



Universidad  
de Concepción

# PIZARRA MÚLTIPLE PARA DIAGRAMAS UML

Proyecto tema 4: "pizarra para hacer clases web"

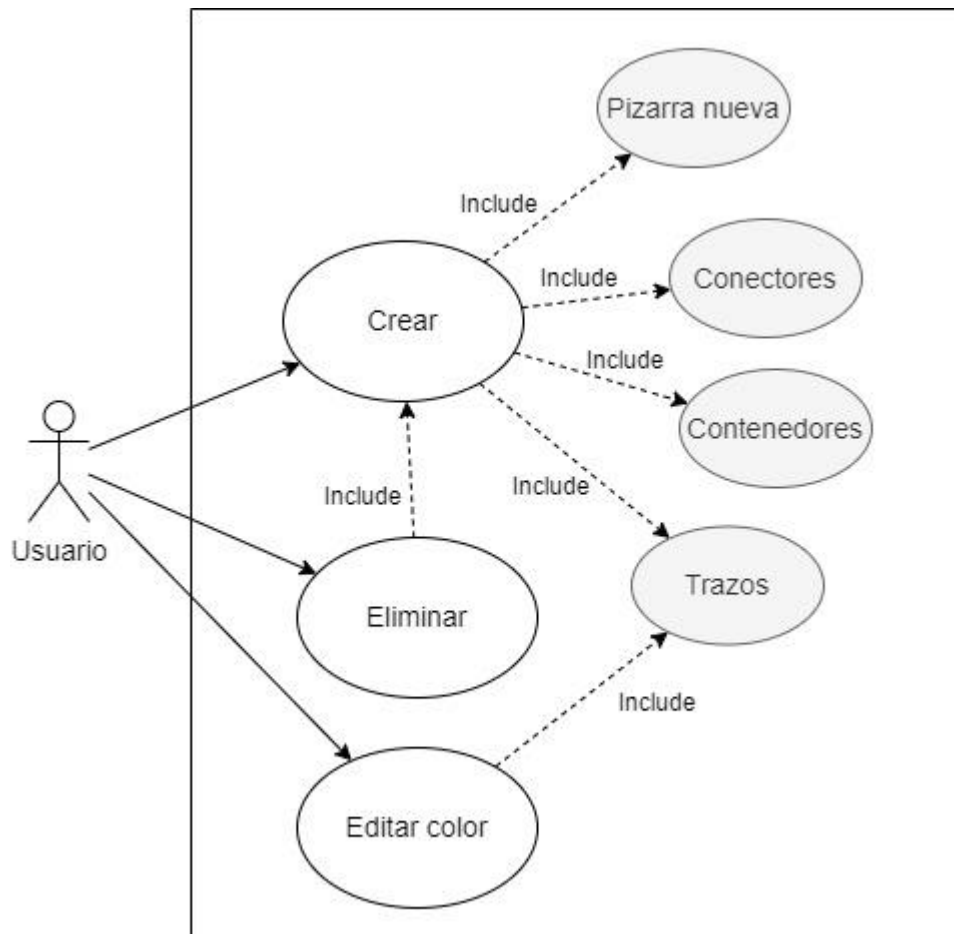
Ingeniería Civil Informática

Alumna:	Catalina Bravo Hernández
Profesor:	Geoffrey Hecht
Asignatura:	Programación II
Fecha:	21/12/2022

## Enunciado general

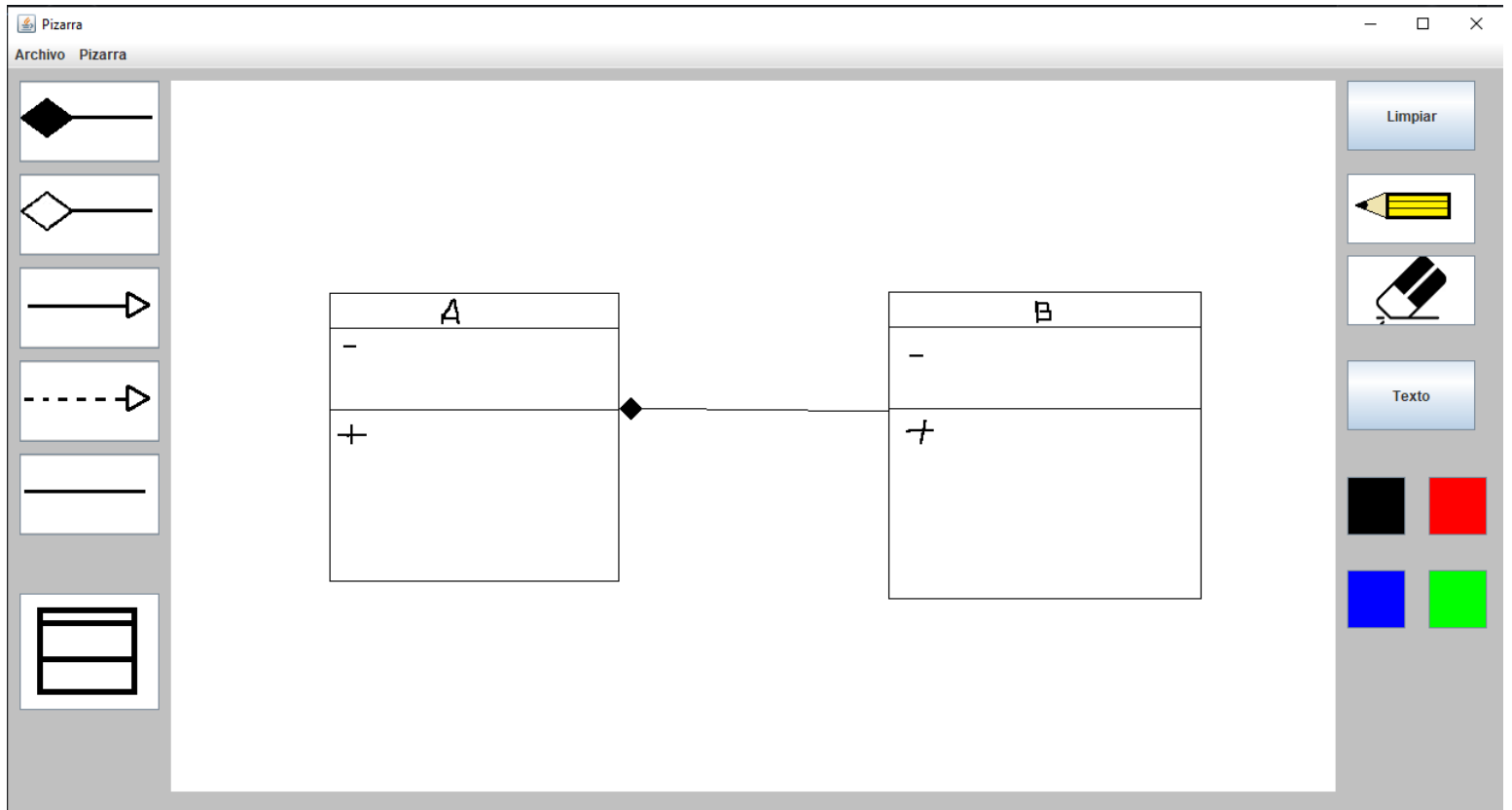
Se debe crear una pizarra múltiple que permita dibujar trazos, rectángulos, y conectores UML con mouseDrag. La pizarra debe modos de edición tales como borrar, y crear las diferentes formas de UML. Para eliminar se busca el elemento que contenga los pixeles encerrados por un rectángulo fantasma que queda entre mousePress y mouseRelease. La pizarra múltiple consiste en tener en el panel central una de ellas a la vez, pudiendo cambiar de una a otra sin perder lo editado. Se debe poder grabar en archivo la pizarra múltiple y cargarla de este. se debe poder borrar la pizarra completa. La pizarra múltiple se debe grabar en el archivo automáticamente al cambiar de una pizarra a otra. Se debe poder elegir el color de las líneas mediante botones GUI.

## Diagrama de casos de uso





## Vista de interfaz



## Decisiones tomadas y problemas encontrados

- La función de pizarra múltiple se programó dentro de arreglos, utilizando métodos que usan funciones de arreglos para agregar, borrar y navegar entre pizarras. Sin embargo, la primera pizarra no se puede eliminar.
- El lápiz desaparecía al colocar otros objetos. Se solucionó utilizando el método `mousePressed` en la función `mouseAdapter`.
- Problema con la clase `Realizacion`, al querer utilizarla no compilaba correctamente y las demás clases de UML comenzaban a funcionar de manera errática. Para poder implementar la función de `BasicStroke` en el panel de la pizarra, se utilizó dos `For`, uno pasándole el parámetro de `BasicStroke` y el otro con la función de `DrawLine`.
- Se agregó métodos a la clase lápiz que cambian el color del lápiz
- Se realizó tests, se ordenó el código con diferentes packages, y se escribieron comentarios en formato `JavaDoc`
- Se creó una clase `Goma`, la cual consiste en un lápiz que pinta blanco, a modo de borrador.
- El botón `Limpiar` utiliza la función `RemoveAll` para quitar todos los objetos que se agregan al panel durante tiempo de ejecución.

En conclusión, no se logró programar el borrado de selección por rectángulo fantasma, sin embargo, se agregó la función de limpiar la pizarra completa, y un borrador tipo lápiz. La funcionalidad de cargar/guardar archivos no se realizó.

Finalmente, se pretendió agregar una entrada de texto dentro del contenedor UML, pero en reemplazo, se programó un lápiz el cual puede dibujar en distintos colores, mencionado anteriormente.