
IPCART-STUDIO

202300694 – Eduardo Sebastian Gutierrez Felipe

Resumen

El siguiente ensayo presenta la descripción del desarrollo de la aplicación IPCArt-Studio, una aplicación de escritorio con interfaz gráfica amigable al usuario, la cual simula una plataforma dedicada a la creación y gestión de arte en píxeles. En esta plataforma existe un usuario Administrador, el cual es el encargado de ingresar todos los usuarios al sistema por medio de un archivo XML. Luego, los usuarios con rol de solicitante pueden solicitar a los artistas que creen su matriz de pixeles a través de la carga de un archivo XML que contiene la fila, columna y color de los pixeles de la imagen. Las solicitudes se enviarán a la cola de solicitudes de los artistas. Posteriormente, cuando el artista inicie sesión podrá ver la primera solicitud en cola, la cual puede aceptar, generando una matriz dispersa que enviará al solicitante como una imagen que el solicitante podrá ver en su galería.

Palabras clave

Cola, pila, listas, XML, clase

Abstract

The following essay presents the description of the development of the IPCArt-Studio app, a desktop application with a user-friendly graphic interface, which simulates a platform dedicated to the creation and management of pixel art. In this platform there is an Administrator user, who is in charge of uploading all users into the system through an XML file. Then, users with the requester role can request artists to create their pixel matrix by uploading an XML file containing the row, column and color of the pixels in the image. The requests will be sent to the artists' request queue. Later, when the artist logs in, he will be able to see the first request in the queue, which he can accept, generating a sparse matrix that will be sent to the requester as an image that the requester can see in his gallery.

Keywords

Queue, stack, lists, XML, class.

Introducción

Se desarrolló un programa que simula una aplicación de arte en píxeles. Para que el programa funcione debe existir un usuario Administrador, quien será el encargado de cargar los usuarios válidos para poder acceder al sistema. Los solicitantes serán los que carguen la imagen que desean a través de un archivo XML que contendrá la descripción de los píxeles que contiene la imagen.

Por otro lado, el artista al iniciar sesión podrá ver la cola de solicitudes y podrá aceptar el trabajo si lo desea. Al aceptar, se comenzará a generar una matriz dispersa según los píxeles del archivo XML que cargó el solicitante. Cuando se termine de crear la imagen, se enviará a la galería del usuario para que éste pueda verla en su perfil.

Todo el sistema se construyó utilizando estructuras de datos conocidas como TDA implementadas en el lenguaje de programación Python.

Desarrollo del tema

La app consiste en una simulación de una plataforma de pixel art (arte en píxeles) en la que un usuario solicita la creación de una imagen, y el artista es el encargado de crear el diseño en píxeles.

Esta solución fue elaborada en el lenguaje de programación Python ya que es un lenguaje multiparadigma que facilita la creación de interfaces gráficas de usuario y de estructuras de datos, así como la lectura de archivos XML.

Para que el sistema funcione, existe un usuario Administrador que es el encargado de cargar los usuarios que tendrán acceso al sistema. Existen dos tipos de usuarios, los solicitantes, que solicitan las

imágenes a partir de un archivo XML; y los artistas, quienes aceptan las solicitudes creando una matriz dispersa y generando el diseño en píxeles.

El XML para cargar a los solicitantes debe contener la siguiente estructura:

```
<?xml version="1.0" encoding="UTF-8" ?>
<solicitantes>
  <solicitante id="IPC-001" pwd="1234">
    <NombreCompleto>Juan Antonio Perez Silva</NombreCompleto>
    <CorreoElectronico>juanperez@gmail.com</CorreoElectronico>
    <NumeroTelefono>3436 3436</NumeroTelefono>
    <Direccion>Guatemala, ciudad de Guatemala</Direccion>
  </solicitante>
  ...
</solicitantes>
```

Figura 1. Archivo XML de solicitantes

Fuente: elaboración propia

donde:

solicitantes: es la raíz del archivo.

solicitante: es la etiqueta que contiene toda la información de un solicitante.

id: es el atributo ID único de cada usuario solicitante.

pwd: es el atributo que contiene la contraseña de la cuenta del solicitante.

NombreCompleto: es la etiqueta que contiene el nombre del solicitante.

CorreoElectronico: es la etiqueta que contiene el e-mail del solicitante.

NumeroTelefono: es la etiqueta que contiene el número telefónico del solicitante.

Direccion: es la etiqueta que contiene la dirección del domicilio del solicitante.

Por otro lado, la estructura que debe contener el XML de artistas debe ser como la siguiente:

```
<Artistas>
  <Artista id="ART-001" pwd="1234">
    <NombreCompleto>Eduardo Gustavo Fring López</NombreCompleto>
    <CorreoElectronico>eduardo.gus@gmail.com</CorreoElectronico>
    <NumeroTelefono>3454 3545</NumeroTelefono>
    <Especialidades>Pixel Art, animación</Especialidades>
    <NotasAdicionales>Disponible para proyectos urgentes</NotasAdicionales>
  </Artista>
  ...
</Artistas>
```

Figura 2. Archivo XML de artistas

Fuente: elaboración propia

donde:

Artistas: es la raíz del archivo

Artista: es la etiqueta que contiene toda la información del artista

id: es el atributo que indica el ID único de cada artista

pwd: es el atributo que contiene la contraseña de la cuenta del artista

NombreCompleto: es la etiqueta que contiene el nombre completo del artista.

CorreoElectronico: es la etiqueta que contiene el e-mail del artista.

NumeroTelefono: es la etiqueta que contiene el número telefónico del artista.

Especialidades: es la etiqueta que contiene información sobre las especialidades de arte del artista.

NotasAdicionales: es la etiqueta que contiene información adicional sobre el artista que no se incluyó en ninguna otra etiqueta.

Con los usuarios cargados en el sistema, el Administrador puede visualizar los reportes en el sistema:

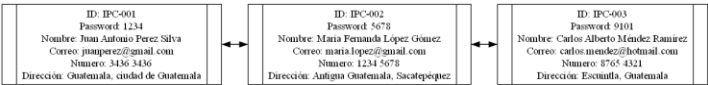


Figura 3. Reporte de lista doble de los solicitantes cargados en el sistema

Fuente: elaboración propia

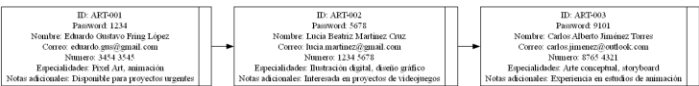


Figura 4. Reporte de lista simple de los artistas cargados en el sistema

Fuente: elaboración propia

Una vez cargados los usuarios, los solicitantes pueden iniciar sesión con su nombre de usuario y contraseña. En su módulo, puede solicitar figuras o ver la galería. La galería de un solicitante únicamente estará disponible cuando algún artista acepte realizar el diseño de dicho solicitante.

En el apartado de Solicitar, el solicitante puede cargar las figuras en XML que posteriormente se volverán una obra pixel-art. Las solicitudes se almacenarán en una pila única para cada solicitante, la cual, al enviar la solicitud, se vaciará y todo su contenido se trasladará a una cola global de solicitudes que será visible para todos los artistas del sistema.

La estructura que debe contener el XML de figuras que subirá el Solicitante debe ser como la siguiente:

```
<figura>
  <nombre id="3">Rana</nombre>
  <diseño>
    <pixel fila="1" col="3">#000000</pixel>
    <pixel fila="1" col="4">#000000</pixel>
    <pixel fila="1" col="5">#000000</pixel>
    <pixel fila="1" col="10">#000000</pixel>
    ...
  </diseño>
</figura>
```

Figura 5. Archivo XML de figura
Fuente: elaboración propia

- donde:
- figura: es la raíz del archivo
- nombre: es la etiqueta que contiene el nombre de la figura
- id: es el atributo que contiene el ID de la figura
- diseño: es la etiqueta que contiene la descripción de todos los pixeles que contiene la figura.
- pixel: es la etiqueta que contiene el color hexadecimal correspondiente a ese pixel.
- fila: atributo que indica la fila de la matriz dispersa en donde se colocará el pixel.
- col: atributo que indica la columna de la matriz dispersa en donde se colocará el pixel.

En el apartado Solicitar, el solicitante también puede ver la pila de imágenes que ha cargado:

ID de solicitante: IPC-001 ID de imagen: 5 Nombre de la figura: Mario Ruta de la imagen: C:/Users/sebas/Documents/Vacaciones de diciembre 2024/Archivos prueba/mario.xml
ID de solicitante: IPC-001 ID de imagen: 2 Nombre de la figura: Capibara Ruta de la imagen: C:/Users/sebas/Documents/Vacaciones de diciembre 2024/Archivos prueba/capibara.xml
ID de solicitante: IPC-001 ID de imagen: 3 Nombre de la figura: Rana Ruta de la imagen: C:/Users/sebas/Documents/Vacaciones de diciembre 2024/Archivos prueba/rana.xml

Pila

Figura 6. Reporte de pila de solicitudes del solicitante IPC-001
Fuente: elaboración propia

Posteriormente, el solicitante puede presionar el botón “Solicitar” lo que vaciará la pila y trasladará las solicitudes a la cola, la cual es visible para todos los artistas del sistema.

Cuando un artista inicie sesión en el sistema, podrá ver la primera solicitud en la cola:



Figura 7. Cola general de solicitudes
Fuente: elaboración propia

El artista puede ver toda la cola de solicitudes presionando el botón de Ver Cola:

ID de solicitante: IPC-001 ID de imagen: 3 Nombre de la figura: Rana Ruta de la imagen: C:/Users/sebas/Documents/Vacaciones de diciembre 2024/Archivos prueba/rana.xml	ID de solicitante: IPC-001 ID de imagen: 2 Nombre de la figura: Capibara Ruta de la imagen: C:/Users/sebas/Documents/Vacaciones de diciembre 2024/Archivos prueba/capibara.xml
---	---

Cola

Figura 8. Reporte de la cola de solicitudes
Fuente: elaboración propia

Al presionar Aceptar, el artista comenzará a crear la matriz dispersa con los pixeles descritos en el XML. Una vez terminada, la imagen se enviará a la lista circular doblemente enlazada del solicitante.

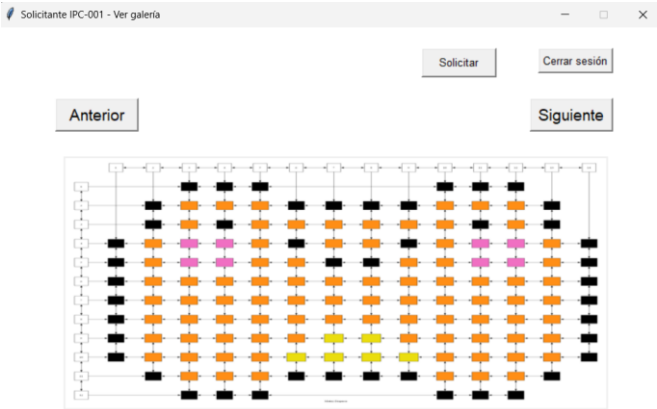


Figura 9. Galería del solicitante
Fuente: elaboración propia

Conclusiones

El programa cumple con la funcionalidad de cargar archivos XML que contienen diferente información y son necesarios para el funcionamiento del sistema.

Además, el sistema cuenta con diferentes roles según el ID de cada usuario, por lo que se garantiza que cada usuario desempeñe únicamente el rol que le corresponde.

También se evidenció a través de los reportes que el sistema está construido con base en varios TDA como listas simples, listas dobles, listas circulares simples y dobles, así como pilas y colas que son los que almacenan los datos necesarios para el correcto funcionamiento de la plataforma.

Por último, se utilizó el modelo de una matriz dispersa para crear las imágenes de arte en pixeles.

Referencias bibliográficas

USAC, T.d.(17 de Diciembre de 2024). Proyecto 1, UEDi. Obtenido de <https://uedi.ingenieria.usac.edu.gt/campus/mod/assign/view.php?id=876300>

Anexos

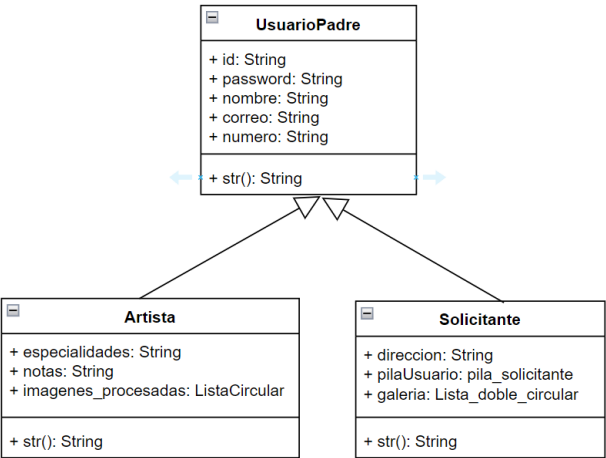


Figura 10. Diagrama de clases de los usuarios
Fuente: elaboración propia

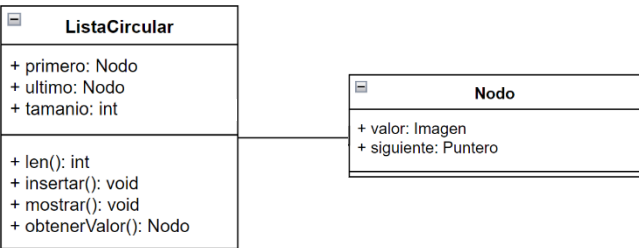


Figura 11. Diagrama de clases de la lista circular de cada artista
Fuente: elaboración propia

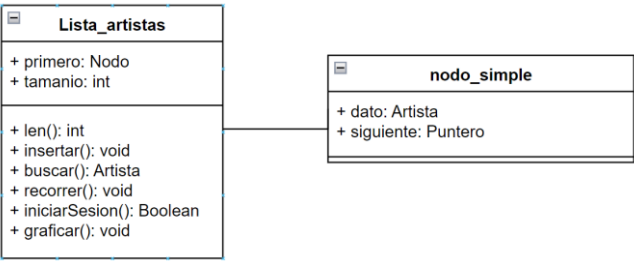


Figura 12. Diagrama de clases de la lista simple que contiene artistas

Fuente: elaboración propia

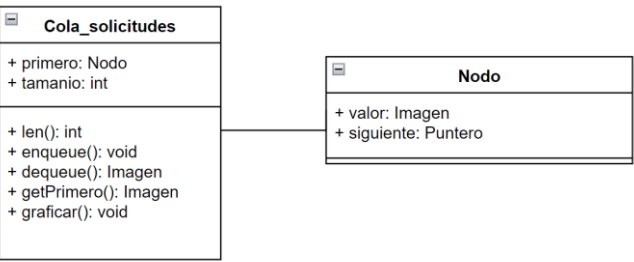


Figura 13. Diagrama de clases de la cola de solicitudes

Fuente: elaboración propia

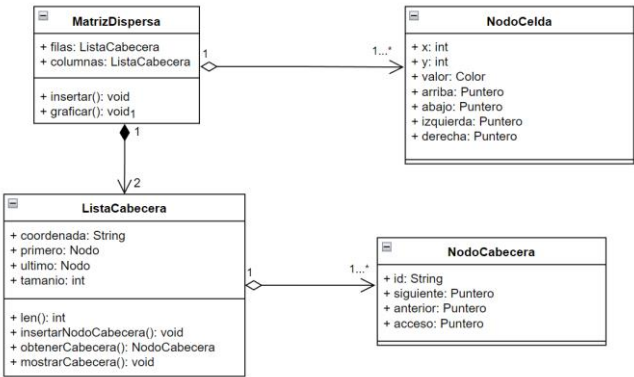


Figura 14. Diagrama de clases de la matriz dispersa

Fuente: elaboración propia

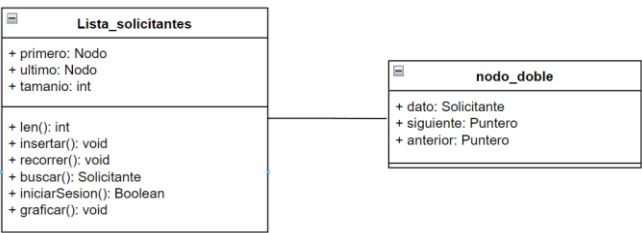


Figura 15. Diagrama de clases de la lista doble que contiene solicitantes

Fuente: elaboración propia

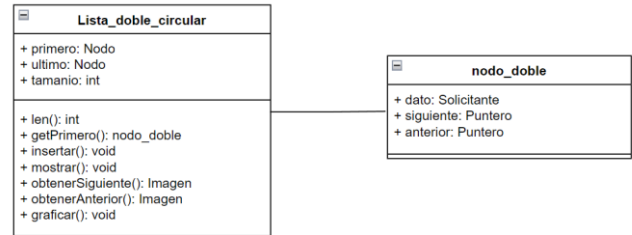


Figura 16. Diagrama de clases de la lista circular doble única de cada solicitante

Fuente: elaboración propia

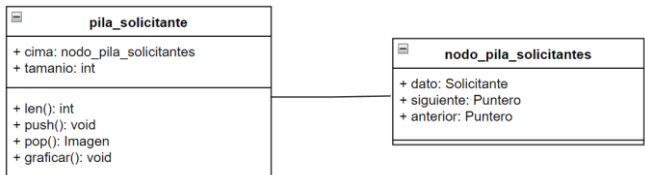


Figura 17. Diagrama de clases de la pila de solicitudes única de cada usuario

Fuente: elaboración propia