GENETIC ALGORITHM VISUALIZER

- Le Tran Thang 20204926
- Nguyen Phuc Thanh 20200594
- Le Thanh Thang 20194451
- Hoang Van Thang 20204892

REQUIREMENTS

- Problem: Visualize genetic algorithm to solve traveling salesman problem
- Design: Design an UI for the problem with some functions
- Aims: Make the algorithm more understandable for users

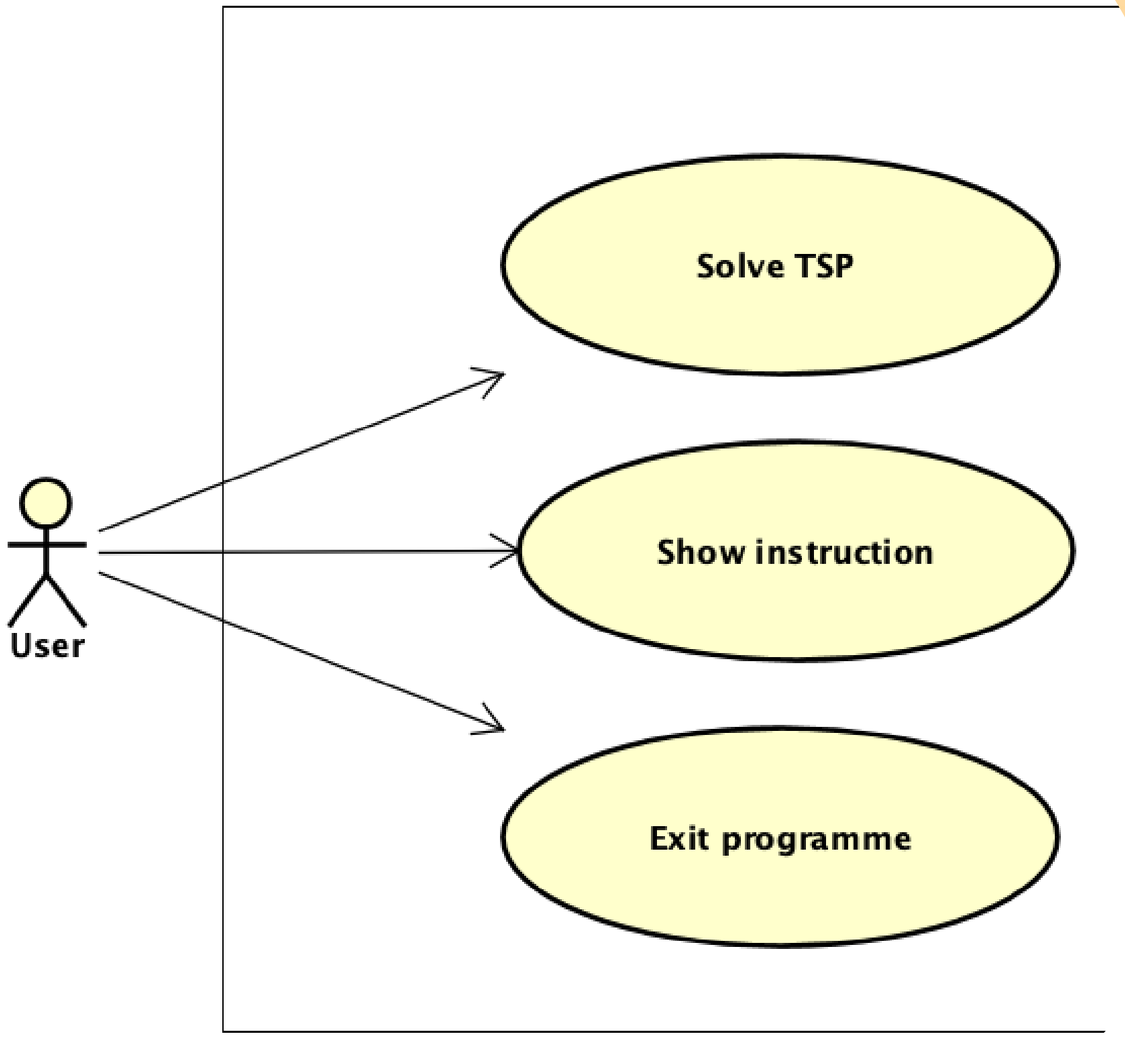
Use case diagram

User can start the visualization

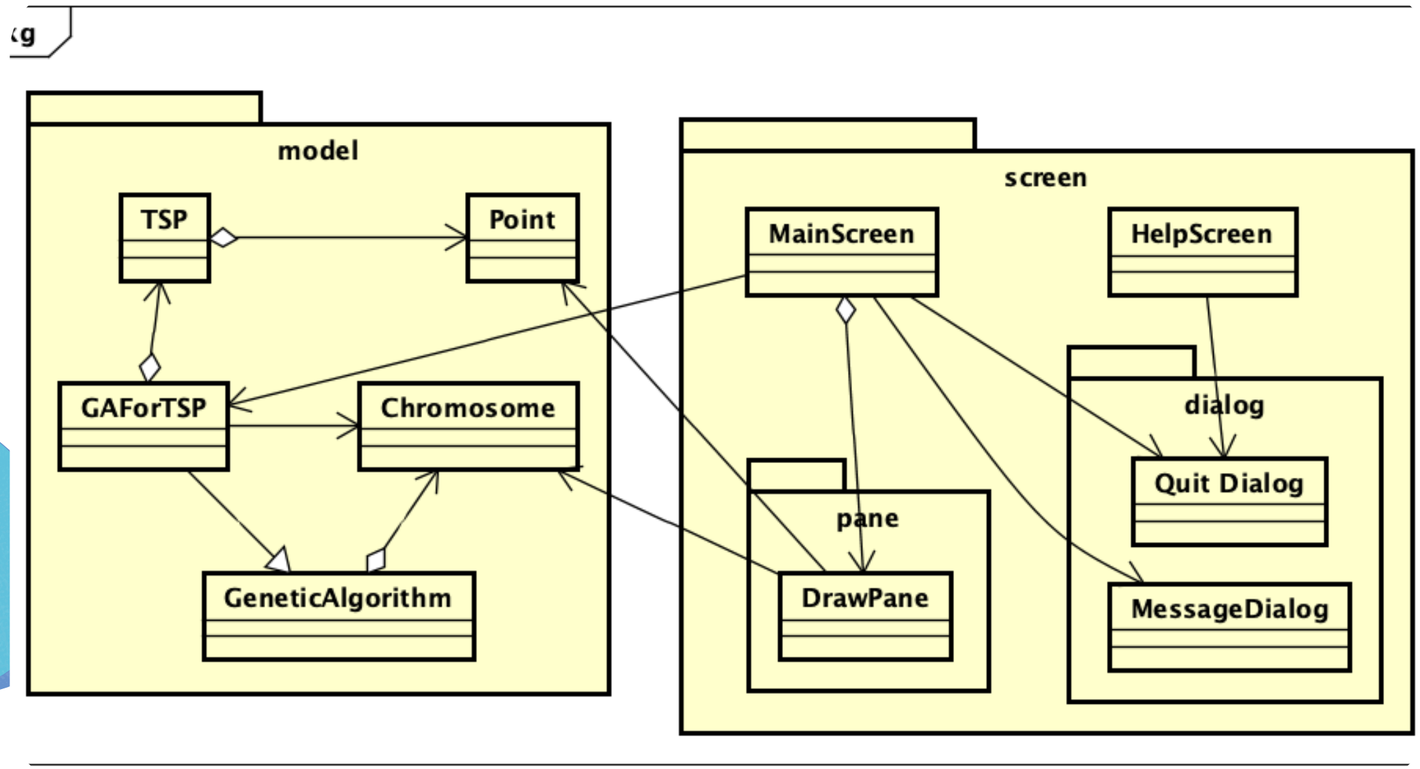
- Takes all user command from the GUI
- Notifies the user if there is any problem with the input
- Runs the algorithm and visualizes it to the user on the GUI
- Wait for the command of the user to control the process of the algorithm

User can view help menu

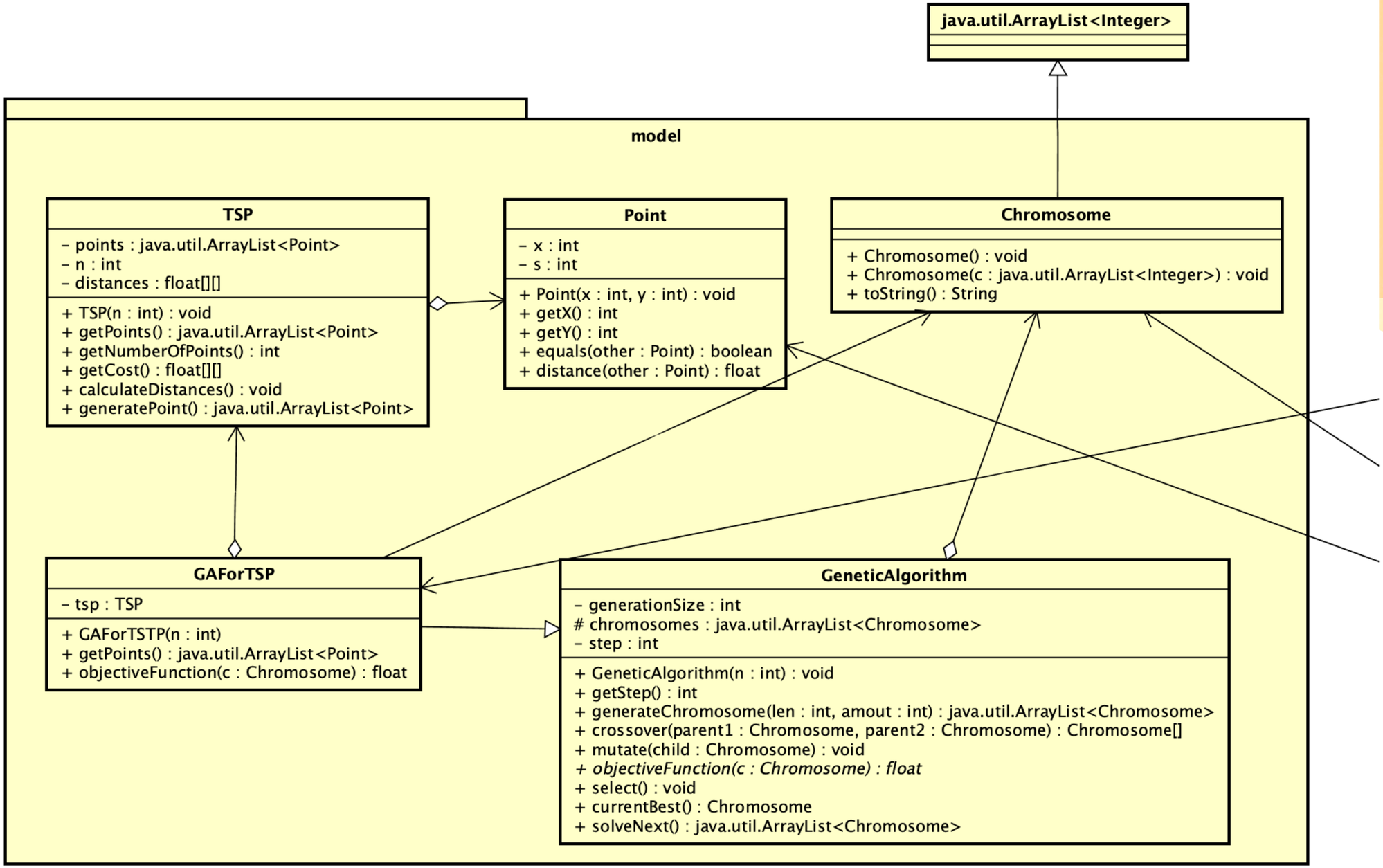
User can quit the program

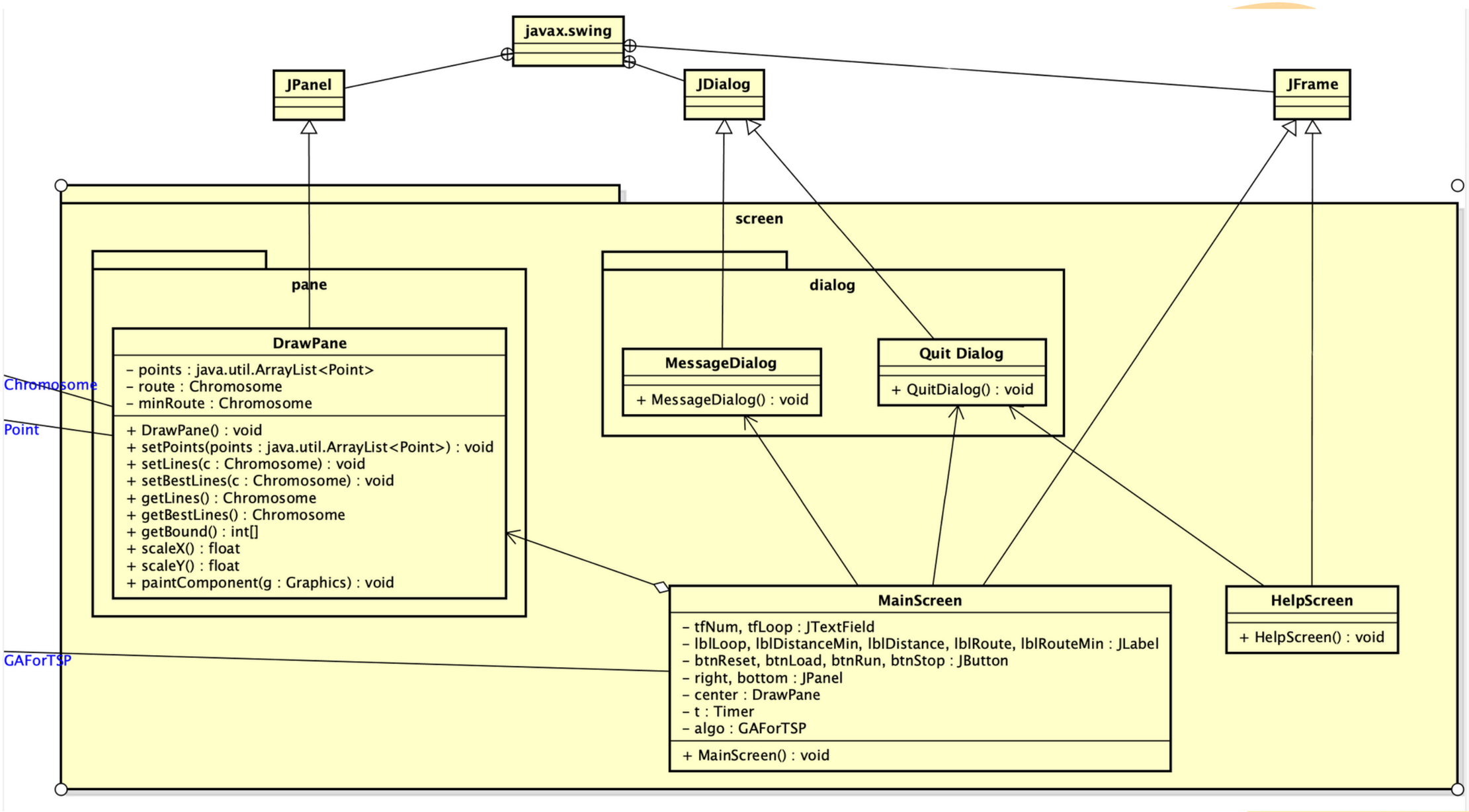


General class diagram



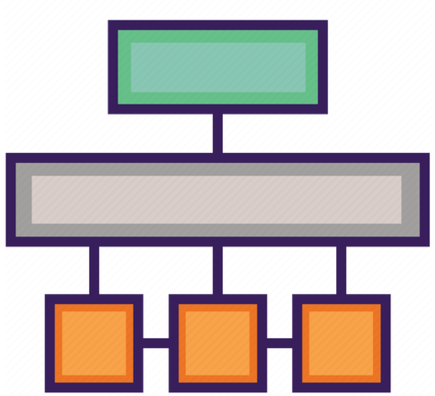
- Package “model”: Stores all basic data structures and objects to implement the genetic algorithm on traveling salesman problem
- Package “screen”: Stores the main screen class, dialog class, DrawPane class, and controllers to visualization in GUI


java.util.ArrayList<Integer>
model
TSP
- points : java.util.ArrayList<Point>
- n : int
- distances : float[][]
+ TSP(n : int) : void
+ getPoints() : java.util.ArrayList<Point>
+ getNumberOfPoints() : int
+ getCost() : float[][]
+ calculateDistances() : void
+ generatePoint() : java.util.ArrayList<Point>
Point
- x : int
- s : int
+ Point(x : int, y : int) : void
+ getX() : int
+ getY() : int
+ equals(other : Point) : boolean
+ distance(other : Point) : float
Chromosome
+ Chromosome() : void
+ Chromosome(c : java.util.ArrayList<Integer>) : void
+ toString() : String
GAForTSP
- tsp : TSP
+ GAForTSTP(n : int)
+ getPoints() : java.util.ArrayList<Point>
+ objectiveFunction(c : Chromosome) : float
GeneticAlgorithm
- generationSize : int
chromosomes : java.util.ArrayList<Chromosome>
- step : int
+ GeneticAlgorithm(n : int) : void
+ getStep() : int
+ generateChromosome(len : int, amout : int) : java.util.ArrayList<Chromosome>
+ crossover(parent1 : Chromosome, parent2 : Chromosome) : Chromosome[]
+ mutate(child : Chromosome) : void
+ objectiveFunction(c : Chromosome) : float
+ select() : void
+ currentBest() : Chromosome
+ solveNext() : java.util.ArrayList<Chromosome>


javax.swing
JPanel
JDialog
JFrame
screen
pane
DrawPane
- points : java.util.ArrayList<Point>
- route : Chromosome
- minRoute : Chromosome
+ DrawPane() : void
+ setPoints(points : java.util.ArrayList<Point>) : void
+ setLines(c : Chromosome) : void
+ setBestLines(c : Chromosome) : void
+ getLines() : Chromosome
+ getBestLines() : Chromosome
+ getBound() : int[]
+ scaleX() : float
+ scaleY() : float
+ paintComponent(g : Graphics) : void
dialog
MessageDialog
+ MessageDialog() : void
Quit Dialog
+ QuitDialog() : void
Chromosome
Point
GAForTSP
MainScreen
- tfNum, tfLoop : JTextField
- lblLoop, lblDistanceMin, lblDistance, lblRoute, lblRouteMin : JLabel
- btnReset, btnLoad, btnRun, btnStop : JButton
- right, bottom : JPanel
- center : DrawPane
- t : Timer
- algo : GAForTSP
+ MainScreen() : void
HelpScreen
+ HelpScreen() : void

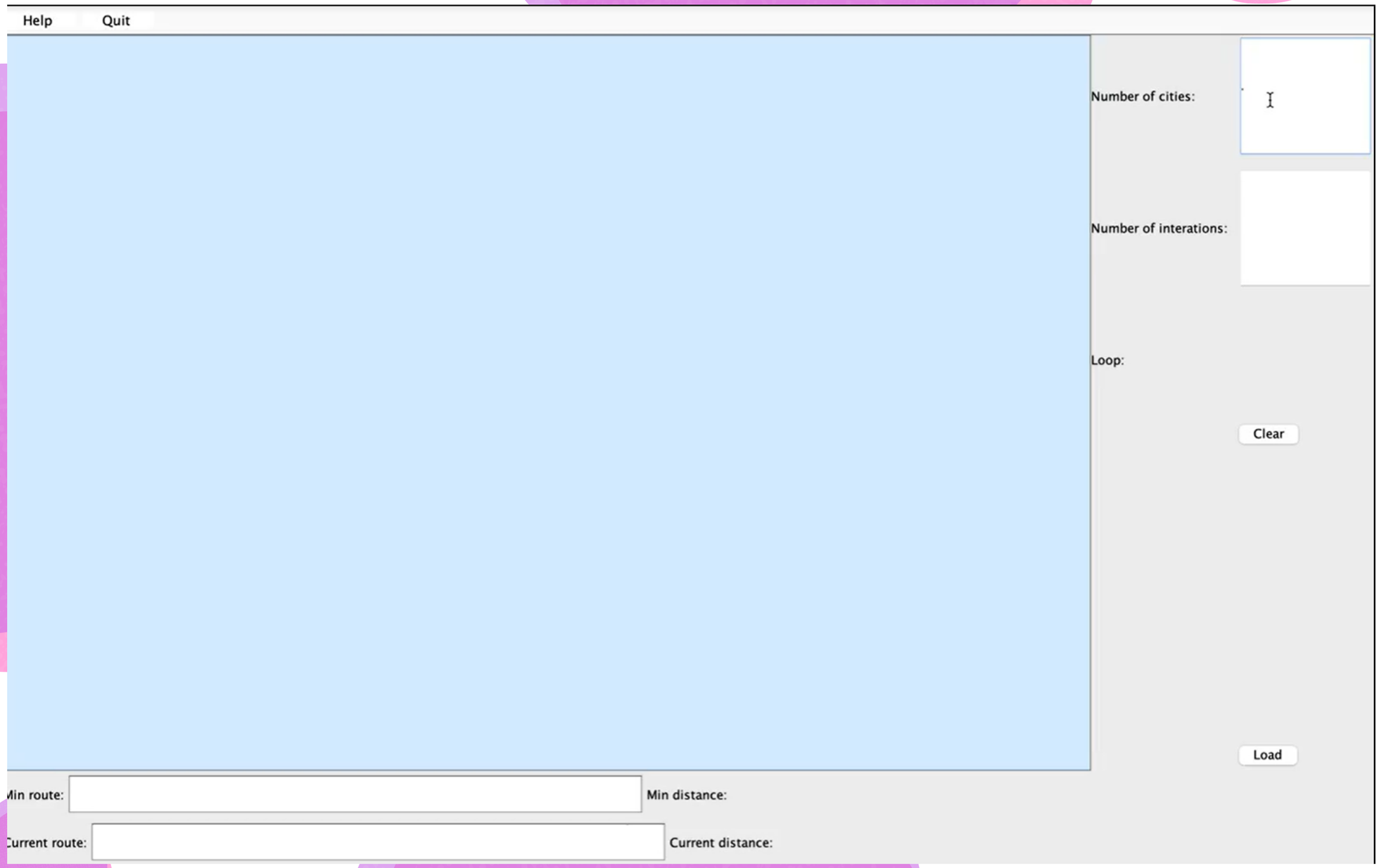
OOP TECHNIQUE

Inheritance



In package model: GAForTSP class inherits from GeneticAlgorithm class to implement genetic algorithm to solve traveling salesman problem

Demo


Help
Quit
Number of cities:
Number of interations:
Loop:
Clear
Load
Min route:
Min distance:
Current route:
Current distance: