

# Maintenance Dokument

Adam und Tim

10. Dezember 2023

# 1 Implementierung

Wir haben uns den Code genauer angeguckt, um ihn zu verstehen und haben realisiert, dass es an sich ziemlich viele hartkodierte Aspekte gibt. Wir sind an sich keinen großen Bugs oder Fehlern begegnet, die wir verbessern, hätten können, weshalb es nicht leicht war Implementierungen zur Verbesserung zu finden.

Angesichts dessen, dass so viel hartkodiert wurde, dachten wir uns, wir könnten neue Formen erstellen, die Rekursiv aufgerufen werden. So kamen wir auf die Idee, fraktal Formen zu generieren. Wir haben uns gedacht, dass wir dadurch viel mehr Möglichkeiten zur Formgenerierung haben und woraufhin viel mehr Mandalas erstellt werden können. Wir haben nun nicht die alten Formen überarbeitet, da wir nun nicht alles umändern wollen, sondern haben einfach neue Fraktal Formen hinzugefügt.

Die Fraktal formen wären unsere verbesserte Implementierung, sowohl als auch einer unsere neuen. Es wurde weder genau im Dokument beschrieben noch wurde es implementiert, jedoch wurde beim eigentlichen Dokument darum gebeten, das man über 2000 Mandalas erstellen kann, was durch die Fraktal-Formen möglich wäre.

Zudem Haben wir mitsamt der Fraktal Formen, eine neue Form hinzugefügt die es vorher nicht gab. Wir haben eine FraktalTriangle Klasse addiert, die ziemlich viele Möglichkeiten produzieren kann, dadurch das die Dreiecke in dem Mandela sich rotieren. Aufgrund dessen, können wir mit ihnen ganz viele verschiedene Variationen erstellen. Sonst haben wir an sich allen Formen die es schon gab eine Fraktal Variation gegeben.

Außerdem hatten wir noch vor das Requirement zu implementieren, dass der Benutzer die Möglichkeit hat das Mandala einzufärben. Wir haben uns dabei dafür entschieden, dass der Nutzer aus einer Farbpalette auswählen kann und die generierten Formen in zufälligen aber ähnlichen Farben eingefärbt werden. Leider war es uns nicht möglich dies für alle generierten Muster zu implementieren da es bei manchen Layouts erwünscht ist, das sich Formen überlappen. Dieses Problem hätte man lösen können, indem aus Formen die sich überschneiden neue Formen erzeugt und diese separat einfärbt. Jedoch hätte man dafür die von JavaFx bereitgestellten Klassen für Circle, Square etc. verwenden müssen welche diese Funktionalität bereitstellen. Da aber als wir auf dieses Problem aufmerksam geworden sind schon alle Formen von Grund auf implementiert waren und selbst eine intersect Methode zu Zeitaufwändig gewesen wäre haben wir uns entschieden sich überlappende Objekte nicht separat einzufärben.