last In, First Out) de los productos.

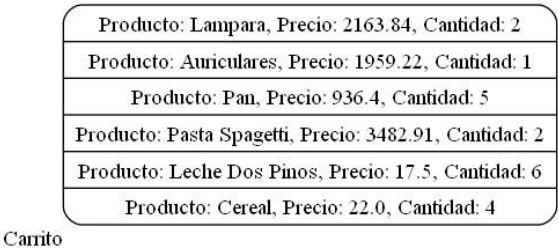


Figura 5. Representación gráfica de la pila como carrito de compras.

Fuente: Elaboración propia, 2024

6. Cola: Empleada para gestionar las solicitudes de los clientes, asegurando un procesamiento FIFO (First In, First Out).

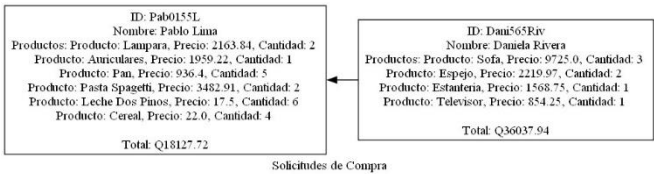


Figura 6. Representación gráfica de la cola con las solicitudes de compra

Fuente: Elaboración propia, 2024

7. Lista simplemente enlazada: Implementada para registrar las compras aceptadas, facilitando el control de las transacciones realizadas.

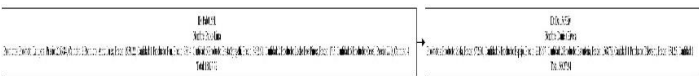


Figura 7. Representación gráfica de la lista simplemente enlazada con las compras autorizadas.

Fuente: Elaboración propia, 2024

Para el procesamiento de archivos XML, se necesita que esos tengan la siguiente estructura:

a. Archivo de usuarios

```
<?xml version="1.0" encoding="UTF-8"?>
<usuarios>
  <usuario id="UserID"
password="UserPassword">
    <nombre>Nombre del Usuario</nombre>
    <edad>Edad del Usuario</edad>
    <email>Email del Usuario</email>
    <telefono>Teléfono del
Usuario</telefono>
  </usuario>
  <!-- Más usuarios pueden ser añadidos
aquí -->
</usuarios>
```

Figura 8. Estructura de archivo XML de usuarios.

Fuente: Elaboración propia, 2024

b. Archivo de productos

```
<?xml version="1.0" encoding="UTF-8"?>
<productos>
  <producto id="ProductID">
    <nombre>Nombre del Producto</nombre>
    <precio>Precio del Producto</precio>
    <descripcion>Descripción del
Producto</descripcion>
    <categoria>Categoría del
Producto</categoria>
    <cantidad>Cantidad del
Producto</cantidad>
    <imagen>Ruta de la Imagen del
Producto</imagen>
```

Figura 9. Estructura de archivo XML de productos.

Fuente: Elaboración propia, 2024

c. Archivo de empleados

```
<?xml version="1.0" encoding="UTF-8"?>
<empleados>
  <empleado codigo="EmployeeCode">
    <nombre>Nombre del Empleado</nombre>
    <puesto>Puesto del Empleado</puesto>
  </empleado>
  <!-- Más empleados pueden ser añadidos
aquí -->
</empleados>
```

Figura 10. Estructura de archivo XML de empleados.

Fuente: Elaboración propia, 2024

d. Archivo de actividades

```
<?xml version="1.0" encoding="UTF-8"?>
<actividades>
  <actividad id="ActivityID">
    <nombre>Nombre de la
Actividad</nombre>
    <descripcion>Descripción de la
Actividad</descripcion>
    <empleado>EmpleadoID</empleado>
    <dia hora="Hora">Día</dia>
  </actividad>
  <!-- Más actividades pueden ser añadidas
aquí -->
</actividades>
```

Figura 11. Estructura de archivo XML de actividades.

Conclusiones

La creación de IPCmarket ha mejorado la organización y administración de un mercado electrónico, facilitando a los administradores la gestión de productos, empleados y actividades de manera eficiente.

IPCmarket ha proporcionado una experiencia de usuario intuitiva y segura para los clientes, permitiéndoles navegar por productos, realizar compras y visualizar información de manera sencilla y efectiva.

El hecho de no haber utilizado estructuras nativas de Python ha permitido organizar y manejar grandes volúmenes de datos de manera eficiente. Esto ha simplificado la implementación de funcionalidades como la adición, eliminación y búsqueda de elementos dentro de la aplicación al utilizar estructuras de datos bien definidas.

Referencias bibliográficas

Facultad de Ingeniería, USAC (2024). *Proyecto*.

Obtenido de:

https://uedi.ingenieria.usac.edu.gt/campus/pluginfile.php/795687/mod_resource/content/2/Proyecto1IPC2%20-%20Junio%202024.pdf

Elder, J. (s.f.). *Tkinter widget. Quick reference guide*.

Obtenido de:

https://uedi.ingenieria.usac.edu.gt/campus/pluginfile.php/792276/mod_resource/content/1/TkinterWidget s.pdf

Anexos

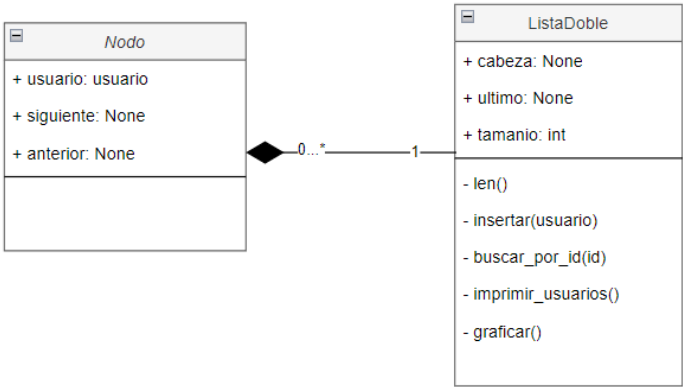


Figura 12. Modelo del tipo de dato ListaDoble

Fuente: Elaboración propia, 2024

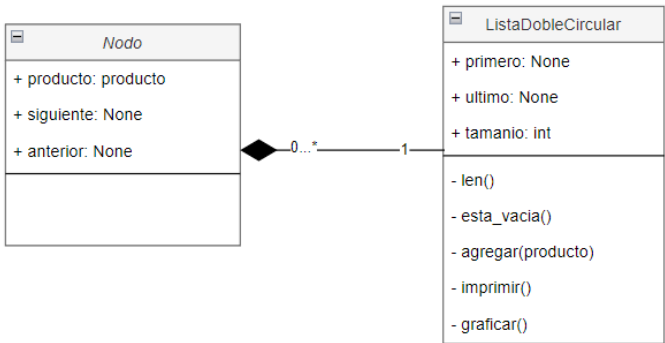


Figura 13. Modelo del tipo de dato ListaDobleCircular

Fuente: Elaboración propia, 2024

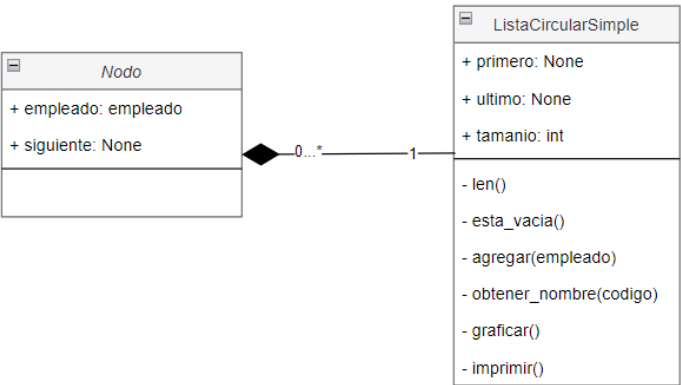


Figura 13. Modelo del tipo de dato ListaCircularSimple

Fuente: Elaboración propia, 2024

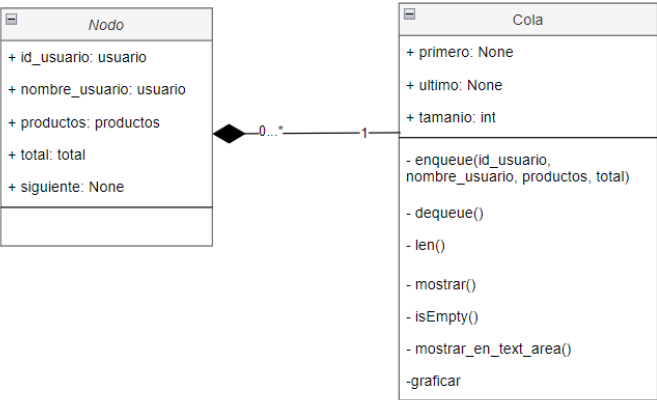


Figura 14. Modelo del tipo de dato Cola

Fuente: Elaboración propia, 2024

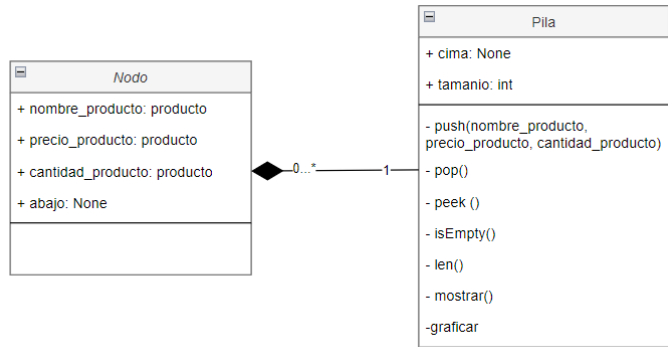


Figura 15. Modelo del tipo de dato Pila

Fuente: Elaboración propia, 2024

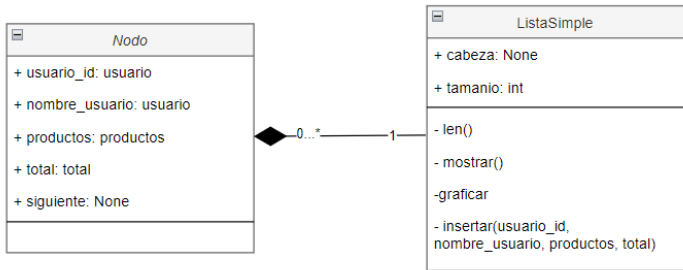


Figura 16. Modelo del tipo de dato ListaSimple

Fuente: Elaboración propia, 2024

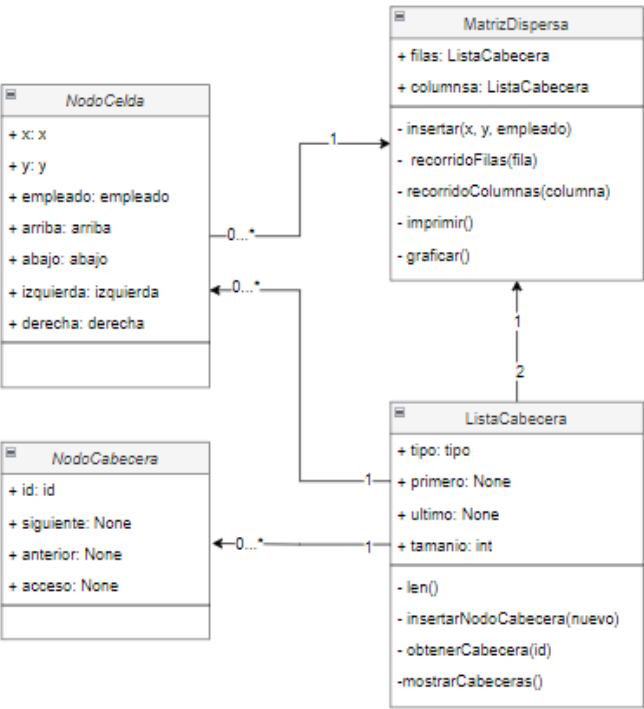


Figura 17. Modelo del tipo de dato MatrizDispersa

Fuente: Elaboración propia, 2024

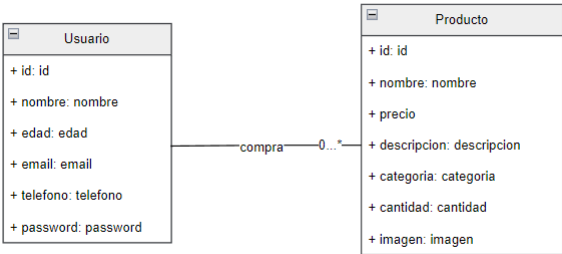


Figura 18. Modelo de asociación entre las clases Usuario y Producto.

Fuente: Elaboración propia, 2024

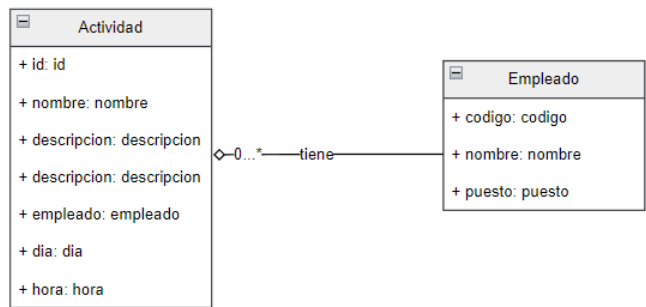


Figura 19. Modelo de agregación entre las clases Empleado y Actividad.

Fuente: Elaboración propia, 2024

Interfaz gráfica



Figura 20. Interfaz gráfica del inicio de sesión

Fuente: Elaboración propia, 2024



Figura 21. Interfaz gráfica de la ventana de administrador

Fuente: Elaboración propia, 2024



Figura 21. Interfaz gráfica de la ventana de usuario.

Fuente: Elaboración propia, 2024