

**DESARROLLO ORIENTADO A OBJETOS**

**Construcción. Clases y objetos.**

**Retrospectiva proyecto inicial, ciclo 1, 2, 3 y 4**

**2025-2**

**Mateo Alejandro Sánchez Buitrago**

**Camilo Alejandro Aguirre**

RETROSPECTIVA CICLO 1

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

Se definieron tres mini-ciclos, fueron los siguientes:

1.a. El diseño inicial: se hizo la construcción del diagrama de clases y se definió responsabilidades en cada clase.

1.b. La implementación básica: Se hizo la creación de las clases principales, que eran: SilkRoadSimulator, SilkRoadModel, SilkRoadView, Robot, Store. Además, se hizo la validación de compilación.

1.c. La integración con shapes y visualización: Se incorporo el paquete shapes para la representación gráfica. Se ajustaron los métodos: eraseAll, getStores, getRobots. Y se hizo la verificación en BlueJ.

Estos mini-ciclos fueron definidos para avanzar del diseño a lo concreto que es la implementación y visualización, asegurando que haya correcciones rápidas en cada etapa.

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

El proyecto se encuentra en el tercer mini-ciclo concluido y logramos que: El código está completo y compile sin errores. Se integró con shapes para cumplir con el requisito de la visualización. Se realizaron los diagramas UML (clases y secuencia). Se alcanzó el objetivo del ciclo final.

3. ¿Cuál fue el tiempo total invertido por cada uno de ustedes? (Horas/Hombre)

Integrante Camilo Aguirre: 10 horas.

Integrante Mateo Sanchez: 8 horas.

Total: 18 horas en total.

4. ¿Cuál consideran fue el mayor logro? ¿Por qué?

El mayor de los logros fue lograr la integración correcta del paquete shapes en BlueJ. Permitió cumplir con el requerimiento gráfico, mostró la dinámica y tiendas en el modelo. Este paso aseguró que el sistema no fuera solamente lógico, sino también visual.

5. ¿Cuál consideran que fue el mayor problema técnico? ¿Qué hicieron para resolverlo?

El mayor problema fue no lograr la compatibilidad con métodos que son inexistentes en shapes como: eraseAll, getStores, getRobots.

Para solucionarlo se editaron los códigos de SilkRoadView y SilkRoadModel. Se justaron los métodos a la versión del paquete shapes. Esto se resolvió revisando las clases disponibles, las cuales eran: Circle, Rectangle, Canvas y logrando adaptar la lógica de dibujo.

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

Lo que hicimos bien: Mantener comunicación constante, nos conocemos de hace buen tiempo entonces además de una comunicación constante se logra tener una buena comunicación. Tambien se dividieron algunas partes y se verificaron avances en cada paso.

Compromiso: Tratar de usar el tiempo desde que habilitan y nos informan de la tarea. Asi no tener presura a ultimo momento. Además, planificar una estrategia al inicio para llevar un orden lógico.

La práctica más útil fue el diseño incremental en los mini-ciclos. Esto permitió ir corrigiendo problemas a tiempo sin afectar todo el proyecto despues de un gran avance. Aplicando esto los errores de integración con shapes se resolvieron sin rehacer por completo el trabajo o afectar algunas partes por corregir otras.

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

Kolling, M., & Barnes, D. (2016). Objects First with Java: A Practical Introduction using BlueJ. Pearson.

Documentación de BlueJ. (2025). Disponible en: [https://bluej.org](https://bluej.org)

<https://www.youtube.com/watch?v=t8kR7urUTu>

<https://www.youtube.com/watch?v=KJd1Un8SEnk>

<https://www.youtube.com/watch?v=Z0yLerU0g-Q>

<https://www.youtube.com/watch?v=-jZ8JkC-Pwc>

<https://www.youtube.com/watch?v=t8kR7urUTuA&t=844s>

Funcionalidades pendientes. Terminamos el proyecto con todo lo básico funcionando y cumpliendo lo que se pidió en cuanto al diseño. Sin embargo, quedaron pendientes algunas mejoras para más adelante:

* Espiral cuadrada: Por ahora, robots y tiendas se muestran en línea recta, pero la idea original era ubicarlos en forma de espiral. Eso requiere cálculos más complejos, así que lo dejamos para después.
* Barra de ganancias: Las ganancias ya se calculan, pero no logramos mostrar una barra gráfica para que representarlas.
* Mensajes de error: Si algo no se puede hacer (como usar un robot inexistente), no dice nada. Nuestra idea es agregar mensajes que expliquen lo que ocurrió.

Nos enfocarnos en que todo lo esencial funcionara bien, y dejamos estas mejoras para futuras versiones.

RETROSPECTIVA CICLO 2

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

Definimos tres mini-ciclos principales:

* Diseño: actualizamos los diagramas de clases y de secuencia en Astah, para incluir los nuevos métodos (moveRobotToBestStore y createFromMarathonInput).
* Implementación: codificación de las nuevas funcionalidades en SilkRoadModel y SilkRoadSimulator, manteniendo la estructura MVC y estándares de documentación.
* Pruebas: construcción de casos en SilkRoadC2Test y SilkRoadCC2Test para validar los escenarios positivos, negativos y especiales.

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

El proyecto está completo en todo el ciclo 2. Ya se implementaron y probaron las funcionalidades pedidas: entrada estilo maratón y selección automática de la tienda más rentable.  
Las pruebas unitarias pasaron en verde, lo que asegura que el sistema cumple los requisitos definidos.

3. ¿Cuál fue el tiempo total invertido por cada uno de ustedes? (Horas/Hombre)

Integrante Camilo Aguirre: 12 horas.

Integrante Mateo Sanchez: 9 horas.

Total: 21 horas en total.

4. ¿Cuál consideran fue el mayor logro? ¿Por qué?

El mayor logro fue lograr el estilo maratón, ya que con esto se pudo crear múltiples tiendas con cadenas de texto, simulando lo de la maratón.

Esto fue un gran reto, pero de gran ayuda una vez se logró resolver.

5. ¿Cuál consideran que fue el mayor problema técnico? ¿Qué hicieron para resolverlo?

El mayor problema técnico fue el manejo de los empates al calcular la tienda más rentable. Al inicio el robot no elegía bien la tienda correcta en igualdad de ganancias.  
La solución fue implementar un criterio de desempate usando el id más bajo de la tienda, garantizando un ganador en el resultado.

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

Lo que hicimos bien: Mantener comunicación constante, nos conocemos de hace buen tiempo entonces además de una comunicación constante se logra tener una buena comunicación. También se dividieron algunas partes y se verificaron avances en cada paso.

Compromiso: Tratar de usar el tiempo desde que habilitan y nos informan de la tarea. Así no tener presura a último momento. Además, planificar una estrategia al inicio para llevar un orden lógico.

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

La práctica más útil fue Pruebas unitarias primero (Test-Driven Development).  
Nos permitió descubrir errores en el parsing de las entradas estilo maratón y en la lógica de rentabilidad antes de que el sistema creciera demasiado.  
Esto redujo significativamente el corregir errores.

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

<https://www.youtube.com/watch?v=svJouML-7Ws>

<https://www.youtube.com/watch?v=tHNWIWxRDDA>

RETROSPECTIVA CICLO 3

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

• **Diseño:** actualizamos los diagramas/contratos del simulador para incorporar los nuevos puntos del ciclo 3, reflejando el constructor SilkRoad, el cálculo en SilkRoadContest.solve y el refactor interno de redraw() en helpers, manteniendo las interfaces existentes.  
• **Implementación:** codificación de las nuevas funcionalidades manteniendo el diseño previo del proyecto. Simulación visual con control de velocidad en simulate y refactor de dibujo sin cambiar la lógica.  
• **Pruebas:** construcción de casos unitarios silenciosos en SilkRoadContestTest (3 positivas y 3 negativas).

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

El proyecto está completo en todo el ciclo 3. Ya se implementaron y probaron las funcionalidades pedidas: entrada estilo maratón y selección automática de la tienda más rentable.  
Las pruebas unitarias pasaron en verde, lo que asegura que el sistema cumple los requisitos definidos

1. ¿Cuál fue el tiempo total invertido por cada uno de ustedes? (Horas/Hombre)

Integrante Camilo Aguirre: 12 horas.

Integrante Mateo Sanchez: 12 horas.

Total: 24 horas en total.

1. ¿Cuál consideran fue el mayor logro? ¿Por qué?

El mayor reto fue tener a solve+simulate integrados con el simulador existente respetando el diseño previo, más una interfaz y pruebas silenciosas.

1. ¿Cuál consideran que fue el mayor problema técnico? ¿Qué hicieron para resolverlo?

El mayor problema técnico fue sincronizar las pausas/animaciones de la simulación: Canvas.getCanvas() fallaba según el entorno. La solución fue reemplazarlo por un helper pause(ms) con Thread.sleep(...) y refactorizar redraw() en helpers, dejando la animación estable y el código más claro.

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

Lo que hicimos bien: Mantener comunicación constante, nos conocemos de hace buen tiempo entonces además de una comunicación constante se logra tener una buena comunicación. También se dividieron algunas partes y se verificaron avances en cada paso.

Compromiso: Tratar de usar el tiempo desde que habilitan y nos informan de la tarea. Así no tener presura a último momento. Además, planificar una estrategia al inicio para llevar un orden lógico.

1. Considerando las prácticas XP incluidas en los laboratorios. ¿cuál fue la más útil? ¿por qué?

Refactor continuo y pruebas automatizadas. El refactor de redraw() redujo complejidad y las pruebas silenciosas evitaron regresiones.

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

<https://www.youtube.com/watch?v=74sClDEYSQ4&pp=ygUWcHJ1ZWJhcyB1bml0YXJpYXMgamF2YQ%3D%3D>

<https://www.youtube.com/watch?v=74sClDEYSQ4&pp=ygUWcHJ1ZWJhcyB1bml0YXJpYXMgamF2YQ%3D%3D>

<https://www.youtube.com/watch?v=74sClDEYSQ4&pp=ygUWcHJ1ZWJhcyB1bml0YXJpYXMgamF2YQ%3D%3D>

RETROSPECTIVO CICLO 4

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

Diseño: Se actualizó el diseño del simulador para incorporar las nuevas funcionalidades del ciclo 3: creación de tiendas de tipo normal, autonomous y fighter, y robots de tipo normal, neverback y tender. Se mantuvo el esquema original de las clases (Store, Robot, SilkRoad) respetando las interfaces previas y el principio de cohesión.}

Pruebas: Se realizaron pruebas directas en BlueJ mediante ejecución controlada de los métodos placeStore, placeRobot y moveRobots. También se verificó la compilación independiente de cada clase (Store, Robot, SilkRoad) para asegurar que todas las referencias entre ellas fueran consistentes.

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

El proyecto se encuentra completo y funcional en todos los mini-ciclos definidos.  
Todas las funcionalidades pedidas fueron implementadas y probadas:

* Las tiendas y robots admiten distintos tipos de comportamiento.
* El simulador ejecuta correctamente el movimiento y las interacciones entre ambos.
* No hay errores de compilación ni de ejecución.

Esto indica que el sistema cumple con los objetivos del ciclo 3 y está listo para presentación o validación.

3. ¿Cuál fue el tiempo total invertido por cada uno de ustedes? (Horas/Hombre)

* Integrante Camilo Aguirre: 15 horas.
* Integrante Mateo Sánchez: 15 horas.

Total: 30 horas hombre.

4. ¿Cuál consideran fue el mayor logro? ¿Por qué?

El mayor logro fue integrar nuevos tipos de comportamiento (robots y tiendas) dentro de una arquitectura ya existente, sin romper su estructura ni crear nuevas clases innecesarias.  
Además, se logró eliminar errores persistentes de compilación y lograr una ejecución limpia en BlueJ, manteniendo la funcionalidad visual del simulador.

5. ¿Cuál consideran que fue el mayor problema técnico? ¿Qué hicieron para resolverlo?

El mayor problema técnico fue el error “cannot find symbol” que impedía compilar correctamente SilkRoad.java.  
Este error se debía a métodos duplicados y versiones viejas de clases compiladas (.class) dentro de BlueJ.

Solución aplicada:

* Limpieza completa del proyecto.
* Recompilación en orden.
* Eliminación de duplicados (withdraw, updateAppearance, atributo type).
* Incorporación correcta de los métodos setType y getType en Robot y Store.

Con esto, el simulador compiló correctamente y pudo ejecutar sin errores.

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

Lo que hicimos bien: Mantener comunicación constante, nos conocemos de hace buen tiempo entonces además de una comunicación constante se logra tener una buena comunicación. También se dividieron algunas partes y se verificaron avances en cada paso.

Compromiso: Tratar de usar el tiempo desde que habilitan y nos informan de la tarea. Así no tener presura a último momento. Además, planificar una estrategia al inicio para llevar un orden lógico.

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

La práctica más útil fue Refactorización (Refactor).  
A lo largo del proceso, se aplicó refactorización constante para mejorar la legibilidad, reducir duplicación de código y mantener la coherencia entre las clases.  
En particular, el refactor de Store.java permitió eliminar métodos repetidos (withdraw, updateAppearance) y variables duplicadas (type), asegurando una estructura más limpia y mantenible.  
Además, se reorganizó el código de SilkRoad.java para mejorar la claridad en los métodos moveRobots() y placeStore(), sin alterar la funcionalidad existente.

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

<https://www.youtube.com/watch?v=XBOFz76U1F0&pp=ygUfcHJ1ZWJhcyB4cCByZWZhY3Rvcml6YWNpb24gamF2YdIHCQkHCgGHKiGM7w%3D%3D>

<https://www.youtube.com/watch?v=74sClDEYSQ4&pp=ygUWcHJ1ZWJhcyBkZSB1bmlkYWQgamF2YdIHCQkHCgGHKiGM7w%3D%3D>