

ANTES DE LEER EL ENUNCIADO

- 1) Hacer file in del archivo adjunto en el mail llamado **2025-1C-Parcial-2.st**
- 2) Grabar la imagen
- 3) Ir al formulario <https://tinyurl.com/inge1-2p-1c-ini-2025>
- 4) Completarlo y llamar al docente para obtener el código de entrega inicial

IMPORTANTE: El file out de la entrega debe hacerse desde la misma imagen que se hizo el file out para completar el paso 4, de lo contrario el parcial no será aprobado. Si por algún motivo no puede usar la misma imagen, deberá llamar al docente.

Parte Práctica: Pintor

Hay que desarrollar un sistema para “pintar pantallas” de aplicaciones, por eso el sistema se denomina **Pintor**. Las pantallas a diseñar pueden ser de distinta tecnología, ventanas de aplicaciones desktop (también conocidas como ‘windows’), páginas web o ventanas de aplicaciones mobile, por lo que en esta etapa se pide desarrollar solamente el modelo de la herramienta, o sea la parte no visual que implementa la funcionalidad más allá de la presentación gráfica y que luego será usado por las distintas tecnologías de presentación.

El objetivo del sistema es que el pintor pinte y manipule pinturas que pondrá sobre una pared. El pintor debe poder pintar sobre distintas paredes (no simultáneamente) y la extensión de cada pared puede variar en sus ejes x e y. La extensión debe ser siempre positiva. La posición 0@0 es la parte de más arriba a la izquierda donde se pinta en la pared y no pueden haber pinturas fuera de ella.

Respecto de las pinturas, por ahora lo único que nos importa es su posición x@y sobre la pared y su extensión hacia la derecha y abajo en los ejes x e y. Por lo tanto, una pintura en posición 10@5 y extensión 3@2 tiene la esquina derecha inferior en la posición 13@7.

En esta versión se espera que el pintor pueda realizar las siguientes acciones:

- Agregar una pintura a una posición de la pared que se está pintando, y de cierta extensión. La pintura debe entrar por completo en la pared. No puede agregarse la misma pintura más de una vez.
(No es necesario tener un mensaje para borrar pinturas)
- Poder seleccionar una pintura de la pared y poder agregar una pintura a una selección ya realizada. Esto implica que puede haber varias pinturas seleccionadas. No se puede volver a seleccionar una pintura ya seleccionada.
(No es necesario tener un mensaje para de-seleccionar una pintura)
- Poder alinear las pinturas seleccionadas a izquierda o derecha. Esto es válido únicamente cuando hay más de una pintura seleccionada.
 - Para alinear a izquierda se toma la pintura cuya **posición x** sea la menor y se mueven todas las pinturas a ese x.
Por ejemplo, dadas la pinturas seleccionadas { 0@0 extent: 5@5. 5@3 extent: 10@10 } luego de alinear a izquierda deben quedar así: { 0@0 extent: 5@5. 0@3 extent: 10@10 }
(unOrigen extent: unaExtensión crea un rectángulo en la posición unOrigen con extensión unaExtensión)
 - Para alinear a derecha se toma la pintura cuya **posición x + ancho** sea mayor, y se mueven todas las pinturas para que su **posición x +**

ancho sea esta.

Por ejemplo, dadas la pinturas seleccionadas { **0@0 extent: 5@5. 5@3 extent: 10@10** } luego de alinear a izquierda deben quedar así: { **10@0 extent: 5@5. 5@3 extent: 10@10** }

El sistema debe llevar un historial de las acciones realizadas para poder **deshacerlas** y volverlas a **rehacer**, o sea debe poder hacer **undo** y **redo** de las acciones. El historial de acciones a deshacer y rehacer debe quedar siempre consistente y no se debe permitir hacer cosas inválidas. Por ejemplo:

- No se puede deshacer una acción (undo) si no se hizo ninguna acción
- No se puede rehacer (redo) una acción si no se deshizo (undo) una antes
- Cuando se realiza una acción se deben descartar las acciones por rehacer porque sino se podría generar un árbol de acciones a rehacer y solo queremos mantener una lista.
- etc

Tips:

- Analizar el protocolo de las clases Point y Rectangle antes de encarar cualquier diseño!
- Las pinturas pueden ser mutables

Trabajo a realizar

Implementar la funcionalidad pedida en la categoría de clase **2025-1C-Parcial-2** usando TDD y las heurísticas de diseño vistas en la materia.

Entrega:

1. Entregar el fileout de la categoría de clase **2025-1C-Parcial-2** que debe incluir toda la solución (modelo y tests). El archivo de fileout se debe llamar: **2025-1C-Parcial-2.st**
2. Entregar también el archivo que se llama **CuisUniversity-nnnn.user.changes**
3. Probar que el archivo generado en 1) se cargue correctamente en una imagen “limpia” (o sea, sin la solución que crearon. Usen otra instalación de CuisUniversity/imagen si es necesario) y que todo funcione correctamente. Esto es fundamental para que no haya problemas de que falten clases/métodos/objetos en la entrega.
4. Deben entregar usando el siguiente form: <https://tinyurl.com/inge1-2p-1c-2025>
5. De forma alternativa si no pudiste entregar con el form, realizar la entrega enviando mail a: **entregas@isw2.com.ar** con el **Subject: LU nnn-aa - Solución 2do parcial 1c2025.**

En caso de rebotar el envío, reintentar comprimiendo los adjuntos.

6. **RECOMENDACIÓN IMPORTANTE: Salvar la imagen de manera frecuente o con el autosave**
7. Se asume que a esta altura de la cursada saben trabajar con la imagen, recuperarla, recuperar código fuente, revertir cambios y demás incidencias que pudieran ocurrir durante el examen.

Revisen bien los puntos de arriba. Cualquier error en los nombres o formato podrían ser penalizados en la nota.

IMPORTANTE: No retirarse sin tener el ok de los docentes de haber recibido la resolución por algún medio.

CERRAR EL TRABAJO A LAS 21:50.
LAS ENTREGAS RECIBIDAS DESPUÉS DE LAS 22:00 HRS NO SERÁN TENIDAS EN
CUENTA