

Proyecto Inicial DOPO

Novoa-Sierra

RETROSPECTIVA

1. ¿Cuáles fueron los mini-ciclos definidos? Justifíquenlos.

Definimos varios mini-ciclos para orientar el desarrollo del simulador:

- **Mini-ciclo 1:** Crear la clase Tower y su constructor – para tener la estructura básica de la torre y los atributos principales (width, maxHeight, visible, cups, looseLids).
- **Mini-ciclo 2:** Manejo de tazas (pushCup, popCup, removeCup) – para garantizar que podíamos agregar, quitar o eliminar tazas de la torre correctamente.
- **Mini-ciclo 3:** Manejo de tapas (pushLid, popLid, removeLid) – para manejar las tapas asociadas a las tazas o sueltas.
- **Mini-ciclo 4:** Reorganización de la torre (orderTower, reverseTower) – para implementar las operaciones de ordenamiento requeridas por el simulador.
- **Mini-ciclo 5:** Consultas y visibilidad (height, stackingItems, makeVisible, makeInvisible) – para poder medir la altura, obtener información y controlar la visualización de la torre.
- **Mini-ciclo 6:** Finalización y limpieza (exit, ok) – para vaciar la torre y probar métodos auxiliares.

Los justificamos porque cada mini-ciclo corresponde a un conjunto de funcionalidades específicas, lo que permitió enfocarnos en partes concretas antes de integrar todo.

2. ¿Cuál es el estado actual del proyecto en términos de mini-ciclos? ¿Por qué?

Actualmente, todos los mini-ciclos definidos están completados. Se implementaron las funcionalidades básicas del simulador: manejo de tazas y tapas, reorganización de la torre, consultas y visibilidad. Los métodos privados que apoyan la lógica interna, como findCup y redraw, también están implementados. El proyecto está completo porque después de la entrega inicial corregimos errores y mejoramos el proyecto, afinando la visualización, el manejo de las listas y la consistencia de las operaciones sobre la torre. Sin embargo, no se resolvió el problema de la maratón de programación (calcular la altura favorita exacta usando los n cups), porque en esta primera entrega el enfoque era construir el simulador, no resolver el algoritmo de la competencia.

3. ¿Cuál fue el tiempo total invertido por cada uno de ustedes? (Horas/Hombre)
Novoa y Sierra invertimos 36 horas, porque realizamos todas las actividades del proyecto en conjunto.

4. ¿Cuál consideran fue el mayor logro? ¿Por qué?
El mayor logro fue lograr un simulador funcional y visual, donde cada taza y tapa se dibuja correctamente en el Canvas y se pueden agregar, eliminar, reorganizar y consultar sus atributos. Esto es relevante porque permite probar visualmente la lógica de apilamiento sin depender de la resolución del problema de la maratón, demostrando que la arquitectura del proyecto es sólida y extensible.

5. ¿Cuál consideran que fue el mayor problema técnico? ¿Qué hicieron para resolverlo?
El mayor problema técnico fue manejar correctamente la posición de las tazas y tapas en el Canvas, especialmente cuando las tapas estaban sueltas o sobre las tazas. Para resolverlo, implementamos el método `redraw()` dentro de Tower, que recorre las listas de Cups y Lids y calcula la posición vertical acumulada de cada elemento. También definimos constantes de conversión de cm a pixeles para que las alturas se dibujaran proporcionalmente.

6. ¿Qué hicieron bien como equipo? ¿Qué se comprometen a hacer para mejorar los resultados?

Bien: Todo lo hicimos juntos, trabajando en equipo en cada mini-ciclo. Siempre nos dábamos comentarios para mejorar el código y revisábamos lo que hacía el otro, así que la comunicación fue constante y todos los cambios se discutían antes de implementarlos.

Compromiso: Analizar mejor el problema desde el principio, porque al inicio no lo hicimos tan bien y eso nos llevó a hacer un diagrama un poco equivocado. También queremos documentar mejor los métodos privados y probar más la interacción entre tazas y tapas para que no haya errores en la visualización.

7. Considerando las prácticas XP incluidas en los laboratorios. ¿Cuál fue la más útil? ¿Por qué?

Para nosotros, la práctica que más nos ayudó fue la integración continua y la revisión de pares. Cada cambio que hacíamos lo probábamos de inmediato en el simulador, y siempre nos revisábamos el código entre nosotros antes de integrarlo.

8. ¿Qué referencias usaron? ¿Cuál fue la más útil? Incluyan citas con estándares adecuados.

Para desarrollar el proyecto usamos varias referencias:

- **Inteligencia Artificial / ChatGPT**, que nos ayudó a aclarar dudas sobre la lógica de los métodos y la organización del código.
- **Documentación oficial de Java**, especialmente para manejar interfaces gráficas y elementos visuales como JOptionPane.

- **Video de YouTube:** <https://www.youtube.com/watch?v=WSyFvjUQHL4>, que nos ayudó a **entender matemáticamente el ejercicio** del Stacking Cups y cómo plantear la lógica de la altura de la torre.

De todas estas referencias, las más útiles fueron **la documentación de Java y el video de YouTube**, porque nos permitieron entender tanto la parte de implementación visual como la parte matemática del problema.