Escuela Colombiana de Ingeniería Julio Garavito

Andrés Cardozo

Tulio Riaño Sánchez

Proyecto Inicial

Programación Orientada a Objetos

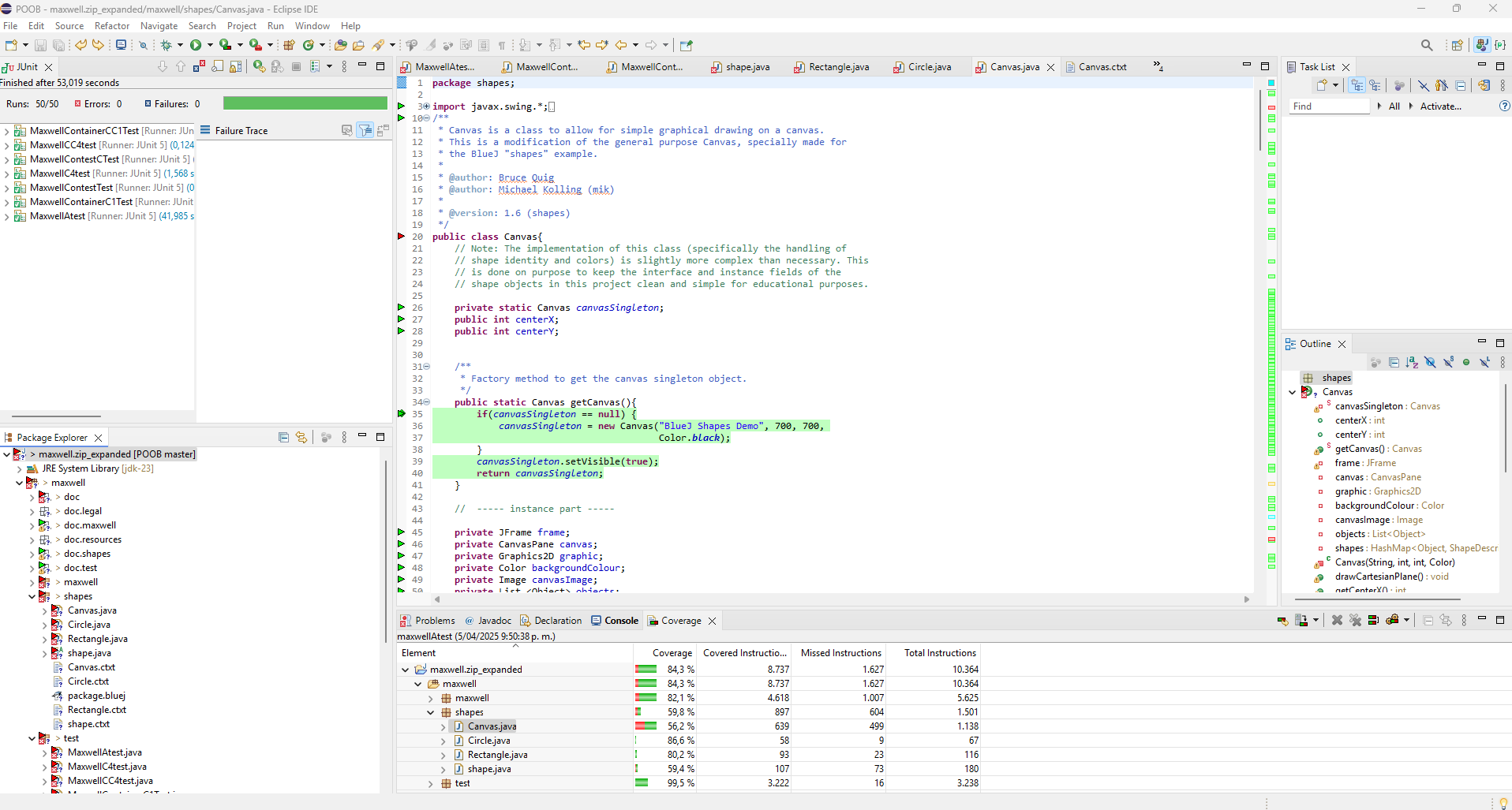
FECHA:

5/04/2025

En la aplicación de Eclipse se organizó todo el proyecto respectivamente mediante sus paquetes.

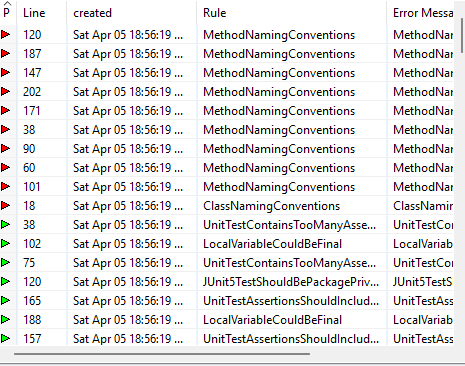
**Informes de análisis dinámico: resultado inicial y resultado final.**

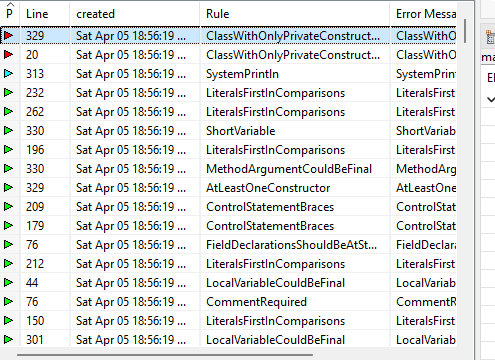
En la imagen se evidencia más del 75% de cubrimiento de código de dominio.



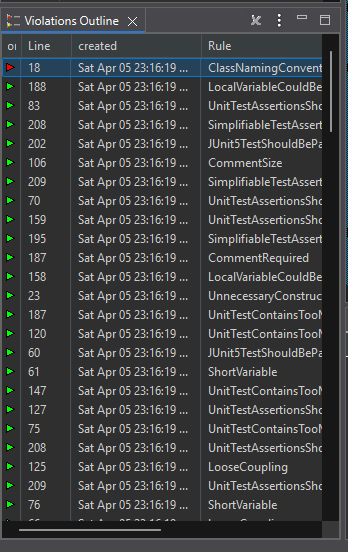
**Informes de análisis estático: resultado inicial y resultado final.**

A partir de la revisión de las posibles reglas de prioridad (alta) que se encuentran en el proyecto, nos encontramos que son por lo general namingConventions al tratar de modificarlas, es decir, solucionar este tipo de problema perdíamos porcentaje en el análisis dinámico por lo que se dificulta la solución a esto.

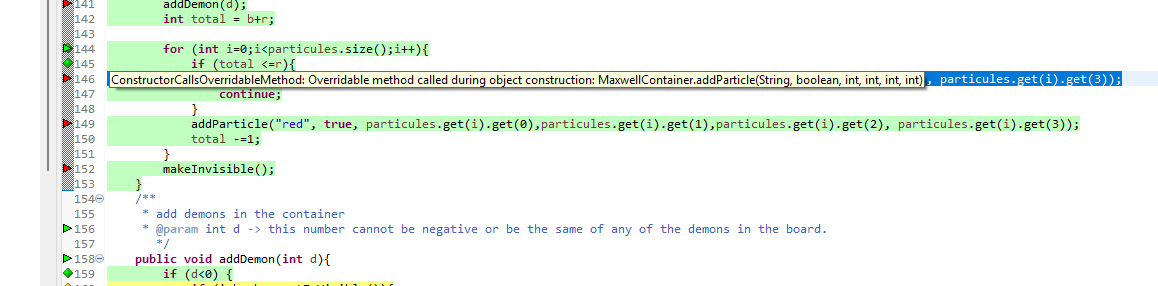




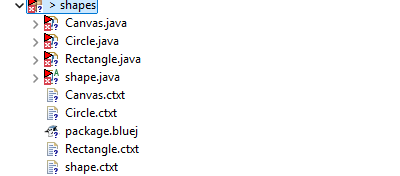
Quitando cosas innecesarias queda:



Se presentan casos como en la clase Canvas que no podemos modificar ya que esta ya venía por defecto en el paquete Shapes, por lo que modificarlo podría llegar a causar distorsión con todo lo desarrollado durante el tercio.



Por otra parte, con posibles malas prácticas como las que se adjunta en la imagen anterior en el que se puede llegar a presentar un flujo incorrecto al no manejar excepciones (caso que no se maneja durante estos primeros ciclos) se nos dificulta la solución. Así como mencione con anterioridad al tratar de solucionar posibles errores en el código se perdía todo tipo de porcentaje relacionado.



En la imagen que acabamos de adjuntar es del paquete de shapes no se encuentra la clase Triangle ya que en el proyecto no se utiliza y cuando se hacía la respectiva revisión del código se nos mostraba un 0% esto debido a que no cubrimos nada relacionado a esta figura en las pruebas.

Las reglas de prioridad alta que se pueden llegar a evidenciar en las diferentes clases de este paquete al tratar de resolver cosas tan sencillas como naming conventions hace que pierda porcentaje todo el proyecto por lo que no se modificaron. Intentando mejorarlo, eliminamos varias cosas que no se usaban como, por ejemplo, la clase triangle, o usar todos los tipos de colores para la creación de partículas, también eliminábamos métodos anteriores como moveUP de shapes ya que nosotros no los usábamos y nos mejoró:



Estos cambios quedan en el archivo comprimido expandido de maxwell.

**Retrospectiva**

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

Para esta parte del proyecto solamente se desarrolló en Eclipse PMD y revisar el estado de las pruebas.

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

Con respecto a los mini-Ciclos planteados anteriormente el estado actual del proyecto es correcto.

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

Andrés Cardozo: 5 horas

Tulio Riaño: 5 horas

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

Manejar la aplicación de Eclipse e identificar cuáles serían las reglas de prioridad alta a modificar. También refactorizar el código para hacer buen uso del polimorfismo, ya que tenemos toda la lógica en chamber, nosotros pensamos que una particula no deberia tener la logica de las dimensiones del chamber mas alla de que hace cada una despues de un bounce. Asi mismo con los holes y demonFace ya que los usamos en chamber y la comprobación de sus comportamientos, por ende, estan en chamber. Por lo que usamos el polimorfismo en las particulas con las nuevas condiciones.

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

El mayor problema técnico fue entender por qué no era posible arreglar ciertos aspectos del código ya que se perdía el respectivo porcentaje al tratar de hacer algo. Para resolver esto, eliminamos una clase que no utilizamos para mejorar las prácticas.

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

De nuevo resaltamos la comunicación y disposición para desarrollar el proyecto, así mismo mantenemos nuestro compromiso por la ayuda mutua que se evidencia en el desempeño.

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

Las prácticas XP incluidas en el proyecto son pair programming y testing.

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

Para el ciclo 4 solo utilizamos las referencias que se encuentran en el archivo:

[codeCoverage](https://www.cs.cornell.edu/courses/JavaAndDS/files/codeCoverage.pdf)

[Java Rules | PMD Source Code Analyzer](https://pmd.github.io/pmd/pmd_rules_java.html)

[PMD](https://pmd.github.io/)