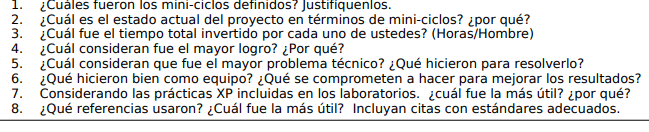
CICLO #1

SILKROAD

MIGUEL SANDOVAL

SANTIAGO GOMEZ

1. RETROSPECTIVA



1. Mini ciclo 1: agregar las nuevas clases al proyecto ya creado en shapes, las cuales fueron silkRoad, Robot y Store ya que es la integración de lo que se quiere lograr en este proyecto nuevo.

Mini ciclo 2: ayudarnos de las clases de shapes para la implementación de figuras básicas y así poder representar las tiendas y los robots.

1. El estado actual es un buen funcionamiento del juego silkRoad con algunos errores técnicos, también contamos con la parte en astah pero no sabemos si este bien estructurado
2. 6 horas: miguel Sandoval, 7 horas: Santiago Gómez
3. El mayor logro fue integrar las nuevas clases junto con el proyecto shapes, creando un sistema funcional y entendible.
4. El mayor problema fue implementar el código y que se pudiera compilar junto con las otras clases ya que todavía hay errores de funcionamiento frente a la clase canvas. Aun no lo resolvemos por completo.
5. Organizar el trabajo en partes, primero miramos lo que necesitábamos implementar junto con los mini ciclos y una buena comunicación al momento de trabajar en el blue j.
6. La de programación en parejas, esta nos ayudo a minimizar el margen de error en el código.
7. Documentación oficial de BlueJ: <https://www.bluej.org>

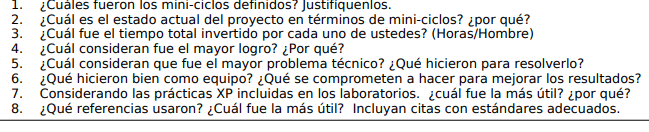
CICLO #2

SILKROAD

MIGUEL SANDOVAL

SANTIAGO GOMEZ

1. RETROSPECTIVA



1. Decidimos dividir el trabajo en 3 partes principales:

**Mini-ciclo 1: Mejorar cómo crear la Ruta de la Seda (Construcción Avanzada (Requisito 10))**

* Hacer que se pudiera crear usando el formato que pide el problema de la maratón
* También permitir crearla pasando arrays con la información de tiendas y robots
* Esto era importante porque facilitaba probar nuestro código con diferentes casos

**Mini-ciclo 2: Hacer los robots más inteligentes (Movimientos Inteligentes (Requisito 11))**

* Los robots ahora pueden decidir por sí solos hacia dónde moverse para ganar más dinero
* Agregamos un método para mover todos los robots de una vez
* Fue la parte más complicada porque teníamos que hacer que eligieran bien

**Mini-ciclo 3: Llevar estadísticas (Sistema de Estadísticas (Requisitos 12 y 13))**

* Contar cuántas veces se vacía cada tienda
* Guardar cuánto dinero gana cada robot en cada movimiento
* Permitir consultar toda esta información cuando se necesite

1. El estado actual del ciclo 2, se completaron los métodos de los tres ciclos, no tiran errores ni fallos al momento de probarlo en C2Test, es decir tienen buena funcionalidad, pero no logramos arreglar el tema visual al ejecutar el juego desde la clase silkroad directamente, tiene algunos errores, pero de igual manera si se realizan los métodos los hace.
2. 12 horas: miguel Sandoval, 12 horas: Santiago Gómez
3. El mayor logro fue integrar los nuevos métodos del ciclo2, que todo funcionara de manera correcta sin errores ni fallos.
4. El mayor problema fue dejar la parte visual funcionando de la mejor manera ya que al crear robot se crean otros sin uno pedirlo, es algo que trataremos de arreglar para el ciclo3, y que tanto la parte visual como al probar los métodos del C2Test funcione de manera correcta.
5. Lograr resolver dudas que el otro compañero tenia y así ir solucionado todo de la mejor manera, también ayudándonos mutuamente para la parte del código y que no tuviera errores ni fallos, y lográramos una buena implementación.
6. La de programación en parejas, esta nos ayudo a minimizar el margen de error en el código.

* Documentación oficial de BlueJ: <https://www.bluej.org>
* Curso de Moodle de DOPO: <https://campusvirtual.escuelaing.edu.co/moodle/course/view.php?id=95&section=4>
* <https://www.ibm.com/docs/es/spss-modeler/saas?topic=language-object-oriented-programming>
* Problema j: <https://drive.google.com/file/d/1jlwi4nlcb0RpT7L4wC8rCcdpl5EZaDgR/view>
* Video Problema j: <https://www.youtube.com/watch?v=mcwVytY5ooU>